DEPARTMENT OF INFORMATION AND COMPUTING SCIENCES

UTRECHT UNIVERSITY

FEBRUARY 2018

Commonsense Reasoning and Argumentation

Author:

HENRY PRAKKEN

Contents

Preface

Logic deals with the formal principles and criteria of validity of patterns of inference. This reader[1] discusses logics for a particular group of patterns of inference, viz. inferences that are not absolutely certain, but that can still be rationally made as long as they cannot be defeated on the basis of information to the contrary (hence the term *defeasible reasoning*). Such patterns can especially be found in commonsense reasoning, i.e., the inferences humans make in their daily life (hence the words *commonsense reasoning* in the title of this reader). Defeasible reasoning lacks one important property of 'standard' or 'deductive' reasoning, viz. the property of *monotonicity* of their inference relation. Since defeasible inferences are not absolutely certain, it may happen that conclusions inferrable from a particular body of information, are not inferrable from an extended body of information (hence the often-used term *nonmonotonic logic*).

Nonmonotonic notions of logical consequence have been studied in artificial intelligence since 1980. The course *Commonsense Reasoning and Argumentation*, for which this reader is written, treats three of the most important nonmonotonic logics, viz. default logic, circumscription, and logics for defeasible argumentation. In addition, dynamic aspects of argumentation will be studied, as well as dialogue systems for multi-agent argumentation.

For default logic a paper will be used that is available online. Circumscription is introduced in this reader in Chapter 1, written especially for this course. The largest part of this reader is devoted to argumentation. Abstract argumentation is discussed in Chapters 3 and 5 (based on and extending Prakken and Vreeswijk (2002) and Vreeswijk and Prakken (2000)), a logical framework for structured argumentation is discussed in Chapter 6 (based on Modgil and Prakken (2014)), dynamic aspects of argumentation are discussed in Chapter 7 (written especially for this course) and, finally, dialogue systems for agent interaction with argumentation are the topic of Chapter 8 (based on Prakken (2006)),

Exercises on default logic and circumscription can with their answers be found in Chapter 2. These exercises and answers are partly taken from earlier collections developed by Rogier van Eijk and Cees Witteveen. Exercises on argumentation can be found at the end of the relevant chapters, while their answers are in Chapter 9.

[1] Thanks are due to the students of earlier years and in particular to Bas van Gijzel, Marc van Zee and Elisa Friscione, for their corrections to previous versions of this reader.

Chapter 1

Circumscription

In this chapter we explore a semantic approach to nonmonotonic reasoning based on the idea of minimal models. Originally, this approach, which is often called *minimal-model semantics* or *preferential entailment*, was meant as a semantics for a nonmonotonic logic called *circumscription*. In this section we briefly summarise the circumscription logic and then present its model-theoretic semantics in detail. To this end, we first recall the main notions of the model-theoretic semantics of first-order predicate logic (FOL).

Recall that the semantics of a logic defines the notion of logical consequence in terms of the meaning of its logical language. In particular, a model-theoretic semantics defines the meaning of a sentence as the way the world looks like if the sentence is true. For instance, in FOL the sentence *Birds can fly* is true just in case all objects in the world that have the property of being a bird, also have the property of being able to fly. A formula is then a logical consequence of a theory iff the formula has to be true whenever all formulas of the theory are true.

Of course, it is impossible to display the actual world in a definition. Therefore, a model-theoretic semantics abstracts the world into a structure which contains only those features which are relevant for interpreting a logical language and it turns a structure into a model by interpreting the logical language in terms of these features. For FOL these features are a set of objects (the domain), and various functions and relations defined over this set.

Definition 1.0.1 [Structures and models in FOL.]

- A *structure* is a triple $\langle D, F, R \rangle$ such that D (the domain) is a set of objects, F is a set of functions on D and R a set of relations over D, i.e., a set of subsets of D^n (where D^n is the set of all n-tuples with elements from D).

- Let $\langle D, F, R \rangle$ be a structure. An *interpretation function* I of a first-order language $\mathcal{L}$ is a function that to each term t from $\mathcal{L}$ assigns an object $I(t) \in D$, to each function symbol f from $\mathcal{L}$ assigns a function $I(f) \in F$ and to each n-ary predicate symbol P^n from $\mathcal{L}$ assigns a set of n-tuples of objects $I(P) \subseteq D^n$ ($I(P)$ is also called the *extent* of P).

- A *model* for a language $\mathcal{L}$ is a pair S, I where S is a structure and I is an interpretation function for $\mathcal{L}$.

- A model M is a *model of* a set of formulas T iff all formulas of T are true in M (where truth is defined with the usual truth definitions). A formula φ is (classically) *entailed* by a set T of formulas iff φ is true in all models of T.

These definitions must be combined with the usual truth definitions for atomic formulas, the connectives and the quantifiers. The classical notion of entailment is then monotonic for the following reason. Given the usual truth definitions, enlarging a theory can only remove some of the models of the old theory as models of the enlarged one: it can never create new models. Therefore, everything that is true in all models of the old theory, is also true in the new one.

1.1 The basic idea: model preference

How can a model-theoretic account of nonmonotonic reasoning be developed? The crucial observation is that to define nonmonotonic entailment, we cannot look at *all* models of the premises. Consider the following example.

Example 1.1.1 The theory T consists of

(1) $\forall x((\mathit{Bird}(x) \wedge \neg \mathit{Ab}(x)) \supset \mathit{Canfly}(x))$
(2) $\mathit{Bird}(\mathit{Tweety})$

Formula (1) expresses the default that birds normally fly, and the only thing T tells us about Tweety is that it is a bird, so we would like to nonmonotonically conclude from T that $\mathit{Canfly}(\mathit{Tweety}))$. However, it is easy to verify that this formula is not classically entailed by T: even though T does not *entail* $\neg \mathit{Ab}(\mathit{Tweety})$, this formula is still *consistent* with T, so it is possible to construct models of T in which Tweety is abnormal and cannot fly.

Now the idea of minimal-model semantics is that, in verifying whether a formula is nonmonotonically entailed by a theory, we restrict our attention to those models of the theory in which things are as normal as possible. More precisely, we inspect only those models of the theory in which as few objects as possible are in the extent of the Ab predicate (hence the term minimal-model semantics). If a formula is true in all these models, it is nonmonotonically entailed by the theory. This new notion of entailment is nonmonotonic since, even though enlarging a theory can only remove some of its models, it may happen that some old models that were not minimal are minimal models of the new theory.

Let us apply these ideas to our example by looking only at those models of T in which the extent of the Ab predicate is as small as possible. Clearly, in all those models the object denoted by *Tweety* belongs to the extent of the *Bird* predicate. Furthermore, all those models satisfy (1). Now since $\mathit{Bird}(\mathit{Tweety})$ is true in all models of T, (1) can only be true in a model of T if in that model the object denoted by *Tweety* either belongs to the extent of the *Canfly* predicate, or belongs to the extent of the *Ab* predicate, or belongs to both (to verify this, apply the truth definition of the material implication). Thus the models of our theory split into three classes. Clearly, the abnormality-minimal class of models is the one in which the object denoted by *Tweety* does not belong to the extent of the *Ab* predicate. But in all those models that object belongs to the extent of the *Canfly* predicate, otherwise these models would not satisfy (1). So all abnormality-minimal models of T satisfy the sentence $\mathit{Canfly}(\mathit{Tweety})$, and so this sentence is nonmonotonically entailed by T.

The main task now is to formally define when a model is minimal. The key idea here is that of minimising, or 'circumscribing' the extent of predicates in a model. First, however, the original circumscription logic will be briefly explained.

1.2 Syntactic form of circumscription

Circumscription was originally formulated by McCarthy (1980) as a syntactic method. The idea was to minimise the extent of a predicate P in a first-order theory by adding a new formula to the theory which (informally) says 'those objects of which T *says* that they have the property are the *only* objects that have the property P'. To specify this formula, the following shorthands are convenient:

$$\begin{array}{lll} P = Q & \text{means} & \forall x[P(x) \equiv Q(x)] \\ P \leq Q & \text{means} & \forall x[P(x) \supset Q(x)] \\ P < Q & \text{means} & P \leq Q \text{ but } P \neq Q \end{array}$$

Let $T(P)$ be a first-order sentence with predicate constant P. The *circumscription of* P *in* $T(P)$ is:

$$T^*(P) =_{Def} T(P) \wedge \neg\exists p[T(p) \wedge p < P]$$

Here $T(p)$ is the formula resulting from substituting all occurences of P in T with p. The idea then is to infer nonmonotonic conclusions from T by reasoning classically with $T^*(P)$. In our example (letting T be the conjunction of (1) and (2)) $T^*(Ab)$ classically entails $\forall x \neg Ab(x)$ so that $T^*(Ab)$ also entails $Canfly(Tweety$.

It turns out that (under certain conditions) the classical models of $T^*(P)$ coincide with the minimal models of T, i.e., with those models of T in which the extent of P is minimal. This correspondence clarifies that, although the reasoning from $T^*(P)$ is classical and therefore monotonic, circumscription still models nonmonotonic reasoning, since conclusions of $T^*(P)$ may not be conclusions of $T'^*(P)$ for a T' that extends T. For example, if the theory $T(Ab)$ of our example is extended with $Ab(Tweety)$ to $T'(Ab)$, then $T'^*(Ab)$ does not classically entail $\forall x \neg Ab(x)$ so it neither classically entails $Canfly(Tweety)$. Semantically this means that there are minimal models of $T'(Ab)$ that are non-minimal models of $T(Ab)$.

At this point the reader will wonder how the classical reasoning with circumscription formulas takes place. In fact, this is rather complicated and what is worse, in general this reasoning cannot be done in first-order predicate logic since $T^*(P)$ quantifies not only over objects but also over predicates. So in general circumscriptive reasoning takes place in second-order logic, which is known to be intractable and even incomplete. Does this mean that circumscription is useless for practical purposes? Fortunately, this is not the case, since for several special classes of theories $T(P)$ the circumscription $T^*(P)$ turns out to be first-order. A particularly useful class is when the only formulas containing the predicate P are material implications with an atomic formula $Px_1, \ldots x_n$ in its consequent and no occurrences of P in its antecedent. In this special case the circumscription formula implies the so-called *completion* of the predicate P. The completion of a predicate can be computed as follows.

Definition 1.2.1 [Predicate completion] Let Px_i (where $x_i = x_1, \ldots x_n$) be an atomic formula and $T =$

$\{\forall x_i(\varphi_1 \supset Px_i)$

.

.

$\forall x_i(\varphi_n \supset Px_i)\}$

such that P does not occur in $\varphi_1 \ldots \varphi_n$. The *completion* of P in T is

$$\forall x_i(\varphi_1 \vee \ldots \vee \varphi_n \equiv Px_i)$$

To understand this definition, note that T is equivalent with $\forall x_i(\varphi_1 \vee \ldots \vee \varphi_n \supset Px_i)$

It turns out that for many practical purposes predicate completion is all that is needed. Therefore, in this course the full syntactic version of circumscription will be left untreated. Its model-theoretic version, on the other hand, will be discussed in detail, for two reasons. Firstly, it gives a semantics to the method of completion and secondly, the idea of minimal-model semantics is much more widely applicable than just to circumscription. If the minimality criterion for first-order models is generalised to any preference relation on models for any classical logic, then the result is the semantics of *preferential entailment*. To verify whether a conclusion is preferentially entailed by a theory, we only need to inspect the *preferred* models of the theory (according to some given preference criterion) and verify whether the conclusion is true in all of these preferred models.

1.3 A semantic model preference relation

How can the idea of preferential entailment be applied to the minimisation of predicates? Consider a theory T with a predicate P that is to be minimised. As a first approximation we can say that a model M of T is P-smaller than a model M' of T iff they have the same domain and if the extent of P in M is a subset of the extent of P in M'. However, this definition has to be refined, since we have to allow for the minimisation of more than one predicate.

Example 1.3.1 Consider the defaults

(1) $\forall x((Bird(x) \wedge \neg Ab(x)) \supset Canfly(x))$
(2) $\forall x((Penguin(x) \wedge \neg Ab(x)) \supset \neg Canfly(x))$

If *Tweety* is a penguin, we want to say that *Tweety* is an abnormal bird, but this should not imply that *Tweety* also is an abnormal penguin. To avoid this, we need two abnormality predicates Ab_1 and Ab_2, capturing, respectively, being abnormal with respect to the birds default and being abnormal with respect to the penguin default.

(1') $\forall x((Bird(x) \wedge \neg Ab_1(x)) \supset Canfly(x))$
(2') $\forall x((Penguin(x) \wedge \neg Ab_2(x)) \supset \neg Canfly(x))$

Accordingly, the model preference relation should be refined follows. Let $\mathbf{P}$ be a *set* of predicates to be minimised. Then a model M of a theory T is $\mathbf{P}$-smaller than a model M' of T iff they have the same domain and if of every predicate in $\mathbf{P}$ the extent in M is a subset of that in M'.

In sum, in circumscription each theory comes with a specification which predicates are to be minimised. Such a specification is called a *circumscription policy*, and a theory plus circumscription policy is called a *circumscriptive theory*.

Definition 1.3.2 Let T be a set of sentences of first-order predicate logic and $\mathbf{P}$ a set of predicates to be minimised. Then $\mathbf{P}$ is a *circumscription policy*, and $T^{\mathbf{P}}$ is a *circumscriptive theory*.

We can now give the formal definition of the semantic model preference relation for circumscription.

Definition 1.3.3 [Model preference.] Let $\mathbf{P}$ be a circumscription policy and M_1 and M_2 two models. We write $M_1 \leq^{\mathbf{P}} M_2$ iff

- M_1 and M_2 have the same domain; and
- $I_{M_1}(P) \subseteq I_{M_2}(P)$ for every predicate $P \in \mathbf{P}$.

In other words, $M_1 \leq^{\mathbf{P}} M_2$ means that M_1 and M_2 may differ only in how they interpret the predicates, and the extent of every predicate from $\mathbf{P}$ in M_1 is a subset of its extent in M_2.

The predicates in $\mathbf{P}$ need not be unary; the definition also applies to relations with arity higher than 1. In the latter case the extent of a predicate is not a set of objects but a set of tuples of objects. An example of a default with a twoplace abnormality predicate is ‘usually, married couples love each other’, which can be formalised as:

$$\forall x \forall y((\mathit{Married}(x,y) \wedge \neg \mathit{Ab}(x,y)) \supset (\mathit{Loves}(x,y) \wedge \mathit{Loves}(y,x)))$$

Since the relation $\leq^{\mathbf{P}}$ is transitive and reflexive, we can talk about the models that are *minimal* relative to this relation. A model M is a $\leq^{\mathbf{P}}$-minimal model of a theory T iff there is no model M' of T such that $M' <^{\mathbf{P}} M$.

We can now formally define the nonmonotonic notion of entailment, which we will call ‘minimal entailment’.

Definition 1.3.4 a sentence φ is *minimally entailed* by a circumscriptive theory $T^{\mathbf{P}}$, or $T^{\mathbf{P}} \vdash_{min} \varphi$, iff φ is true in all $\leq^{\mathbf{P}}$-minimal models of T.

In applications of circumscription it is usually assumed that all objects in a domain have a unique name. The following example illustrates the need for this assumption.

Example 1.3.5 Consider the single default

(1) $\forall x((\mathit{Bird}(x) \wedge \neg \mathit{Ab}(x)) \supset \mathit{HasWings}(x))$

And assume that we also know that $\mathit{Bird}(\mathit{Tweety})$, $\mathit{Bird}(\mathit{Polly})$ and $Ab(\mathit{Polly})$. Then $\mathit{HasWings}(\mathit{Tweety})$ is not minimally entailed since there are minimal models in which $\mathit{Tweety} = \mathit{Polly}$ is true so in those models $Ab(\mathit{Tweety})$ is true.

Usually, the unique-name assumption is combined with the domain-closure assumption, which says that all objects in a domain have a name. If the domain is finite, then these two assumptions can be expressed as first-order predicate logic sentences. Suppose $c_1, \ldots, c_n$ are all ground terms of the language. Then the *domain closure axiom* is

$$\forall x(x = c_1 \vee \ldots x = c_n)$$

And the *unique names axiom* is

$$c_1 \neq c_2 \wedge \ldots c_1 \neq c_n \wedge \ldots c_{n-1} \neq c_n$$

The conjunction of these two axioms for a tuple $c_1, \ldots, c_n$ is sometimes denoted as UNA$[c_1, \ldots, c_n]$.

In our example, these axioms amount to:

$$\forall x(x = \mathit{Tweety} \vee \ldots x = \mathit{Polly})$$

$$\mathit{Tweety} \neq \mathit{Polly}$$

The latter axiom excludes the undesired minimal models where $\mathit{Tweety} = \mathit{Polly}$ and $Ab(\mathit{Tweety})$ are true.

1.4 Examples

In this section some further examples will be discussed. Let us first look at some examples (adapted from Lifschitz 1994) where only one predicate P is minimised, i.e., $\mathbf{P} = P$ (when $\mathbf{P}$ is a singleton, the brackets will be omitted). These examples illustrate that circumscription formalises the idea that only those objects have a certain property that can be shown to have this property. Accordingly, we are interested whether a given theory minimally entails the completion of a minimised predicate P, i.e. whether all minimal models satisfy a formula of the form

$$\forall x(Px \equiv \varphi)$$

such that φ does not contain P.

Example 1.4.1 Let the theory T contain only Pa. Then all minimal models satisfy

$$\forall x(Px \equiv x = a)$$

This is because in all models of T the extent of P must contain a but need not contain any other object, so that in all minimal models of T the extent of P only contains a.

Example 1.4.2 Let T now contain only $\neg Pa$. Then all minimal models satisfy

$$\forall x(Px \equiv \bot)$$

which is equivalent to $\forall x \neg Px$.

Example 1.4.3 Next we consider a theory consisting of $Pa \wedge Pb$. Then all minimal models satisfy

$$\forall x(Px \equiv (x = a \vee x = b))$$

Example 1.4.4 Let T next consist of $Pa \vee Pb$. This theory does not minimally entail a completion of P; the strongest that is entailed is the following disjunction of two completions.

$$\forall x(Px \equiv x = a) \vee \forall x(Px \equiv x = b)$$

Example 1.4.5 A similar but slightly more complicated example is a theory T consisting of $Pa \ \vee \ (Pb \ \wedge \ Pc)$. It is easy to verify that all minimal models of T satisfy

$$\forall x(Px \equiv x = a) \vee \forall x(Px \equiv (x = b \vee x = c))$$

However, this can be strengthened by taking into account that a may be equal to b or c, in which case the second disjunctive term does not give a minimal P. So T also minimally entails

$$\forall x(Px \equiv x = a) \vee (\forall x(Px \equiv (x = b \vee x = c)) \wedge a \neq b \wedge a \neq c)$$

Example 1.4.6 Finally, we consider a theory T with $\forall x(Qx \supset Px)$. Minimising P transforms the implication into an equivalence. T minimally entails:

$$\forall x(Qx \equiv Px)$$

If instead Q is allowed to vary, a stronger result is obtained, viz.

$$\forall x(\neg Qx \wedge \neg Px)$$

Next two classic examples from nonmonotonic logic will be discussed, the ‘Tweety Triangle’ and the ‘Nixon Diamond’, starting with the Tweety Triangle, which extends Example 1.1.1 discussed above.

Example 1.4.7 Consider a circumscriptive theory $T^{\mathbf{P}}$ where $\mathbf{P} = \{Ab_1, Ab_2\}$ and T consists of

(1) $\forall x((\mathit{Bird}(x) \wedge \neg Ab_1(x)) \supset \mathit{Canfly}(x))$
(2) $\forall x(\mathit{Penguin}(x) \supset \mathit{Ab_1}(x))$
(3) $\forall x((\mathit{Penguin}(x) \wedge \neg Ab_2(x)) \supset \neg\mathit{Canfly}(x))$
(4) $\forall x(\mathit{Penguin}(x) \supset \mathit{Bird}(x))$
(5) $\mathit{Bird}(t)$

We are interested whether Tweety can fly, i.e., whether this theory minimally entails $\mathit{Canfly}(t)$. This is the case if $Ab_1(t)$ is false in all minimal models. It is easy to verify that this holds: the example extends Example 1.1.1 with a ‘Penguins cannot fly’ default and the information that all Penguins are birds; however, since it is not given that Tweety is a penguin, this additional information does not give rise to models where $\neg\mathit{Canfly}(t)$ and so $Ab_1(t)$ is true. In conclusion, $T^{\mathbf{P}}$ minimally entails that Tweety can fly.

Let us now extend T with the following information:

(6) $\mathit{Penguin}(t)$

The presence of both (2) and (6) in T makes that all models of T now satisfy $Ab_1(t)$. This enables minimal models of T that satisfy $\neg\mathit{Canfly}(t)$, so the new information has invalidated the previous conclusion that Tweety can fly. In fact, since $\neg Ab_2(t)$ is consistent with T, all minimal models of T satisfy this sentence, so they also all satisfy $\neg\mathit{Canfly}(t)$. Hence $T^{\mathbf{P}}$ now minimally entails that Tweety cannot fly.

Next we turn to the Nixon Diamond.

Example 1.4.8 Consider a circumscriptive theory $T^{\mathbf{P}}$ where $\mathbf{P} = \{Ab_1, Ab_2\}$ and T consists of

(1) $\forall x((\mathit{Quaker}(x) \wedge \neg Ab_1(x)) \supset \mathit{Pacifist}(x))$
(2) $\forall x((\mathit{Republican}(x) \wedge \neg Ab_2(x)) \supset \neg\mathit{Pacifist}(x))$
(3) $\mathit{Quaker}(n) \wedge \mathit{Republican}(n)$

We are interested whether Nixon was a pacifist, i.e. whether $T^{\mathbf{P}}$ minimally entails $\mathit{Pacifist}(n)$ or $\neg\mathit{Pacifist}(n)$. Clearly, no model of T can satisfy both $\neg Ab_1(n)$ and $\neg Ab_2(n)$, since that would require the model to satisfy both $\mathit{Pacifist}(n)$ and $\neg\mathit{Pacifist}(n)$. Also, models that satisfy both $Ab_1(n)$ and $Ab_2(n)$ can be made smaller by omitting $I(n)$ either from the extent of Ab_1 or from the extent of Ab_2. Doing the first results in minimal models of T that satisfy $\neg Ab_1(n)$ and therefore also satisfy $\mathit{Pacifist}(n)$, while doing the latter results in minimal models of T that satisfy $\neg Ab_2(n)$ and therefore also satisfy $\neg\mathit{Pacifist}(n)$. In conclusion, nothing of interest about Nixon’s Pacifism is minimally entailed by $T^{\mathbf{P}}$.

Finally, a well-known somewhat problematic example will be discussed. It is an extension of the Tweety Triangle (Example 1.4.7) with default information on when something is a penguin.

Example 1.4.9 Consider a circumscriptive theory $T^{\mathbf{P}}$ where $\mathbf{P} = \{Ab_1, Ab_2, Ab_3\}$ and T consists of

(1) $\forall x((Bird(x) \wedge \neg Ab_1(x)) \supset Canfly(x))$
(2) $\forall x(Penguin(x) \supset Ab_1(x))$
(3) $\forall x((Penguin(x) \wedge \neg Ab_2(x)) \supset \neg Canfly(x))$
(4) $\forall x((ObservedAsPenguin(x) \wedge \neg Ab_3(x)) \supset Penguin(x))$
(5) $\forall x(Penguin(x) \supset Bird(x))$
(6) $ObservedAsPenguin(t)$

Intuitively, we would expect that, as in Example 1.4.7, this theory also minimally entails $\neg Canfly(t)$. All that has changed is replacing the fact that Tweety is a penguin with a default 'Normally, if something is observed as a penguin, it is a penguin' and the fact that Tweety is observed as a penguin. And the information does not seem to give rise to an exception to this default.

However, perhaps surprisingly, the conclusion that Tweety cannot fly is not minimally entailed. The point is that $T \models Ab_1(t) \vee Ab_3(t)$, which not only allows a minimal model satisfying $\neg Ab_3(t)$ and $Ab_1(t)$ but also one satisfying $\neg Ab_1(t)$ and $Ab_3(t)$.

Examples of this kind have been much discussed in the literature. Some have argued that, to obtain the intuitive outcome, the model preference relation must be refined. Others have blamed the material implication for the problems, and have proposed the use of a conditional operator that does not satisfy contraposition, such as default-logic's domain-specific inference rules.

1.5 Prioritised circumscription

As with default logic, prioritised variants have also been developed of circumscription. With circumscription the idea is that some predicates are minimised with higher priority than other predicates. In this section only a brief sketch of this idea will be given; for the details the reader is referred to Baker and Ginsberg (1989).

Model-theoretically, the idea leads to a refinement of the model preference relation. Suppose, for instance, that in Example 1.4.8 the Republican default is regarded as stronger than the Quaker default. This can be captured by minimising Ab_2 with higher priority than Ab_1. Then a model in which $\neg Ab_2(n)$ holds at the expense of $Ab_1(n)$ is preferred over a model in which $\neg Ab_1(n)$ holds at the expense of $Ab_1(n)$, so that the conclusion $\neg Pacifist(n)$ is defeasibly entailed by the prioritised circumscriptive theory.

Chapter 2

Exercises on default logic and circumscription

2.1 Exercises on default logic

All exercises below which ask to determine extensions should, unless indicated otherwise, be answered by giving a process tree.

EXERCISE 2.1.1 Try to think of exceptions to the following rules, and to the eventual exceptions.

1. If a kept object is released, it will fall.
2. Tomatoes are red.
3. One ought to stop in front of a red light.
4. Presidents of the USA are male.
5. A bachelor is unmarried.

EXERCISE 2.1.2 Show that the default theory with $W = \varnothing$ and the following set of defaults:

$$D = \left\{ \frac{: p}{\neg q}, \frac{: q}{\neg r}, \frac{: r}{\neg s} \right\}$$

has only one extension.

EXERCISE 2.1.3 Determine the extensions of the default theory given by:

$$W = \{p \supset (\neg q \wedge \neg r)\}$$

$$D = \left\{ \frac{: p}{p}, \frac{: q}{q}, \frac{: r}{r} \right\}$$

EXERCISE 2.1.4 Show that the default theory with $W = \{p\}$ and the set of defaults below:

$$D = \left\{ \frac{p : r}{q}, \frac{p : s}{\neg q} \right\}$$

has no extension.

EXERCISE 2.1.5 Determine the extensions of the following default theories.

1. The default theory (W_1, D_1) with

$$W_1 = \{d, a \supset \neg b, d \supset \neg c\}$$

$$D_1 = \left\{ \frac{d : a}{a}, \frac{\neg c : b}{b}, \frac{b : e}{e}, \frac{b \wedge d : \neg e}{\neg e} \right\}$$

2. The default theory (W_2, D_2) with

$$W_2 = \{a, d, e \supset \neg c\}$$

$$D_2 = \left\{ \frac{a : b \wedge c}{b}, \frac{d : \neg b}{\neg b}, \frac{: d \wedge e}{e}, \frac{: \neg e}{\neg e} \right\}$$

3. The default theory (W_3, D_3) with

$$W_3 = \{a, (b \vee e) \supset \neg d\}$$

$$D_3 = \left\{ \frac{a : b}{b}, \frac{a : c}{c}, \frac{c : \neg b}{\neg b}, \frac{\neg b : e}{e}, \frac{: d}{d} \right\}$$

EXERCISE 2.1.6

1. Determine the extensions of the default theory given by:

$$W_1 = \{p\}$$

$$D_1 = \left\{ \frac{p : q, \neg q}{r} \right\}$$

2. Answer the same question for the following default theory:

$$W_2 = \{p\}$$

$$D_2 = \left\{ \frac{p : q \wedge \neg q}{r} \right\}$$

Compare your answer to that of 1.

EXERCISE 2.1.7 Answer Exercise 2.1.5(1,3) for Prioritised Default logic, given the following partial default orderings:

- (1) $\frac{b \wedge d : \neg e}{\neg e} < \frac{b : e}{e}$
- (3) $\frac{c : \neg b}{\neg b} < \frac{a : b}{b}, \frac{: d}{d} < \frac{a : c}{c}$

EXERCISE 2.1.8 Show that if a default theory has an inconsistent extension, this extension is its unique extension.

EXERCISE 2.1.9 Translate the defeasible rules and their exceptions from your answer to Exercise 2.1.1 into defaults.

EXERCISE 2.1.10 Consider the following default rules from the legal domain.

- *Drivers ought not to drive next to each other*
- *Cyclists are allowed to drive next to each other*
- *In case of danger for obstruction, cyclists ought not to drive next to each other*

Assume further as a hard fact that cyclists are drivers.

1. Translate this information into a propositional default theory which has a unique extension for each consistent W that includes these hard facts, and such that in case of conflicting defaults the most specific one is applied.
2. Answer the same question for Prioritised Default Logic.

EXERCISE 2.1.11 Consider the following empirical default rules.

- *Birds normally can fly*
- *Penguins normally cannot fly*

Assume further as hard facts that penguins are birds and that genetically modified penguins are abnormal penguins.

1. Translate this information with the help of abnormality predicates into a default theory (W, D) which has a unique extension for each consistent W that includes these hard facts, and such that in case of conflicting defaults the most specific one is applied.
2. Answer the same question for Prioritised Default Logic, using priorities instead of abnormality predicates.

EXERCISE 2.1.12 We define two consequence relations for default theories, one for *skeptical* reasoning ($\mid\!\sim^s$) and one for *credulous* reasoning ($\mid\!\sim^c$):

- $(W, D) \mid\!\sim^s \varphi =_{df}$ all extensions of (W, D) contain φ.
- $(W, D) \mid\!\sim^c \varphi =_{df}$ some extension of (W, D) contains φ.

Determine the skeptical consequences of the default theories of Exercise 2.1.5.

EXERCISE 2.1.13 A default theory (W, D) is called *finite* if D is finite. Can it be determined whether a default theory with

$$D = \left\{ \frac{Px : Qx}{Rx} \right\}$$

is finite? If so, is it finite? If not, which information is lacking?

EXERCISE 2.1.14 Consider a default theory $\Delta = (D, W)$ with the following set of defaults:

$$D = \{ \frac{\top : P(f(c))}{P(f(c))}, \frac{\top : P(f(f(c))}{P(f(f(c)))}, \dots \}$$

and

$$W = \{\forall x\, c \neq f(x), \forall x \forall y((f(x) = f(y)) \supset x = y), \forall x \forall y((P(x) \wedge P(y)) \supset x = y)\}$$

Show that this default theory has infinitely many extensions.

2.2 Exercises on circumscription

EXERCISE 2.2.1 Specify all models with one object for the theory of Example 1.1.1, and verify the line of reasoning in this example. Illustrate by extending the theory that the new entailment notion is nonmonotonic.

EXERCISE 2.2.2 In this exercise you should apply Definition 1.3.3. Consider a first-order language with object constants a and b, a unary predicate symbol P, a binary predicate symbol R and no other terms and predicate symbols.

1. Give all $\leq^P$-minimal models with two distinct objects d_1 and d_2 such that $I(a) = d_1$ and $I(b) = d_2$, for the following formulas:

   (a) $\neg Pa$

   (b) $Pa \vee Raa$

   (c) $Pa \vee \neg Pb$

   (d) $\exists x Px$

   (e) $\forall x \forall y (Px \supset Rxy)$

   (f) $\forall x \forall y (Rxy \supset Px)$

   (g) $Rab \wedge \forall x \forall y (Rxy \supset Px)$

   (h) $\forall x \forall y (\neg Px \supset Rxy)$

2. Give for each of the formulas under 1 one or more formulas that are true in all $\leq^P$-minimal models of the formula, but not in all its models.

EXERCISE 2.2.3

1. Consider a circumscriptive theory T^P such that $T = \{Pa, Rb\}$. Is $\neg Pb$ minimally entailed by T^P?

2. Formulate the unique-names and domain-closure axioms for T^P.

3. Consider T'^P which is formed from T^P by adding the unique-names and domain closure axioms. Is $\neg Pb$ minimally entailed by T'^P?

EXERCISE 2.2.4 Consider the circumscriptive theory T^{Ab} where T =

$\forall x((\mathit{Bird}(x) \wedge \neg \mathit{Ab}(x)) \supset \mathit{Canfly}(x))$
$\mathit{Bird}(\mathit{Sam})$
$\mathit{Ab}(\mathit{Tweety})$

1. Is $\mathit{Canfly}(\mathit{Sam})$ minimally entailed by T^{Ab}?

2. Extend T to T' by formulating the domain closure and unique-names axioms for T and adding them to T.

3. Is $\mathit{Canfly}(\mathit{Sam})$ minimally entailed by T'^{Ab}?

EXERCISE 2.2.5 Consider the following formulas:

$\forall x((P(x) \wedge \neg Ab(x)) \supset Q(x))$
$\forall x(R(x) \supset Ab(x))$
$P(a)$

Give all $\leq^{Ab}$-minimal models for this set of formulas. Is $Q(a)$ true in all these models? And what can you say about $R(a)$?

EXERCISE 2.2.6 Consider the following empirical default rules.

- *Birds normally can fly*
- *Penguins normally cannot fly*
- *Genetically modified penguins normally can fly*

Assume further as facts that all penguins are birds and all genetically modified penguins are penguins. Translate this information with the help of abnormality predicates into a circumscriptive theory. Ensure that in case of conflict the most specific default is applied; test this with minimal models, assuming that the language contains one constant, viz. *Tweety*.

EXERCISE 2.2.7 Consider a circumscriptive theory $T^{\mathbf{P}}$, where $\mathbf{P} = \{Ab_1, Ab_2, Ab_3, Ab_4\}$ and $T =$

(1) $\forall x((\mathit{BornInNL}(x) \wedge \neg Ab_1(x)) \supset \mathit{Dutch}(x))$
(2) $\forall x((\mathit{NorwegianName}(x) \wedge \neg Ab_2(x)) \supset \mathit{Norwegian}(x))$
(3) $\forall x((\mathit{Dutch}(x) \wedge \neg Ab_3(x)) \supset \mathit{LikesSkating}(x))$
(4) $\forall x((\mathit{Norwegian}(x) \wedge \neg Ab_4(x)) \supset \mathit{LikesSkating}(x))$
(5) $\forall x \neg(\mathit{Dutch}(x) \wedge \mathit{Norwegian}(x))$
(6) $\mathit{BornInNL}(\mathit{Brigt}) \wedge \mathit{NorwegianName}(\mathit{Brygt})$

Verify whether $\mathit{LikesSkating}(\mathit{Brigt})$ is minimally entailed by $T^{\mathbf{P}}$.

EXERCISE 2.2.8 Let T^P be a circumscriptive theory consisting of the following sentences:

$$a_i \neq a_j (0 \leq i < j \leq 3),$$

$$\forall x, y((S(x,y) \equiv [(x = a_1 \wedge y = a_0) \vee (x = a_2 \wedge y = a_1) \vee (x = a_3 \wedge y = a_2)]),$$

$$\forall x, y((\neg Px \wedge S(y,x)) \supset Py)$$

The a_i could, for instance, denote time units and the relation S could denote the successor relation, while P could be regarded as an abnormality predicate.
a) What is the strongest information on the extent of P that is minimally entailed by T^P?
b) Answer the same question when we add to T the sentence $\neg Pa_0$.

2.3 Answers to the Exercises

2.3.1 Default Logic

Below, defaults are assumed to be named as $d_1, \ldots, d_n$ in their order of appearance in the set of defaults.

Exercise 2.1.1:

1. Except in space, ...
2. Except if it is not ripe, or painted ...
3. Except police cars with their sirenes on, ...
4. No exceptions yet ...
5. No exceptions, since this is a lexical definition.

Exercise 2.1.2: The unique extension of this default theory is $Th(\{\neg q, \neg s\})$. The following process tree shows that there are no more extensions. The tree has four processes:

$$\begin{array}{l}
\Pi_1 = \{d_1, d_3\} \\
\Pi_2 = \{d_2, d_1\} \\
\Pi_3 = \{d_3, d_2\} \\
\Pi_4 = \{d_3, d_1\}
\end{array}$$

The first and last process are closed and successful, and lead to the same extension, while the other two are failed:

$$\begin{array}{ll}
In(\Pi_1[1]) = Th(\{\neg q\}) & Out(\Pi_1[1]) = \{\neg p\} \\
In(\Pi_1[2] = Th(\{\neg q, \neg s\}) & Out(\Pi_1[2]) = \{\neg p, \neg r\} \\
 & \\
In(\Pi_2[1]) = Th(\{\neg r\}) & Out(\Pi_2[1]) = \{\neg q\} \\
In(\Pi_2[2] = Th(\{\neg r, \neg q\}) & Out(\Pi_2[2]) = \{\neg q, \neg p\} \\
 & \\
In(\Pi_3[1]) = Th(\{\neg s\}) & Out(\Pi_3[1]) = \{\neg r\} \\
In(\Pi_3[2] = Th(\{\neg s, \neg r\}) & Out(\Pi_3[2]) = \{\neg r, \neg q\} \\
 & \\
In(\Pi_4[1]) = Th(\{\neg s\}) & Out(\Pi_4[1]) = \{\neg r\} \\
In(\Pi_4[2] = Th(\{\neg s, \neg q\}) & Out(\Pi_4[2]) = \{\neg r, \neg p\}
\end{array}$$

Exercise 2.1.3: The extensions are

- $E_1 = Th(W \cup \{p\}$, generated by the process d_1.
- $E_2 = Th(W \cup \{q, r\}$, generated by the processes d_2, d_3 and d_3, d_2.

The trick is to see that W makes that applying d_1 blocks both d_2 and d_3 and applying d_2 or d_3 blocks d_1.

Exercise 2.1.4: Any process that applies zero or one of the defaults is not closed, while the process Π that applies both defaults is failed: since both p and $\neg p$ are in $In(\Pi)$, every well-formed formula is $In(\Pi)$, and since s and r are in $Out(\Pi)$, we have that $In(\Pi) \cap Out(\Pi) \neq \varnothing$.

Exercise 2.1.5: Question 1: This default theory has 3 extensions:

$$\begin{array}{l}
E_1 = Th(W \cup \{a\}) \\
E_2 = Th(W \cup \{b, e\}) \\
E_3 = Th(W \cup \{b, \neg e\})
\end{array}$$

The corresponding process tree has three closed and successful branches:

$\Pi_1 = \{d_1\}$
$\Pi_2 = \{d_2, d_3\}$
$\Pi_3 = \{d_2, d_4\}$

The extensions are constructed as follows:

$In(\Pi_1[1]) = Th(W \cup \{a\}) \qquad Out(\Pi_1[1]) = \{\neg a\}$

$In(\Pi_2[1]) = Th(W \cup \{b\}) \qquad Out(\Pi_2[1]) = \{\neg b\}$
$In(\Pi_2[2]) = Th(W \cup \{b, e\}) \qquad Out(\Pi_2[2]) = \{\neg b, \neg e\}$

$In(\Pi_3[1]) = Th(W \cup \{b\}) \qquad Out(\Pi_3[1]) = \{\neg b\}$
$In(\Pi_3[2]) = Th(W \cup \{b, \neg e\}) \qquad Out(\Pi_3[2]) = \{\neg b, e\}$

No failed branches can be constructed. *Question 2*: This default theory has 3 extensions:

$E_1 = Th(W_2 \cup \{b, \neg e\})$
$E_2 = Th(W_2 \cup \{\neg b, e\})$
$E_3 = Th(W_2 \cup \{\neg b, \neg e\})$

E_1 is created by the processes d_1, d_4 and d_4, d_1
E_2 is created by the processes d_2, d_3 and d_3, d_2
E_3 is created by the processes d_2, d_4 and d_4, d_2
There is one failed process, viz. d_1, d_3.

Question 3: This default theory has 3 extensions:

$E_1 = Th(W_3 \cup \{b, c\})$
$E_2 = Th(W_3 \cup \{\neg b, c, d\})$
$E_3 = Th(W_3 \cup \{\neg b, c, e\})$

E_1 is created by the processes d_1, d_2 and d_2, d_1
E_2 is created by the processes d_2, d_3, d_5 and d_2, d_5, d_3 and d_5, d_2, d_3
E_3 is created by the process d_2, d_3, d_4.

Exercise 2.1.6:

1. This default theory has one extension, viz. $Th(\{p, r\})$, generated by applying the only default in D_1.

2. This default theory has a different unique extension, viz. $Th(\{p\})$, generated by the empty process. Note that $\neg(q \wedge \neg q)$ is in $In(\Pi)$ for any process Π.

Exercise 2.1.7:

(1) Only E_1 and E_3 are PDL-extensions of this theory, since the corresponding processes are generated by a total order containing $<$. By contrast, in Π_2 the default d_3 is applied while according to the priority ordering d_4 should have been applied instead. So E_2 is not generated by any total order containing $<$.

(3) E_3 is not a PDL-extension, since its generation requires that $d_2 \ll d_5$, which contradicts the fact that $d_5 < d_2$. However, E_1 and E_2 are also PDL-extensions: one ordering that generates E_1 is $d_3 \ll d_1 \ll d_5 \ll d_2 \ll d_4$, while one ordering that generates E_2

is $d_5 \ll d_3 \ll d_2 \ll d_1 \ll d_4$.

Exercise 2.1.8: Th. 4.5 of Antoniou implies that any default theory with an inconsistent extension has an inconsistent W. Then $Th(W)$ contains all well-formed formulas. But then no default is applicable to $In(\varnothing) = Th(W)$ so $\varnothing$ is a closed process, which means that no other extension than $Th(W)$ can be created (we already proved at HC3 that since $Out(\varnothing) = \varnothing$ we have that $\varnothing$ is successful, so that $Th(W)$ is an extension).

Exercise 2.1.9 Left to the student.

Exercise 2.1.10:

(1) with specific exception clauses:

$$D = \left\{ d_1 : \frac{driver : \neg next \wedge \neg cyclist}{\neg next}, d_2 : \frac{cyclist : next \wedge \neg danger}{next}, d_3 : \frac{cyclist \wedge danger : \neg next}{\neg next} \right\}$$

$$W = \{cyclist \supset driver\}$$

(1) with general exception clauses:

$$D = \left\{ d_1 : \frac{driver : \neg next \wedge \neg exc_1}{\neg next}, d_2 : \frac{cyclist : next \wedge \neg exc_2}{next}, d_3 : \frac{cyclist \wedge danger : \neg next \wedge \neg exc_3}{\neg next} \right\}$$

$$W = \{cyclist \supset driver, cyclist \supset exc_1, (cyclist \wedge danger) \supset exc_2\}$$

(2):

$$D = \left\{ d_1 : \frac{driver : \neg next}{\neg next}, d_2 : \frac{cyclist : next}{next}, d_3 : \frac{cyclist \wedge danger : \neg next}{\neg next} \right\}$$

$$d_3 < d_2 < d_1$$

$$W = \{cyclist \supset driver\}$$

Exercise 2.1.11:

$$D = \left\{ d_1 : \frac{Bird(x) : Flies(x) \wedge \neg Ab_1(x)}{Flies(x)}, d_2 : \frac{Penguin(x) : \neg Flies(x) \wedge \neg Ab_2(x)}{\neg Flies(x)} \right\}$$

$W =$
$\{\forall x(Penguin(x) \supset Bird(x)),$
$\forall x(Penguin(x) \supset Ab_1(x)),$
$\forall x((Penguin(x) \wedge GeneticallyModified(x)) \supset Ab_2(x))\}$

(2): Since the exception for genetically modified penguins intuitively is an 'undercutter' instead of a 'rebuttal' (i.e., it only blocks conclusions but does not support conclusions), the optimal formalisation in PDL is slightly contrived:

$D = \{d_1, d_2, d_3, d_4\}$ where

$$d_1 : \frac{Bird(x) \wedge \neg Ab_1(x) : Flies(x)}{Flies(x)}, d_2 : \frac{Penguin(x) \wedge \neg Ab_2(x) : \neg Flies(x)}{\neg Flies(x)}$$

$$d_3 : \frac{: \neg Ab_1(x)}{\neg Ab_1(x)}, d_4 : \frac{: \neg Ab_2(x)}{\neg Ab_2(x)}$$

$d_2 < d_1$
W is as under (1)

Exercise 2.1.12:

(1) $Th(W_1)$
(2) $Th(W_2)$
(3) $Th(W_3 \cup \{c\})$

Exercise 2.1.13: This can be determined only if it is known whether the set of terms in the object language is finite. If it is, the default theory is finite, otherwise it is infinite.

Exercise 2.1.14: Observe first that $P(f^i(c))$ and $P(f^j(c))$ (for $0 \leq i \neq j$) cannot be together in the same extension, since together with W these two formulas are inconsistent while W alone is consistent and therefore has no inconsistent extension. Consider next the following sets:

$$E_i = Th(W \cup \{P(f^i(c))\})$$

It is easy to verify that each E_i (for i a natural number) is an extension of Δ, created by applying exactly one default.

The first observation explained in more detail: suppose we apply to defaults to obtain $Pf^1(c)$ and $Pf^2(c)$. Then the third formula in W implies $f^1(c) = f^2(c)$. Then with the second formula in W this implies $c = f(c)$ but this contradicts the first formula in W. It is easy to see that this line of reasoning holds for any two (or more) defaults we apply, so we can apply just one default. Since we have infinitely many choices, we end up with infinitely many extensions.

2.3.2 Circumscription

Exercise 2.2.1:

M_1: $I(\mathit{Bird}) = \{\mathit{Tweety}\}$ $I(\mathit{Ab}) = \varnothing$ $I(\mathit{Canfly}) = \{\mathit{Tweety}\}$
M_2: $I(\mathit{Bird}) = \{\mathit{Tweety}\}$ $I(\mathit{Ab}) = \{\mathit{Tweety}\}$ $I(\mathit{Canfly}) = \{\mathit{Tweety}\}$
M_3: $I(\mathit{Bird}) = \{\mathit{Tweety}\}$ $I(\mathit{Ab}) = \{\mathit{Tweety}\}$ $I(\mathit{Canfly}) = \varnothing$

M_1 is the only Ab-minimal model and in this model $\mathit{Canfly}(\mathit{Tweety})$ is true, so it is true in all Ab-minimal models of T, so it is nonmonotonically entailed by T.

If T is extended with $\mathit{Ab}(\mathit{Tweety})$ then M_1 is not a model of the new theory and the remaining models M_2 and M_3 are both ab-minimal. In M_3 $\mathit{Canfly}(\mathit{Tweety})$ is false, so this formula is not nonmonotonically entailed by the new theory.

Exercise 2.2.2:

Question 1:

a:	$I(P) = \varnothing,$	$I(R) =$ any
b:	$I(P) = \varnothing,$	$\{(d_1, d_1)\} \subseteq I(R) \subseteq \{(d_1, d_1), (d_1, d_2), (d_2, d_1), (d_2, d_2)\}$
c:	$I(P) = \varnothing,$	$I(R) =$ any
d:	$I(P) = \{d_1\},$	$I(R) =$ any
	$I(P) = \{d_2\},$	$I(R) =$ any
e:	$I(P) = \varnothing,$	$I(R) =$ any
f:	$I(P) = \varnothing,$	$I(R) = \varnothing$
g:	$I(P) = \{d_1\},$	$I(R) = \{(d_1, d_2)\}$
	$I(P) = \{d_1\},$	$I(R) = \{(d_1, d_1), d_1, d_2)\}$
h:	$I(P) = \varnothing,$	$I(R) = \{(d_1, d_1), (d_1, d_2), (d_2, d_1), (d_2, d_2)\}$

Question 2:

a: $\forall x \neg Px$
b: $\neg Pa, \forall x \neg Px, Raa$
c: $\neg Pa, \forall x \neg Px$
d: $\neg \forall x Px, \forall x(Px \leftrightarrow x = a) \vee \forall x(Px \leftrightarrow x = b)$
e: $\forall x \neg Px$
f: $\forall x \neg Px$
g: $\forall x(Px \equiv x = a)$
h: $\forall x \forall y Rxy$

Exercise 2.2.3:

1. $\neg Pb$ is not minimally entailed. Consider a model M_1 with domain $\{d_1\}$ and with $I(a) = I(b) = \{d_1\}$: M_1 satisfies Pb and it is a $\leq^{\mathbf{P}}$-minimal model of T^P.

2. Unique-names: $a \neq b$. Domain closure: $\forall x(x = a \vee x = b)$.

3. Yes. M_1 is not a countermodel any more, since it does not satisfy the unique-names axiom.

Exercise 2.2.4:

1. No, since there is a minimal model of this theory in which $I(\mathit{Tweety}) = I(\mathit{Sam})$, $I(\mathit{Ab}) = \{\mathit{Sam}, \mathit{Tweety}\}$ and $\mathit{Sam} \notin I(\mathit{Canfly})$.

2. Unique-names: $\mathit{Tweety} \neq \mathit{Sam}$. Domain closure: $\forall x(x = \mathit{Tweety} \vee x = \mathit{Sam})$.

3. Yes. The countermodel of question (1) is not a model of T'^{Ab}, since it does not satisfy the unique-names axiom. The minimal models of T'^{Ab} are those in which $I(\mathit{Tweety}) \neq I(\mathit{Sam})$ and $I(\mathit{Ab}) = \{\mathit{Tweety}\}$, and in those models we have that $\mathit{Sam} \in I(\mathit{Canfly})$.

Exercise 2.2.5: Let $I(a) = d$. Then the minimal models are all models such that $d \in I(P)$, $I(R) = \varnothing$, $I(\mathit{Ab}) = \varnothing$, $d \in I(Q)$. So $Q(a)$ is true in all minimal models. Moreover, $R(a)$ is false in all minimal models.

Exercise 2.2.6:

$$\forall x((\mathit{Bird}(x) \wedge \neg \mathit{Ab}_{\mathit{1}}(x)) \supset \mathit{Canfly}(x))$$
$$\forall x((\mathit{Penguin}(x) \wedge \neg \mathit{Ab}_{\mathit{2}}(x)) \supset \neg \mathit{Canfly}(x))$$
$$\forall x((\mathit{Penguin}(x) \wedge \mathit{GeneticallyManipulated}(x) \wedge \neg \mathit{Ab}_{\mathit{3}}(x)) \supset \mathit{Canfly}(x))$$
$$\forall x(\mathit{Penguin}(x) \supset \mathit{Ab}_{\mathit{1}}(x))$$
$$\forall x((\mathit{Penguin}(x) \wedge \mathit{GeneticallyManipulated}(x)) \supset \mathit{Ab}_{\mathit{2}}(x))$$
$$\forall x(\mathit{Penguin}(x) \supset \mathit{Bird}(x))$$

Minimised predicates: Ab_1, Ab_2, Ab_3

Exercise 2.2.7: Yes, this is minimally entailed. This theory has two classes of minimal models. In one class $\mathit{Ab}_{\mathit{1}}(\mathit{Brigt})$ is true while $\mathit{Ab}_{\mathit{2}}(\mathit{Brigt})$, $\mathit{Ab}_{\mathit{3}}(\mathit{Brigt})$ and $\mathit{Ab}_{\mathit{4}}(\mathit{Brigt})$ are false, and in the other class of models t$\mathit{Ab}_{\mathit{2}}(\mathit{Brigt})$ is true while the other three abnormality expressions are false. In the first class of models $\mathit{Norwegian}(\mathit{Brigt})$ is true because of (2) and so $\mathit{LikesSkating}(\mathit{Brigt})$ is true because of (4). In the second class of models $\mathit{Dutch}(\mathit{Brigt})$ is true because of (1) and so $\mathit{LikesSkating}(\mathit{Brigt})$ is true because of (3).

Exercise 2.2.8: Let us inspect the minimal models (where $I(a_i) = a_i$).

(a) Assume $a_0 \in P$; since we look at minimal models, we then have that $a_1 \notin P$. Because of the content of T, a_2 must then be an element of P. Finally, we have $a_3 \notin P$. So $P = \{a_0, a_2\}$. If we inspect all models in this way, we obtain:

$$\forall x(P(x) \equiv x = a_1 \vee x = a_3) \vee \forall x(P(x) \equiv x = a_0 \vee x = a_2) \vee$$
$$\forall x(P(x) \equiv x = a_1 \vee x = a_2)$$

(b) We lose the minimal model $P = \{a_0, a_2\}$, since this is not a model of the new theory. So we obtain:

$$\forall x(P(x) \equiv x = a_1 \vee x = a_3)$$

Chapter 3

Argumentation logics: introduction

This chapter introduces another way to conceptualise nonmonotonic reasoning, viz. as patterns of inference where arguments for and against a certain claim are produced and evaluated, to test the tenability of the claim. In the present chapter some motivating examples will be presented and the main concepts will be informally introduced, while in Chapters 4$-$6 the formal theory of argumentation systems will be developed.

3.1 Motivating examples

We shall illustrate the idea of argumentation-based inference with a dispute between two persons, A and B. They disagree on whether it is morally acceptable for a newspaper to publish a certain piece of information concerning a politician's private life. Let us assume that the two parties have reached agreement on the following points.

(1) The piece of information I concerns the health of person P;
(2) P does not agree with publication of I;
(3) Information concerning a person's health is information concerning
that person's private life

A now states the moral principle that

(4) Information concerning a person's private life may not be published
if that person does not agree with publication.

and A says "So the newspapers may not publish I" (Fig. 3.1, page 30). Although B accepts principle (4) and is therefore now committed to (1-4), B still refuses to accept the conclusion that the newspapers may not publish I. B motivates her refusal by replying that:

(5) P is a cabinet minister
(6) I is about a disease that might affect P's political functioning
(7) Information about things that might affect a cabinet minister's
political functioning has public significance

Furthermore, B maintains that there is also the moral principle that

(8) Newspapers may publish any information that has public significance

B concludes by saying that therefore the newspapers may write about P's disease (Fig. 3.2, page 31). A agrees with (5–7) and even accepts (8) as a moral principle, but A does not give up his initial claim. Instead he tries to defend it by arguing that he has the stronger argument: he does so by arguing that in this case

(9) The likelihood that the disease mentioned in I affects P's functioning is small.
(10) If the likelihood that the disease mentioned in I affects P's functioning is small, then principle (4) has priority over principle (8).

Thus it can be derived that the principle used in A's first argument is stronger than the principle used by B (Fig. 3.3, page 31), which makes A's first argument stronger than B's, so that it follows after all that the newspapers should be silent about P's disease.

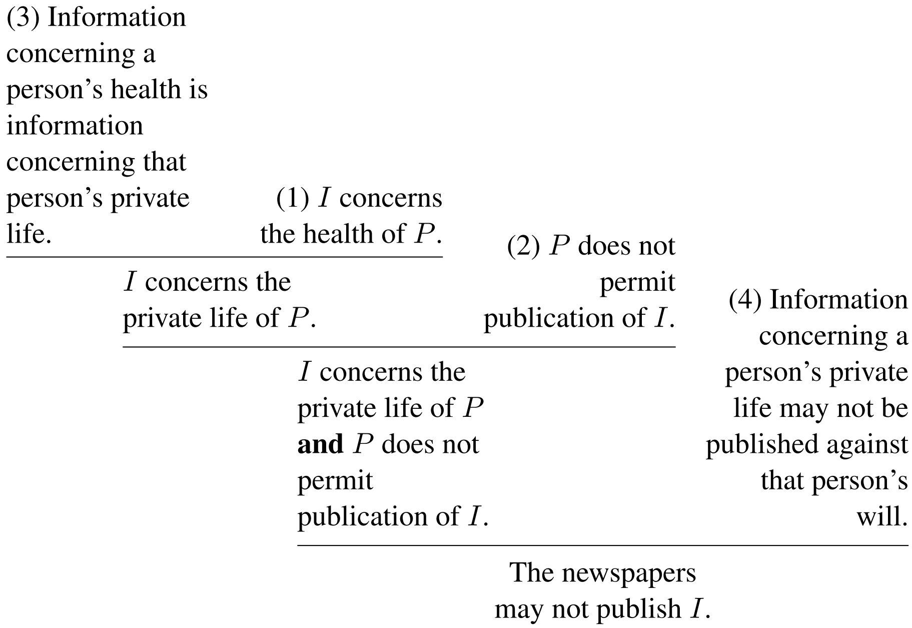

(3) Information concerning a person's health is information concerning that person's private life.
(1) I concerns the health of P.
I concerns the private life of P.
(2) P does not permit publication of I.
I concerns the private life of P **and** P does not permit publication of I.
(4) Information concerning a person's private life may not be published against that person's will.
The newspapers may not publish I.

Figure 3.1: A's argument.

Let us examine the various stages of this dispute in some detail. Intuitively, it seems obvious that the accepted basis for discussion after A has stated (4) and B has accepted it, viz. (1,2,3,4), warrants the conclusion that the piece of information I may not be published. However, after B's counterargument and A's acceptance of its premises (5-8) things have changed. At this stage the joint basis for discussion is (1-8), which gives rise to two conflicting arguments. Moreover, (1-8) does not yield reasons to prefer one argument over the other: so at this point A's conclusion has ceased to be warranted. But then A's second argument, which states a preference between the two conflicting moral principles, tips the balance in favour of his first argument: so after the basis for discussion has been extended to (1-10), we must again accept A's moral claim as warranted.

Logical systems that formalise this kind of reasoning are called 'argumentation logics', or 'argumentation systems'. As the example shows, these systems lack the monotonicity property of 'standard', deductive logic (say, first-order predicate logic, FOL). According to FOL, if A's claim is implied by (1–4), it is surely also implied by (1–8). From the point of view of FOL it is pointless for B to accept (1–4) and yet

$$\dfrac{\dfrac{\dfrac{\text{(5) } P \text{ is a cabinet minister.} \qquad \text{(6) } I \text{ is about a disease that might affect } P\text{'s political functioning.}}{I \text{ is about a disease that might affect a cabinet minister's political functioning.}} \qquad \text{(7) Information about things that might affect a cabinet minister's political functioning has public significance.}}{I \text{ has public significance.}} \qquad \text{(8) Newspapers may publish any information that has public significance.}}{\text{The newspapers may publish } I.}$$

Figure 3.2: B's argument.

$$\dfrac{\text{(9) The likelihood that the disease mentioned in } I \text{ affects } P\text{'s functioning is small.} \qquad \text{(10) If the likelihood that the disease mentioned in } I \text{ affects } P\text{'s functioning is small, then principle (4) has priority over principle (8).}}{\text{Principle (4) has priority over principle (8).}}$$

Figure 3.3: A's priority argument.

state a counterargument; B should also have refused to accept one of the premises, for instance, (4).

Does this mean that our informal account of the example is misleading, that it conceals a subtle change in the interpretation of, say, (4) as the dispute progresses? This is not so easy to answer in general. Although in some cases it might indeed be best to analyse an argument move like B's as a reinterpretation of a premise, in other cases this is different. In actual reasoning, rules are not always neatly labelled with an exhaustive list of possible exceptions; rather, people are often forced to apply 'rules of thumb' or 'default rules', in the absence of evidence to the contrary, and it seems natural to analyse an argument like B's as an attempt to provide such evidence to the contrary. When the example is thus analysed, the force of the conclusions drawn in it can only be captured by a consequence notion that is nonmonotonic: although A's claim is warranted on the basis of (1–4), it is not warranted on the basis of (1–8).

Argumentation logics are the most direct attempt to formalise examples like the above one, by defining notions like argument, counterargument, attack and defeat, and by defining nonmonotonic consequence in terms of the interaction of arguments for and against certain conclusions. This approach was initiated by the philosopher John

Pollock (Pollock, 1987), based on his earlier work in epistemology, e.g. (Pollock, 1974), and the AI researcher Ronald Loui (Loui, 1987).

One application of argumentation logics is to to formalise 'quick-and-dirty' commonsense reasoning with empirical generalisations. In everyday life people often reason with generalisations such as 'Birds fly', 'Italians usually like coffee', 'Chinese usually do not like coffee', 'Witnesses usually speak the truth' or 'When the streets are wet, it must have rained'. In commonsense reasoning, people apply such a generalisation if nothing is known about exceptions, but they are prepared to retract a conclusion if further knowledge tells us that there is an exception (for instance, a given bird is in fact a penguin, a witness has a reason to lie or the streets are wet because they are being cleaned).

However, argumentation systems have wider scope than just reasoning with such empirical generalisations. Firstly, argumentation systems can be applied to any form of reasoning with contradictory information, whether the contradictions have to do with generalisations and exceptions or not. For instance, the contradictions may arise from reasoning with several sources of information, or they may be caused by disagreement about beliefs or about moral, ethical or political claims. Moreover, it is important that several argumentation systems allow the construction and attack of arguments that are traditionally called 'ampliative', such as inductive, analogical and abductive arguments; these reasoning forms fall outside the scope of most other nonmonotonic logics.

One domain in which argumentation systems have become popular is legal reasoning. This is not surprising, since legal reasoning often takes place in an adversarial context, where notions like argument, counterargument, rebuttal and defeat are very common. Argumentation systems have also been applied in, for instance, the medical domain and in multi-agent models of negotiation and collaboration.

3.2 Argumentation systems: a conceptual sketch

In this section we give a conceptual sketch of the general ideas behind argumentation logics. First we sketch the general idea, and then we discuss the five main elements of such logics.

3.2.1 The general idea

Argumentation systems formalise nonmonotonic reasoning as the construction and comparison of arguments for and against certain conclusions. The idea is that the construction of arguments on the basis of a theory is monotonic, i.e., an argument stays an argument if the theory is enlarged with new information. Nonmonotonicity is explained in terms of the interactions between conflicting arguments: it arises from the fact that the new information may give rise to stronger counterarguments, which defeat the original argument. For instance, in case of Tweety the penguin we may construct one argument that Tweety flies because it is a bird, and another argument that Tweety does not fly because it is a penguin, and then we may prefer the latter argument because it is about a specific class of birds, and is therefore an exception to the general rule.

3.2.2 Five elements of argumentation systems

Argumentation systems contain the following five elements (although sometimes implicitly): an underlying logical language plus inference rules, definitions of an argument, of conflicts between arguments and of defeat between arguments and, finally, a definition of the dialectical status of arguments, which can be used to define a nonmonotonic notion of logical consequence.

A logical language plus inference rules

Argumentation systems are built around an underlying logical language and a set of inference rules defined over this language. Some systems assume a specific logical language and set of inferene rules, while other systems leave these things partly or wholly unspecified. The latter systems can thus be instantiated in alternative ways, which makes them frameworks rather than systems. An example of such a framework will be presented in Chapter 6.

Arguments

The notion of an argument corresponds to a tentative proof (or the existence of such a proof) in the 'logic' of the chosen logical language, where this 'logic' is expressed in the set of inference rules over the language. 'Logic' is here written between quotes because the logic does not need to be a standard deductive logic but can also contain defeasible inference rules (cf. the defaults of default logic). The nature of the inference rules of an argumentation system will be further discussed in Chapter 6. For now it suffices to say that the underlying logic of an argumentation system is still monotonic in the sense that new information cannot invalidate arguments as arguments but can only give rise to new counterarguments.

As for the layout of arguments, in the literature on argumentation systems three basic formats can be distinguished, all familiar from the logic literature. Sometimes arguments are defined as a tree of inferences grounded in the premises, and sometimes as a sequence of such inferences, i.e., as a deduction. Finally, some systems simply define an argument as a premises - conclusion pair, leaving implicit that the underlying logic validates a proof of the conclusion from the premises.

The notions of an underlying logic and an argument still fit with the standard picture of what a logical system is. The remaining three elements are what makes an argumentation system a framework for nonmonotonic reasoning.

Conflicts between arguments

The first is the notion of a *conflict* between arguments (also used are the terms 'attack' and 'counterargument'). In the literature, three types of conflicts are discussed. Firstly, arguments can be attacked on one of their premises, with an argument whose conclusion negates that premise. For example, an argument 'Tweety flies, because it is a bird' can be attacked by arguing that Tweety is not a bird. This kind of attack will in Chapter 6 be called *undermining* attack. The second type of attack is to negate the conclusion of an argument, as in 'Tweety flies, because it is a bird' and 'Tweety does not fly because it is a penguin' (cf. the left part of Fig. 3.4). Finally, when an argument uses a non-deductive, or *defeasible* inference rule, it can be attacked on its inference by arguing

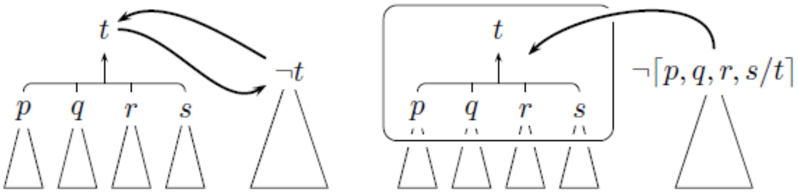


Figure 3.4: Rebutting attack (left) vs. undercutting attack (right).

that there is a special case to which the inference rule does not apply (cf. the right part of Fig. 3.4). After Pollock (1974, 1987), this is usually called *undercutting* attack. Unlike a rebutting attack, an undercutting attack does not negate the conclusion of its target but just says that its conclusion is not supported by its premises and can therefore not be drawn. In order to formalise this type of conflict, the rule of inference that is to be undercut (in Fig. 3.4: the rule that is enclosed in the dotted box, in flat text written as $p, q, r, s/t$) must be expressed in the object language: $\lceil p, q, r, s/t \rceil$) and denied: $\neg\lceil p, q, r, s/t \rceil$. [1] While all arguments can be attacked on their premises, only defeasible arguments can be attacked on their conclusion or inference. The reason why deductive arguments cannot be rebutted or undercut is that deductive inferences are by definition truth-preserving, i.e., the truth of their premises guarantees the truth of their conclusion, so the only way to disagree with the conclusion of a deductive argument is to deny one of its premsies. By contrast, the conclusion of a defeasible argument can be rejected even if all its premises are accepted. In Chapter 6 the difference between deductive and defeasible inference rules will be formalised and several examples of defeasible rules will be discussed. For now, consider the following example of a defeasible argument applying the principle of induction: the argument 'Raven $_{101}$ is black since the observed ravens raven$_1$ …raven$_{100}$ were black' is undercut by an argument 'I saw raven$_{102}$, which was white'.

Note, finally, that all three kinds of attack have a direct and an indirect version; indirect attack is directed against a subconclusion or a substep of an argument, as illustrated by Figure 3.5 for indirect rebutting.

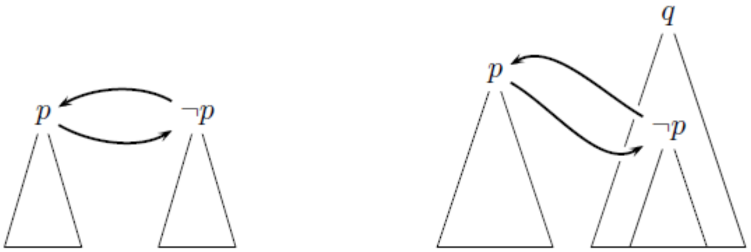


Figure 3.5: Direct attack (left) vs. indirect attack (right).

[1] Ceiling brackets around a meta-level formula denote a conversion of that formula to the object language, provided that the object language is expressive enough to enable such a conversion.

Defeat between arguments

The notion of conflicting, or attacking arguments does not embody any form of evaluation; evaluating conflicting pairs of arguments, or in other words, determining whether an attack is successful, is another element of argumentation systems. It has the form of a binary relation between arguments, standing for 'attacking and not weaker' (in a weak form) or 'attacking and stronger' (in a strong form). The terminology varies: some terms that have been used are 'defeat', 'attack' and 'interference'. Other systems do not explicitly name this notion but leave it implicit in the definitions. In this text we shall use 'defeat' for the weak notion and 'strict defeat' for the strong, asymmetric notion. Note that the several forms of attack, rebutting vs. assumption vs. undercutting and direct vs. indirect, have their counterparts for defeat.

Argumentation systems vary in their grounds for determining the defeat relations. Often only domain-specific criteria are available, which, moreover, are often defeasible. For this reason argumentation systems have been developed that allow for defeasible arguments on these criteria. To give some examples of domain-specific criteria, in domains where observations are important, defeat may depend on the reliability of tests, observers or sensors. In advice giving or consultancy, defeat may be determined by the level of expertise of the advisors or consultants. And in legal applications, defeat may depend on the legal hierarchy among statutes, on the court's level of authority, or on social or moral values. Our example in the introduction contains an argument on the criteria for defeat, viz. A's use of a priority rule (10) based on the expected consequences of certain events. This argument might, for instance, be attacked by an argument that in case of important officials even a small likelihood that the disease affects the official's functioning justifies publication, or by an argument that the negative consequences of publication for the official are small.

The dialectical status of arguments

The notion of defeat is a binary relation on the set of arguments. It is important to note that this relation does not yet tell us with what arguments a dispute can be won; it only tells us something about the relative strength of two individual conflicting arguments. The ultimate status of an argument depends on the interaction between all available arguments: it may very well be that argument B defeats argument A, but that B is itself defeated by a third argument C; in that case C 'reinstates' A (see Figure 3.6)[2]. Suppose, for instance, that the argument A that Tweety flies because it is a bird is regarded as being defeated by the argument B that Tweety does not fly because it is a penguin (for instance, because conflicting arguments are compared with respect to specificity). And suppose that B is in turn defeated by an argument C, attacking

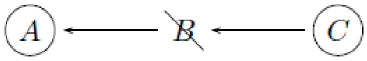


Figure 3.6: Argument C reinstates argument A.

[2] While in figures 3.4 and 3.5 the arrows stood for attack relations, from now on they will depict defeat relations.

B's intermediate conclusion that Tweety is a penguin. C might, for instance, say that the penguin observation was done with faulty instruments. In that case C reinstates argument A.

Therefore, what is also needed is a definition of the dialectical status of arguments on the basis of all the ways in which they interact. Besides reinstatement, this definition must also capture the principle that an argument cannot be justified unless all its subarguments are justified. There is a close relation between these two notions, since reinstatement often proceeds by indirect attack, i.e., attacking a subargument of the attacking argument (as illustrated by Figure 3.5). It is this definition of the status of arguments that produces the output of an argumentation system: it typically divides arguments in at least two classes: arguments with which a dispute can be 'won' and arguments with which a dispute should be 'lost'. Sometimes a third, intermediate category is also distinguished, of arguments that leave the dispute undecided. The terminology varies here also: terms that have been used are justified vs. defensible vs. defeated (or overruled), defeated vs. undefeated, in force vs. not in force, preferred vs. not preferred, etcetera. Unless indicated otherwise, we shall use the terms 'justified', 'defensible' and 'overruled' arguments.

These notions can be defined both in a 'declarative' and in a 'procedural' form. The declarative form, usually with fixed-point definitions, just declares certain sets of arguments as acceptable, (given a set of statements and evaluation criteria) without defining a procedure for testing whether an argument is a member of this set; the procedural form amounts to defining just such a procedure. Thus the declarative form of an argumentation system can be regarded as its (argumentation-theoretic) semantics, and the procedural form as its proof theory. Note that it is very well possible that, while an argumentation system has an argumentation-theoretic semantics, at the same time its underlying logic for constructing arguments has a model-theoretic semantics in the usual sense, for instance, the semantics of standard first-order logic, or a possible-worlds semantics of some modal logic.

EXERCISE 3.2.1 Reinstatement.

1. Extend Figure 3.6 (p. 35) with an argument D, such that D defeats C. Are there arguments that are justified? If so, which arguments? Are there arguments that are reinstated by D? If so, which?

2. Extend the figure just drawn with a fifth argument, E, such that E defeats D. Are there arguments that are justified? If so, which arguments? Are there arguments that are reinstated by D? If so, which? Are there arguments that are reinstated by E? If so, which?

The content of the remaining chapters on argumentation is as follows. Chapter 4 presents a fully abstract formal framework for the semantics of argumentation systems, which leaves the structure of arguments and the nature of the defeat relation unspecified. Chapter 5 discusses the proof-theory of these abstract argumentation systems in the form of so-called argument games. Chapter 6 then presents an instantiation of the abstract framework with structured arguments and two kinds of inference rules, deductive and defeasible ones. This framework is still partly abstract in that it abstracts from the nature and origin of these rules and from the nature of the logical language.

Chapter 4

A framework for abstract argumentation

This chapter presents a fully abstract framework for the semantics of argumentation, which leaves the internal structure of arguments and the nature of the defeat relation completely unspecified. As input it assumes nothing else but a set (of arguments) ordered by a binary relation (of defeat) and then defines several 'semantics', that is, properties that subsets of the set of all arguments should satisfy to be justified or defensible. Note that such argumentation semantics are, unlike the semantics of, say, standard first-order logic, not based on the notion of truth: since argumentation systems formalise reasoning that is defeasible, they are not concerned with truth of propositions, but with justification of accepting a proposition as true. In particular, one is justified in accepting a proposition as true if there is an argument for the proposition that one is justified in accepting. Argument-based semantics specify the conditions for when this is the case.

The abstract framework was introduced by Dung (1995). Historically, it came after the development of a number of more concrete argumentation systems, such as the systems of Pollock (1987)−(1994) and Vreeswijk (1993) (which are both predecessors of the framework to be discussed in Chapter 6). Nevertheless, Dung's article is by now widely regarded as seminal. It was a breakthrough in several ways. Firstly, it contains a general account of argumentation semantics, applicable to all systems that instantiate his framework. Secondly, it made a precise comparison possible between different systems by translating them into his abstract format. Third, it made a general study of formal properties of systems possible, which are inherited by all systems that instantiate his framework. Finally, all this applies not just to argumentation systems but also to other nonmonotonic logics, since Dung (1995) showed for several such logics how they can be translated into his abstract framework. In Section 4.6 we shall discuss his argument-based reconstruction of default logic.

4.1 The status of arguments: preliminary remarks

We now start the discussion of abstract argument-based semantics. As explained above, the task of argument-based semantics is to specify the conditions under which it is justified to accept an argument. These conditions assume an 'input' set of arguments, ordered by a binary relation of 'defeat'.[1] The framework is as abstract as possible, leaving both the structure of arguments and the grounds for defeat unspecified.

With Dung (1995) we shall call the input of the framework an 'abstract argumentation framework (sometimes 'argumentation framework' for short), abbreviated as AF.

Definition 4.1.1 [Abstract argumentation frameworks.]

1. An *abstract argumentation framework* (AF) is a pair $\langle Args, defeat \rangle$, where $Args$ is a set of arguments, and *defeat* a binary relation on $Args$.

2. We say that a set S of arguments defeats an argument A iff some argument in S defeats A; and S defeats a set S' of arguments iff it defeats a member of S'.

As for applications of the framework, one might think of the set $Args$ as all arguments that can be constructed in a given logic from a given set of premises (although this is not always the case: the framework equally applies to cases where just some of the constructible arguments are constructed). Unless stated otherwise, we shall below implicitly assume an arbitrary but fixed argumentation framework. Recall that we read 'A defeats B' in the weak sense of 'A conflicts with B and is not weaker than B'; so in some cases it may happen that A defeats B and B defeats A. If A defeats B, then if B does not defeat A we say that A *strictly defeats* B, otherwise A *weakly defeats* B.

Let us now concentrate on the task of defining the notion of a justified argument. Which properties should such a definition have? A simple definition is the following.

Definition 4.1.2 Arguments are either justified or not justified.

1. An argument is *justified* iff all arguments defeating it (if any) are not justified.

2. An argument is *not justified* iff it is defeated by an argument that is justified.

This definition works well in simple cases, in which it is clear which arguments should emerge victorious, as in the following example.

Example 4.1.3 Consider three arguments A, B and C such that B defeats A and C defeats B:

$$A \longleftarrow B \longleftarrow C$$

A concrete version of this example is

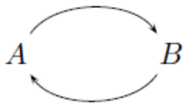
$A =$ 'Tweety flies because it is a bird'
$B =$ 'Tweety does not fly because it is a penguin'
$C =$ 'The observation that Tweety is a penguin is unreliable'

C is justified since it is not defeated by any other argument. This makes B not justified, since B is defeated by C. This in turn makes A justified: although A is defeated by B, A is reinstated by C, since C makes B not justified.

In other cases, however, Definition 4.1.2 is circular or ambiguous. In particular when arguments of equal strength interfere with each other, it is unclear which argument should remain undefeated.

[1] Dung (1995) uses the term 'attack', but to maintain uniformity throughout this text, we shall use 'defeat'.

Example 4.1.4 (Even cycle.) Consider the arguments A and B such that A defeats B and B defeats A.



A concrete example is

$A =$ 'Nixon was a pacifist because he was a quaker'
$B =$ 'Nixon was not a pacifist because he was a republican'

Can we regard A as justified? Yes, we can, if B is not justified. Can we regard B as not justified? Yes, we can, if A is justified. So, if we regard A as justified and B as not justified, Definition 4.1.2 is satisfied. However, it is obvious that by a symmetrical line of reasoning we can also regard B as justified and A as not justified. So there are two possible 'status assignments' to A and B that satisfy Definition 4.1.2: one in which A is justified at the expense of B, and one in which B is justified at the expense of A. Yet intuitively, we are not justified in accepting either of them.

In the literature, two approaches to the solution of this problem can be found. The first approach consists of changing Definition 4.1.2 in such a way that there is always precisely one possible way to assign a status to arguments, and which is such that with 'undecided conflicts' as in our example both of the conflicting arguments receive the status 'not justified'. The second approach instead regards the existence of multiple status assignments not as a problem but as a feature: it allows for multiple assignments and defines an argument as 'genuinely' justified if and only if it receives this status in all possible assignments. The following two sections discuss the details of both approaches.

First, however, another problem with Definition 4.1.2 must be explained, having to do with self-defeating arguments.

Example 4.1.5 (Self-defeat.) Consider an argument L, such that L defeats L (Figure 4.1). Suppose L is not justified. Then all arguments defeating L are not justified, so by clause 1 of Definition 4.1.2 L is justified. Contradiction. Suppose now L is justified. Then L is defeated by a justified argument, so by clause 2 of Definition 4.1.2 L is not justified. Contradiction.

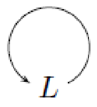


Figure 4.1: A self-defeating argument.

Thus, Definition 4.1.2 implies that there are no self-defeating arguments. Yet in ordinary discourse examples of self-defeating arguments can be found, as in the following example.

Example 4.1.6 (The Liar.) An elementary self-defeating argument can be fabricated on the basis of the so-called *paradox of the Liar*. There are many versions of this paradox. The one we use here, runs as follows:

> Dutch people can be divided into two classes: people who always tell the truth, and people who always lie. Hendrik is Dutch monk, and from Dutch monks we know that they tend to be consistent truth-tellers. Therefore, it is reasonable to assume that Hendrik is a consistent truth-teller. However, Hendrik *says* he is a liar. Is Hendrik a truth-teller or a liar?

The Liar-paradox is a paradox, because either answer leads to a contradiction.

1. Suppose that Hendrik tells the truth. Then what Hendrik says must be true. So, Hendrik is a lier. Contradiction.

2. Suppose that Hendrik lies. Then what Hendrik says must be false. So, Hendrik is not a lier. Because Dutch people are either consistent truth-tellers or consistent liers, it follows that Hendrik always tells the truth. Contradiction.

From this paradox, a self-defeating argument L can be made out of (1):

$$\dfrac{\dfrac{\text{Hendrik says: ``I lie''} \qquad \dfrac{\text{Dutch monks tend to be consistent truth-tellers} \qquad \text{Hendrik is a Dutch monk}}{\text{Hendrik is a consistent truth-teller}}}{\text{Hendrik lies}}}{\text{Hendrik is } \textbf{not} \text{ a consistent truth-teller}}$$

If the argument for "Hendrik is *not* a consistent truth-teller" is as strong as its subargument for "Hendrik is a consistent truth-teller," then L defeats one of its own subarguments, and thus is a self-defeating argument.

In conclusion, the treatment of self-defeating arguments deserves special attention. Below we shall discuss for each particular semantics how it deals with self-defeat.

4.2 The unique-status-assignment approach

We now discuss an approach that changes Definition 4.1.2 in such a way that there is always precisely one possible way to assign a status to arguments. This 'unique-status-assignment' approach can best be explained by the way it formalises 'reinstatement' (see above, Section 3.2). It does so by combining a notion of *acceptability* with a fixed-point operator. Recall that an argument that is defeated by another argument can only be justified if it is reinstated by a third argument, viz. by a justified argument that defeats its defeater. Part of this idea is captured by the notion of *acceptability* (which, by the way, is also relevant for the multiple-status-assignments approach, as we shall see below in Section 4.3).

Definition 4.2.1 [Acceptability.] An argument A is *acceptable* with respect to a set S of arguments iff each argument defeating A is defeated by S. When A is acceptable with respect to S, we also say that S *defends* A.

The arguments in S can be seen as the arguments capable of reinstating A in case A is defeated. To illustrate acceptability, consider again Example 4.1.3: A is acceptable with respect to $\{C\}$, $\{A, C\}$, $\{B, C\}$ and $\{A, B, C\}$, but not with respect to $\varnothing$ and $\{B\}$.

The notion of acceptability is not yet sufficient. Consider in Example 4.1.4 the set $S = \{A\}$. It is easy to see that A is acceptable with respect to S, since all arguments defeating A (viz. B) are defeated by an argument in S, viz. A itself. Clearly, we do not want that an argument can reinstate itself, and this is the reason why, to obtain a unique status assignment, a fixed-point operator must be used.

Intermezzo: fixed point operators Below we need some basics on fixed-point operators. Let S be a set and $O : Pow(S) \longrightarrow Pow(S)$ be an operator which for any subset of S returns a subset of S. $T \subseteq S$ is a *fixed point* of O iff $O(T) = T$. It is known that if O satisfies certain properties, it has a *least fixed point*, i.e. a fixed point which is a subset of all other fixed points of O. The most important of these properties is monotonicity, which is that $O(T) \subseteq O(T')$ whenever $T \subseteq T'$.

Consider now the following operator, which for each set of arguments returns the set of all arguments that are acceptable to it.

Definition 4.2.2 [Grounded semantics.] Let AF be an abstract argumentation framework, and let $S \subseteq Args_{AF}$. Then the operator F^{AF} is defined as follows:

- $F^{AF}(S) = \{A \in Args_{AF} \mid A \text{ is acceptable with respect to } S\}$

The *grounded extension* of AF is defined as the least fixed point of F^{AF}.

It can be shown that the operator F has a least fixed point, so that the notion of a grounded extension is well-defined[2]. (The basic idea is that if an argument is acceptable with respect to S, it is also acceptable with respect to any superset of S, so that F is monotonic.) Self-reinstatement can then be avoided by defining the set of justified arguments as that least fixed point. Note that in Example 4.1.4 the set $\{A\}$ and $\{B\}$ are fixed points of F but not its least fixed point, which is the empty set. In general we have that if no argument is undefeated, then $F(\varnothing) = \varnothing$.

These observations allow the following definition of a justified argument.[3]

Definition 4.2.3 [Justified arguments in grounded semantics.] An argument is *justified* with respect to grounded semantics iff it is a member of the grounded extension.

In applying these definitions, it is useful to know that the least fixed point of F can be approximated, and under certain conditions even obtained, by iterative application of F to the empty set.

Proposition 4.2.4 Dung (1995) Consider the following sequence of arguments.

[2] Below the superscript of F will usually be omitted.

[3] Henceforth, the definitions in this and the next chapter will, unless specified otherwise, impicitly assume an arbitrary but fixed abstract argumentation framework.

- $F^0 = \varnothing$
- $F^{i+1} = \{A \in Args \mid A$ is acceptable with respect to $F^i\}$.

Let $F^\omega = \cup_{i=0}^{\infty}(F^i)$. The following observations hold.

1. All arguments in F^ω are justified.
2. If each argument is defeated by at most a finite number of arguments, then an argument is justified iff it is in F^ω.

Proof: (1) follows from the facts that F^ω is included in the least fixed point of F and that if an argument is acceptable with respect to S, it is also acceptable with respect to any superset of S. For (2), assume that each argument has at most a finite number of defeaters. Let $S_0 \subseteq \ldots \subseteq S_n \subseteq \ldots$ be an increasing sequence of sets of arguments, and let $S = S_0 \cup \ldots S_n \cup \ldots$. Let $A \in F(S)$. Since there are only finitely many arguments which defeat A, there exists a number m such that $A \in F^m(S)$. Therefore, $F(S) = F(S_0) \cup \ldots F(S_n) \cup \ldots$ □

Note that if the condition of (2) does not hold, it is possible that $F^\omega \subset F(F^\omega)$.

In the iterative construction of the set of justified arguments first all arguments that are not defeated by any argument are added, and at each further application of F all arguments that are reinstated by arguments that are already in the set are added. This is achieved through the notion of acceptability. To see this, suppose we apply F for the ith time: then for any argument A, if all arguments that defeat A are themselves defeated by an argument in F^{i-1}, then A is in F^i.

It is instructive to see how this works in Example 4.1.3. We have that

$$F^1 = F(\varnothing) = \{C\}$$
$$F^2 = F(F^1) = \{A, C\}$$
$$F^3 = F(F^2) = F^2$$

The following example, with an infinite chain of defeat relations, provides another illustration.

Example 4.2.5 Consider an infinite chain of arguments $A_1, \ldots, A_n, \ldots$ such that A_1 is defeated by A_2, A_2 is defeated by A_3, and so on.

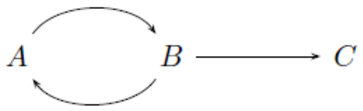
$$A_1 \longleftarrow A_2 \longleftarrow A_3 \longleftarrow A_4 \longleftarrow A_5 \longleftarrow \cdots$$

The least fixed point of this chain is empty, since no argument is undefeated. Consequently, $F(\varnothing) = \varnothing$. Note that this example has two other fixed points, which also satisfy Definition 4.1.2, viz. the set of all A_i where i is odd, and the set of all A_i where i is even.

Defensible arguments

Definition 4.2.3 allows a distinction between two types of arguments that are not justified. Consider first again Example 4.1.3 and observe that, although B defeats A, A is still justified since it is reinstated by C. Consider next the following extension of Example 4.1.4.

Example 4.2.6 (Zombie arguments.) Consider three arguments A, B and C such that A defeats B, B defeats A, and B defeats C.



A concrete example is

$A =$ ‘Dixon is no pacifist because he is a republican’
$B =$ ‘Dixon is a pacifist because he is a quaker, and he has no gun because he is a pacifist’
$C =$ ‘Dixon has a gun because he lives in Chicago’

According to Definition 4.2.3, neither of the three arguments are justified. For A and B this is since their relation is the same as in Example 4.1.4, and for C this is since it is defeated by B. Here a crucial distinction between the two examples becomes apparent: unlike in Example 4.1.3, B is, although not justified, not defeated by any justified argument and therefore B retains the potential to prevent C from becoming justified: there is no justified argument that reinstates C by defeating B. Sometimes arguments like B are called ‘zombie arguments’: B is not ‘alive’, (i.e., not justified) but it is not fully dead either; it has an intermediate status, in which it can still influence the status of other arguments.

We shall call the intermediate status of zombie arguments ‘defensible’. In the unique-status-assignment approach it can be defined as follows.

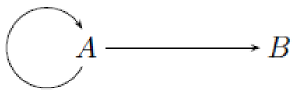
Definition 4.2.7 [Overruled and defensible arguments in grounded semantics.] With respect to grounded semantics, an argument is:

- *overruled* iff it is not justified, and defeated by a justified argument;
- *defensible* iff it is not justified and not overruled.

Self-defeating arguments

How does Definition 4.2.2 deal with self-defeating arguments? Consider the following extension of Example 4.1.5.

Example 4.2.8 Consider two arguments A and B such that A defeats A and A defeats B.

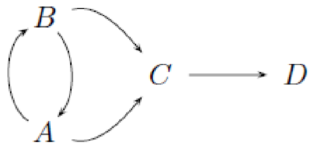


We have that $F(\varnothing) = \varnothing$, so neither A nor B are justified. Moreover, they are both defensible, since they are not defeated by any justified argument. At first sight, it might be thought that this is undesired since it would seem that self-defeating arguments should always be overruled. However, in Chapter 6 we will see that that things are more subtle and that a proper analysis of self-defeating arguments can only be given if the internal structure of arguments is made explicit.

Unique status assignments: problems

We have seen that the unique-assignment approach can be formalised in a mathematically elegant way, and that it produces intuitive results in many cases. However, there are also problems, in particular with examples of the following kind.

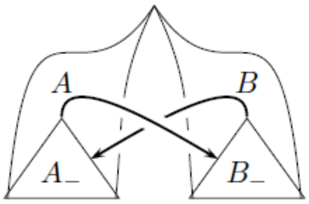
Example 4.2.9 (Floating arguments.) Consider the arguments A, B, C and D such that A defeats B, B defeats A, A defeats C, B defeats C and C defeats D.



Since no argument is undefeated, Definition 4.2.3 tells us that all of them are defensible. However, it might be argued that for C and D this should be otherwise: since C is defeated by both A and B, C should be overruled. The reason is that as far as the status of C is concerned, there is no need to resolve the conflict between A and B: the status of C 'floats' on that of A and B. And if C should be overruled, then D should be justified, since C is its only defeater.

A variant of this example is the following piece of default reasoning. To analyse this example, we must make two assumptions on the structure of arguments, viz. that they have a conclusion and that they have subarguments.

Example 4.2.10 (Floating conclusions.) Consider the arguments A^-, A, B^- and B such that A^- and B^- defeat each other and A and B have the same conclusion.



An intuitive reading is

A^- = Brigt Rykkje is Dutch because he was born in Holland
B^- = Brigt Rykkje is Norwegian because he has a Norwegian name
A = Brigt Rykkje likes ice skating because he is Dutch
B = Brigt Rykkje likes ice skating because he is Norwegian

The point is that whichever way the conflict between A^- and B^- is decided, we always end up with an argument for the conclusion that Brigt Rykkje likes ice skating, so it seems that it is justified to accept this conclusion as true, even though it is not supported by a justified argument. In other words, the status of this conclusion floats on the status of the arguments A^- and B^-.

While the unique-assignment approach is inherently unable to capture floating arguments and conclusions, there is a way to capture them, viz. by working with multiple status assignments. To this approach we now turn.

4.3 The multiple-status-assignments approach

A second way to deal with competing arguments of equal strength is to let them induce two alternative status assignments, in both of which one is justified at the expense of the other. In this approach, an argument is 'genuinely' justified iff it receives this status in all status assignments. This approach can be formalised in various ways, of which so-called stable and preferred semantics are the two best-known.

4.3.1 Stable semantics

The first way to allow for multiple status assignments, called stable semantics, is to take Definition 4.1.2 as the basis, and simply use the fact that it allows for multiple assignments. To this end, we turn this definition into one of a 'stable status assignment'.

Definition 4.3.1 [stable status assignments.]

Let $AF = \langle Args, defeat \rangle$ be an abstract argumentation framework and In and Out two subsets of *Args*. Then (In, Out) is a *stable status assignment* on the basis of AF iff $In \cap Out = \varnothing$ and $In \cup Out = Args$ and for all $A \in Args$ it holds that:

1. A is *in* (that is, $A \in In$) iff all arguments defeating A (if any) are *out*.

2. A is *out* (that is, $A \in Out$) iff A is defeated by an argument that is *in*.

Note that the conditions 1 and 2 are just the conditions of Definition 4.1.2.

Definition 4.3.1 is said to define *stable* status assignments for the following reasons. Firstly, with each stable status assignment a so-called *stable argument extension* can be associated, containing all the arguments that are *in* in the status assignment.

Definition 4.3.2 [Stable argument extensions.] A set of arguments is a *stable argument extension* iff for some stable status assignment it is the set of all arguments that are assigned the status *in*.

Now stable argument extensions coincide with what Dung (1995) calls *stable extensions*. In fact, Dung gives another but equivalent definition, which uses the notion of a conflict-free set of arguments.

Definition 4.3.3 [Conflict-free sets.] A set S of arguments is *conflict-free* iff no argument in S defeats an argument in S.

Then Dung defines stable extensions as follows.

Definition 4.3.4 [Stable extensions.] A set S of arguments is a *stable extension* iff S is conflict-free and every argument that is not in S, is defeated by S.

Proposition 4.3.5 The stable argument extensions induced by Definition 4.3.1 are precisely the stable extensions defined by Definition 4.3.4.

Proof: $\Rightarrow$:

Suppose (In, Out) is a stable status assignment. To be proven:

1. In is conflict-free.

   Assume for contradiction that In contains arguments A and B such that A defeats B. Then by condition (2) of Definition 4.3.1 B is in Out. But since $In \cap Out = \varnothing$, we have that B is not in In. Contradiction. So there are no such A and B, so In is conflict-free.

2. In defeats every argument outside In.

   Since stable status assignments assign a status to all arguments in $Args$ and $In \cap Out = \varnothing$, every argument outside In is in Out. Then by condition (2) of Definition 4.3.1 every such argument is defeated by an argument in In.

$\Leftarrow$:

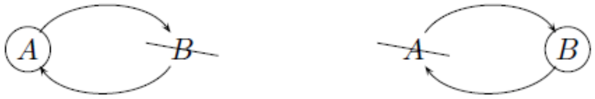
Suppose S is a stable extension. To be proven: $(S, Args/S)$ is a stable status assignment. Note first that by construction this is a partition of $Args$, so $In \cap Out = \varnothing$ and $In \cup Out = Args$. Then it must be verified that the two labelling conditions of Definition 4.3.1 are satisfied.

1. Condition (1) of Definition 4.3.1 is satisfied as follows. For the only if-part, if $A \in S$ then since S is conflict-free, no $B \in S$ defeats A, so all defeaters of A are in $Args/S$. For the if-part, if all defeaters of an argument A are in $Args/S$, then A cannot be in $Args/S$, since no defeater of A is in S. So A is in S.

2. Condition (2) of Definition 4.3.1 is satisfied as follows. For the only-if part, suppose $A \in Args/S$. Then since S defeats all arguments outside it, A is defeated by an argument in S. For the if-part, suppose A is defeated by an argument in S. Then since S is conflict-free, $A \in Args/S$. □

Below we shall use the term *stable extension* both for stable argument extensions and for Dung's stable extensions.

Example 4.1.3 has only one stable extension, viz. $\{A, C\}$, while Example 4.1.4 has two, induced by the following two status assignments:



Recall that an argumentation system is supposed to define when it is justified to accept an argument. What can we say in case of A and B in Example 4.1.4? Since both of them are *in* in one stable status assignment but *out* in the other, we must conclude that with respect to stable semantics neither of them is justified. This is captured by the following definition:

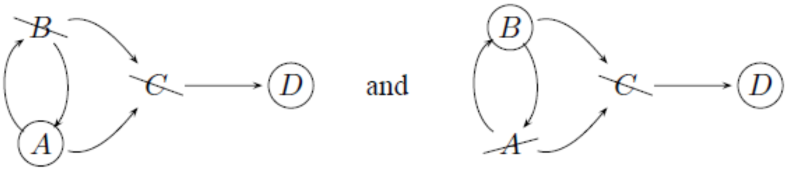
Definition 4.3.6 [Justified arguments in stable semantics.] With respect to stable semantics, an argument is *justified* iff it is *in* in all stable status assignments.

However, this is not all; just as in the unique-status-assignment approach, it is possible to distinguish between two different categories of arguments that are not justified. Some of those arguments are in no stable status assignment, but others are at least in some extensions. The first category can be called the *overruled*, and the latter category the *defensible* arguments.

Definition 4.3.7 [Overruled and defensible arguments in stable semantics.] With respect to stable semantics, an argument is:

- *overruled* iff it is *out* in all stable status assignments;
- *defensible* iff it is *in* in some but not in all stable status assignments.

It is easy to see that the unique-assignment and multiple-assignments approaches are not equivalent. Consider again Example 4.2.9. Argument A and B form an even defeat loop, thus, according to the multiple-assignments approach, either A and B can be assigned *in* but not both. So the above defeat relation induces stable two status assignments:



While in the unique-assignment approach all arguments are defensible, we now have that, while A and B are defensible, D is justified and C is overruled.

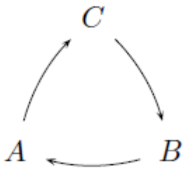
Multiple status assignments also make it possible to capture floating conclusions. Informally, this can be done by defining that a formula φ is justified as 'all extensions contain an argument for φ', rather than as 'there exists an argument for φ that is in all extensions'. In Chapter 6, in which the structure of arguments is formally defined, these alternative consequence notions for formulas will be fully formalised.

4.3.2 Preferred semantics

There is reason to discuss a second variant of the multiple-status-assignments approach. Since a stable extension is conflict-free, it reflects in some sense a coherent point of view. It is also a maximal point of view, in the sense that every possible argument is either accepted or rejected. In fact, stable semantics is the most 'aggressive' type of semantics, since a stable extension defeats every argument not belonging to it, whether or not that argument is hostile to the extension.

This feature is the reason why not all argumentation frameworks have stable extensions, as the following example shows. It contains an 'odd loop' of defeat relations.

Example 4.3.8 (Odd loop.) Let A, B and C be three arguments, represented in a triangle, such that A defeats C, B defeats A, and C defeats B.



In this situation, Definition 4.3.1 has some problems, since this example has no stable status assignments.

1. Assume that A is *in*. Then, since A defeats C, C is *out*. Since C is *out*, B is *in*, but then, since B defeats A, A is *out*. Contradiction.

2. Assume next that A is *out*. Then, since A is the only defeater of C, C is *in*. Then, since C defeats B, B is *out*. But then, since B is the only defeater of A, A is *in*. Contradiction.

Note that a self-defeating argument is a special case of Example 4.3.8, viz. the case where B and C are identical to A. This means that argumentation frameworks containing a self-defeating argument may have no stable status assignment.

To give such examples also a multiple-assignment semantics, we need allow for the possibility of *partial* status assignments.

Definition 4.3.9 [(Preferred) status assignments.] Let $AF = \langle Args, defeat \rangle$ be an abstract argumentation framework and In and Out two subsets of *Args*. Then (In, Out) is a *status assignment* on the basis of AF iff $In \cap Out = \varnothing$ and for all $A \in Args$ it holds that:

1. A is *in* (that is, $A \in In$) iff all arguments defeating A (if any) are *out*.

2. A is *out* (that is, $A \in Out$) iff A is defeated by an argument that is *in*.

A status assignment (In, Out) is *preferred* iff it maximises the set of argument labelled *in*, that is, if there exists no status assignment (In', Out') such that $In \subset In'$.

To go back to Example 4.3.8, preferred semantics gives it a unique preferred status assignment, viz. $(\varnothing, \varnothing)$.

The notions of justified, overruled and defensible arguments defined in Definitions 4.3.6 and 4.3.7 can be easily defined also for preferred semantics, by uniformly replacing ‘stable’ by ‘preferred’. However, in preferred semantics there are reasonable alternatives for the definitions of defensible and overruled arguments (and conclusions). This is because in each status assignment the status of an argument can be one of three kinds: *in*, *out* or undefined. Hence there are, unlike in stable semantics, situations where an argument is *in* in some but not in all assignments but yet not *out* in any assignment. Likewise, there are situations where an argument is *out* in some but not in all assignments but yet not *in* in any assignment. In the remainder of this reader we will for simplicity interpret the notions of defensible and overruled arguments as defined in Definitions 4.3.7.

To return to the notion of preferred extensions, Dung (1995) defines it not in terms of partial status assignments but with the notion of an admissible set, which in turn is defined in terms of acceptability.

Definition 4.3.10 [conflict-free and admissible sets.]

1. A set of arguments is *conflict-free* iff no argument in the set defeats an argument in the set.

2. A set of arguments S is *admissible* iff S is conflict-free and each argument in S is acceptable with respect to S.

Intuitively, an admissible set represents an admissible, or defendable, point of view. In Example 4.1.3 the sets $\varnothing$, $\{C\}$ and $\{A, C\}$ are admissible but all other subsets of $\{A, B, C\}$ are not admissible.

Definition 4.3.11 [Preferred extensions.] A conflict-free set of arguments is a *preferred extension* iff it is a maximal (with respect to set inclusion) admissible set.

There is a one-to-one correspondence between preferred status assignments and preferred extensions.

Proposition 4.3.12

1. If (In, Out) is a status assignment, then In is an admissible set;
2. Let $Out(E)$ be the set of all arguments defeated by E. If E is a preferred extension, then $(E, Out(E))$ is a status assignment;
3. (In, Out) is a preferred status assignment iff In is a preferred extension.

Proof: We first prove the following lemma (which is Lemma 10 of Dung 1995).

(*) If E is an admissible set and A is acceptable wrt E, then $\{A\} \cup E$ is admissible.

Proof of (*): It suffices to show that $\{A\} \cup E$ is conflict-free. Assume for contradiction the contrary. Then there is a $B \in E$ such that either A defeats B or B defeats A. Since E is admissible and A is acceptable wrt E, there is a B' in E such that B' defeats B or B' defeats A. Since E is conflict-free, it follows that B' defeats A. But then there is an argument B'' in E such that B'' defeats B'. Contradiction. □

Proof of (1):

Let (In, Out) be any status assignment and A be any member of In. Observe first that In is conflict-free. Next, all arguments defeating A are in Out, so all arguments defeating A are defeated by In. But then In is an admissible set.

Proof of (2):

Let E be any preferred extension. Condition 2 of Definition 4.3.9 is satisfied by definition of $Out(E)$. To verify condition 1, observe first that all members of E are acceptable with respect to E, so all their defeaters are in $Out(E)$. Next, let A be any argument such that all its defeaters are in $Out(E)$. Then A is acceptable with respect to E, and by (*), $\{A\} \cup E$ is admissible. But then, since E is maximally admissible, it follows that $A \in E$.

Proof of (3), $\Rightarrow$:

Consider any preferred status assignment (In, Out). By (1), In is admissible. To prove that In is maximally admissible, assume for contradiction that there is an admissible set $In' \supset In$. By a result of Dung (1995) we may without loss of generality assume that In' is maximally admissible. Then (In', Out') is a status assignment by (2). But since $In' \supset In$, (In, Out) is not a preferred status assignment. Contradiction.

Proof of (3), $\Leftarrow$:

Assume that E is a preferred extension. By (2), $(E, Out(E))$ is a status assignment. Next, to prove that E is a preferred status assignment, assume for contradiction otherwise, viz. that there is a status assignment (In, Out) such that $In \supset E$. By (1), In is an admissible set. But then E is not maximally admissible. Contradiction. □

It follows from Definition 4.3.11 that:

Proposition 4.3.13 (Dung, 1995) Every abstract argumentation framework has at least one preferred extension.

Proof: We begin by proving that every admissible set is contained in a maximal admissible set. From this the observation follows since the empty set is admissible.

Consider a sequence $S_0 \subseteq \ldots \subseteq S_i \subseteq \ldots$ of admissible sets. Clearly, $S = S_0 \cup \ldots \cup S_i \cup \ldots$ is maximal in this sequence.[4] We prove that S is also admissible by proving that the union of any two elements of S is admissible.

Consider any $S_i, S_j \in S$. Observe first that if $S_i \subseteq S_j$, then since S_j is conflict-free, S_i does not defeat S_j. Suppose next that S_j defeats S_i. Since S_i is admissible, S_i then also defeats S_j. Contradiction. So $S_i \cup S_j$ is conflict-free. Next, since S_i as well as S_j defeats each argument that defeats one of its members, the same holds for $S_i \cup S_j$, so that this set is admissible. □

Grounded status assignments It turns out that grounded semantics can also be formulated in terms of status assignments, namely, as those assignments that minimise the set of arguments that is labelled *in*.

Definition 4.3.14 [Grounded status assignments.] A status assignment $S = (In, Out)$ is *grounded* iff there is no status assignment $S' = (In', Out')$ such that $In' \subset In$.

Proposition 4.3.15 (Caminada, 2006) S is the grounded extension of AF if and only if (S, Out) is a grounded status assignment of AF.

Self-defeat in preferred semantics Finally, how does preferred semantics deal with self-defeating arguments? It turns out that, just as in grounded semantics, self-defeating arguments can prevent other arguments from being justified. This can be illustrated with Example 4.2.8 (two arguments A and B such that A defeats A and A defeats B). The set $\{B\}$ is not admissible, so the only preferred extension is the empty set. As said above, a full analysis of self-defeat requires that the internal structure of arguments is made explicit; this will be further discussed in Chapter 6, Section 6.6.

4.4 Formal relations between grounded, stable and preferred semantics

We now give some results on the relation between the various semantics proven by Dung (1995).

Proposition 4.4.1 Every stable extension is preferred, but not vice versa.

Proof: It is clear that each stable extension is a preferred extension. And Example 4.2.8 shows that the reverse does not hold: the empty set is a preferred extension of this argumentation framework, but it is not stable. □

The following results are listed without proofs.

1. The grounded extension is contained in the intersection of all preferred extensions (Example 4.2.9 is a counterexample against 'equal to').

[4] Strictly speaking, this follows from a result in lattice theory.

2. If an abstract argumentation framework does not give rise to infinite paths $A_1, \ldots, A_n, \ldots$ through the defeat graph such that each A_{i+1} defeats A_i then it has exactly one stable extension, which is also grounded and preferred. (Note that the even loop of Example 4.1.4 and the odd loop of Example 4.3.8 give rise to such an infinite defeat path.)

3. Finally, Dung (1995) identifies conditions under which preferred and stable semantics coincide. A necessary condition is that an abstract argumentation framework does not contain odd defeat loops.

4.5 Comparing the two approaches

How do the unique- and multiple-assignment approaches compare to each other? It is sometimes said that their difference reflects a difference between a 'skeptical' and 'credulous' attitude towards drawing defeasible conclusions: when faced with an unresolvable conflict between two arguments, a skeptic would refrain from drawing any conclusion, while a credulous reasoner would choose one conclusion at random (or both alternatively) and further explore its consequences. However, the distinction skeptical-credulous is independent of the distinction between the unique- and multiple-status-assignment approach. When deciding what to accept as a justified belief, what is important is not whether one or more possible status assignments are considered, but how the arguments are ultimately evaluated given these assignments. And this evaluation is captured by the qualifications 'justified' and 'defensible', which thus capture the distinction between 'skeptical' and 'credulous' reasoning. And since, as we have seen, the distinction justified vs. defensible arguments can be made in both the unique-assignment and the multiple-assignments approach, these approaches are independent of the distinction 'skeptical' vs. 'credulous' reasoning.

The use of skeptical reasoning (in whatever way it is formalised) is often defended by saying that since in an unresolvable conflict no argument is stronger than the other, neither of them can be accepted as justified, while the use of credulous reasoning has sometimes been defended by saying that the practical circumstances often require a person to act, whether or not s/he has conclusive reasons to decide which act to perform. In our opinion the notions of skeptical and credulous reasoning do not exclude but completement each other: whether it is better to reason skeptically or credulously may depend on the application context. For example, for a judge in a law court the reasoning about whether the suspect is guilty must clearly be skeptical, while for an intelligent software agent faced with two conflicting goals it makes sense to reason credulously, to achieve at least one of the goals.

As for their outcomes, the unique- and multiple-assignment approaches mainly differ in their treatment of floating arguments and conclusions. With respect to these examples, the question easily arises whether one approach is the right one. However, we prefer a different attitude: instead of speaking about the 'right' or 'wrong' definition, we prefer to speak of 'senses' in which an argument or conclusion can be justified. For instance, the sense in which the conclusion that Brigt Rykkje likes ice skating in Example 4.2.10 is justified is different from the sense in which, for instance, the conclusion that Tweety flies in Example 4.1.3 is justified: only in the second case is the conclusion supported by a justified argument. And the status of D in Example 4.2.9 is not quite the same as the status of, for instance, A in Example 4.1.3. Although both arguments need the help of other arguments to be justified, the argument helping A is itself justified, while the arguments helping D are merely defensible. Again it may depend on the application context which sense of justification is the best.

4.6 Argument-based reconstruction of other nonmonotonic logics

The application of Dung's abstract argumentation framework is not restricted to argument-based systems; it can also be used to reformulate other nonmonotonic logics in argument-based terms. The advantage of this is that these logics can thus be compared in terms of a general theory: it can be systematically investigated in which respects they differ, and what the consequences are of these differences. Moreover, it becomes easier to formulate alternative versions of these logics. For instance, it is very easy to switch from one type of semantics to another.

We shall illustrate this for one of the best-known nonmonotonic logics, default logic. Our reconstruction is based on the one of Dung (1995), but somewhat deviates from it: while Dung bases his reconstruction on Reiter's original version of default logic, we base it on Antoniou's (1999) reformulation in terms of processes.

One way to reconstruct default logic in argument-based terms is by defining an argument as a finite *process* in the sense of Antoniou (1999). Recall that (informally) a process is a sequence of defaults without multiple occurrences such that the prerequisite of each default is logically implied by the union of the 'hard' knowledge W and the consequents of all preceding defaults in the sequence. A process is *closed* iff no more defaults can be appended to the sequence, and it is *successful* iff each of its assumptions is consistent with what is derived during the process. Clearly, processes as arguments do not have to be closed, since arguments are typically constructed to prove a particular conclusion. Moreover, they do not have to be successful, since unsuccessful processes correspond to self-defeating arguments.

A default theory can now be interpreted as an abstract argumentation framework as follows.

Definition 4.6.1 For any default theory $\Delta = (W, D)$, the abstract argumentation framework $AF(\Delta) = \langle Args_\Delta, \mathit{defeat}_\Delta \rangle$ is defined as follows.

- $Args_\Delta = \{\Pi \mid \Pi \text{ is a finite process of } \Delta\}$;
- Π *defeats*$_\Delta$ Π' iff $\varphi \in In(\Pi)$ for some $\varphi \in Out(\Pi')$.

A formula φ is a *conclusion* of an argument Π iff $\varphi \in In(\Pi)$.

Thus an argument can be defeated by deriving the negation of one of its assumptions.

Under this translation of default logic into an argumentation system, a correspondence can be proven between default logic and stable semantics. More precisely, let Δ be a default theory, and

- for any set E of formulas, let $Args(E)$ be the set of all $\Pi \in Args_\Delta$ such that for all $k \in Out(\Pi) : \{\neg k\} \cup E$ is consistent,
- for any set $S \subseteq Args_\Delta$, let $Concs(S)$ be the union of all sets $In(\Pi_i)$ such that $\Pi_i \in S$.

Then the following holds:

Proposition 4.6.2 For any default theory Δ:

1. If S is a stable extension of $AF(\Delta)$, then $Concs(S)$ is a Reiter-extension of Δ;

2. If E is a Reiter-extension of Δ, then $Args(E)$ is a stable extension of $AF(\Delta)$.

The proof of this proposition uses some notation: if d is a default, then $Pre(d)$, $Jus(d)$ and $Cons(d)$ respectively denote d's prerequisite, justifications and consequent. We first prove the following lemma, which in effect says that violating the consistency check in testing applicability of a default gives rise to a defeating counterargument.

Lemma 4.6.3 If S is a stable extension of $AF(\Delta)$ and $\Pi \in S$, then:

1. all subsequences Π' of Π that are arguments are in S;

2. all arguments in S are processes.

Proof: For (1), observe that any defeater of Π' also is a defeater of Π, so is outside S; but then $\Pi' \in S$ by definition of a stable extension.

For (2), suppose $\Pi \in S$ and Π is not a process. Then for some subsequence $\Pi[i]$ of Π and $d_i \in \Pi$ the negation of some $j \in Jus(d_i)$ is in $In(\Pi[i])$. So $\Pi[i]$ defeats Π. Also, $\Pi[i] \in S$ by (1); but then S is not conflict-free. Contradiction. □

Proof: To prove (1) of Proposition 4.6.2, we first append all arguments in S into a sequence of defaults Π and delete each repeated occurrence of every default. Clearly, by Lemma 4.6.3 and conflict-freeness of S we have that Π is a process. We claim that Π is a closed and successful process.

Firstly, since S is conflict-free, it follows by definition of defeat that $In(\Pi) \cap Out(\Pi) = \varnothing$, so Π is successful.

Next, consider any default d not in Π and suppose that $Pre(d) \in In(\Pi)$. We claim that $In(\Pi) \vdash \neg k$ for some $k \in Jus(d)$. By compactness[5] of first-order logic, $Pre(d)$ is implied by some finite subset of $In(\Pi)$. With this subset a finite subprocess $\Pi[i]$ of Π can be associated. Since d is not an element of Π, we have that $\Pi[i], d$ is not a subprocess of Π. So by construction of Π we have that $\Pi[i], d \notin S$. But then since S is stable, S defeats $\Pi[i], d$ so $In(\Pi) \vdash \neg k$ for some $k \in Jus(d)$. Hence Π is closed.

Next, to prove (2), consider a closed process Π generating E and let $Args(\Pi)$ be the set of all finite processes that only use defaults from Π. Since Π is closed, we have that $Args(\Pi) = Args(E)$.

We next show that $Args(\Pi)$ is a stable extension. Conflict-freeness of $Args(\Pi)$ follows immediately from successfulness of Π. To show that $Args(\Pi)$ defeats any argument outside it, consider any such argument $A = d_1, \ldots, d_n$ and let d_i be the first default in A that is not in Π. Then since Π is closed, we have that $In(\Pi) \vdash \neg k$ for some $k \in Jus(d)$. But then by compactness of first-order logic, some argument in $Args(\Pi)$ defeats A. □

[5] Compactness means that if a sentence follows from an infinite set of premises, it also follows from a finite subset of these premises.

Example 4.6.4 Consider the following default theory $\Delta_1 = (W, D)$ where $W = \{p\}$ and

$$D = \left\{ d_1 : \frac{p : q \wedge r}{q}, \quad d_2 : \frac{q : s}{t}, \quad d_3 : \frac{p : u \wedge \neg t}{\neg t} \right\}$$

The argumentation framework $AF(\Delta_1)$ consists of the following arguments.

$A = \varnothing$
$B = d_1$
$C = d_1, d_2$
$D = d_3$
$E = d_1, d_3$
$F = d_3, d_1$
$G = d_1, d_3, d_2$
$H = d_3, d_1, d_2$

And the defeat relations are depicted in figure 4.2, except that the figure leaves implicit that G and H also defeat all other arguments.

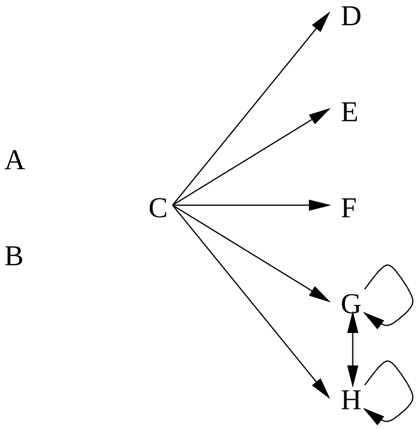


Figure 4.2: $AF(\Delta_1)$

It is easy to verify that the default theory Δ_1 has one default logic extension, viz. $Th(\{p, q, t\})$, generated by the process d_1, d_2. Correspondingly, $AF(\Delta_1)$ has a unique stable extension, viz. $\{A, B, C\}$. Note that this stable extension contains the process that generates the default logic extension of Δ_1, as well as all its subprocesses.

Example 4.6.5 Consider next a default theory $\Delta_2 = (\varnothing, \{\frac{:p}{\neg p}\})$. We know from Antoniou (1999) that this default theory has no extensions. We have that $AF(\Delta_2)$ contains two arguments, viz. $\varnothing$ and $\frac{:p}{\neg p}$. The only defeat relation is that the latter argument defeats itself. Then it is easy to see that this argumentation framework has no stable extensions.

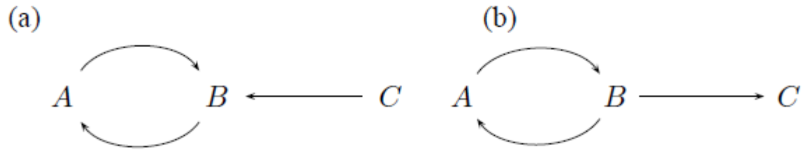
4.7 Final remarks

As remarked above, Dung's fully abstract approach was a major innovation in the study of defeasible argumentation, in that it provided an elegant general framework for investigating the various argumentation systems. Moreover, the framework also applies to other nonmonotonic logics, since Dung showed how several of these logics can be translated into argumentation systems. Thus it becomes very easy to formulate alternative semantics for nonmonotonic logics. For instance, default logic, which above was shown to have a stable semantics, can very easily be given an alternative semantics in which extensions are guaranteed to exist, like preferred or grounded semantics. Moreover, the proof theories that have been or will be developed for the various argument-based semantics immediately apply to the systems that are an instance of these semantics.

On the other hand, the fully abstract nature of Dung's framework also leaves much to the developers of particular systems. In particular, they have to define the internal structure of an argument, the ways in which arguments can conflict, and the origin of the defeat relation. In the next chapter a more concrete framework will be discussed in which these elements have been defined.

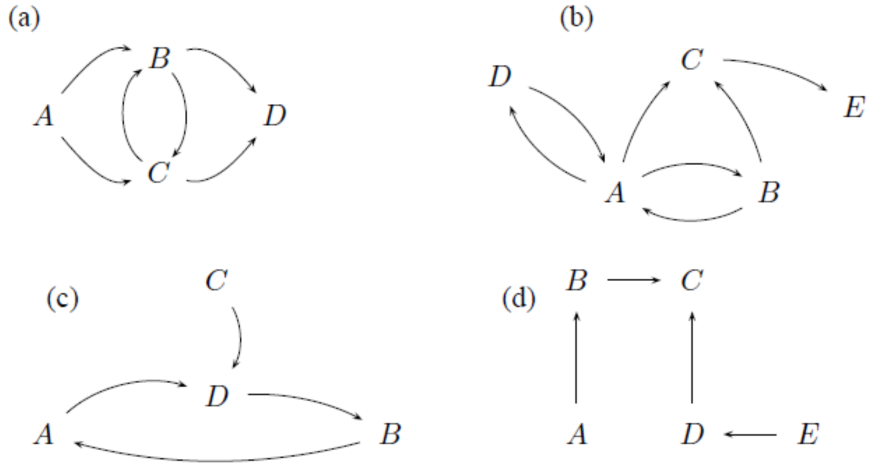
4.8 Exercises

EXERCISE 4.8.1 Determine, if possible, with Definition 4.1.2 which arguments are justified in the following two examples.



EXERCISE 4.8.2 Prove that if no argument of AF is undefeated, then $F^{AF}(\varnothing) = \varnothing$.

EXERCISE 4.8.3 Determine the grounded extension of the following defeat graphs. Show in each case its construction as in Proposition 4.2.4.



EXERCISE 4.8.4 Let

- $G(S) = \{A \in Args \mid A$ is not defeated by a member of $S\}$

1. Show that, for every set of arguments X, $F(X) = G^2(X)$ $[= G(G(X))]$.

2. Show that G is anti-monotonic. G is anti-monotonic if $A \subseteq B$ implies $G(B) \subseteq G(A)$.

3. Show on the basis of (2) that F is monotonic.

4. Let $\{G_i\}_{i \geq 0}$ be sets of arguments, such that

$$\begin{aligned} G_0 &=_{Def} \varnothing, \\ G_i &=_{Def} G(G_{i-1}). \end{aligned}$$

   Show that $G_0 \subseteq G_2 \subseteq G_4 \subseteq \ldots \subseteq G_5 \subseteq G_3 \subseteq G_1$.

EXERCISE 4.8.5 Determine for each of the defeat graphs in Exercise 4.8.3 which arguments are justified, which are defensible and which are overruled, all according to grounded semantics.

EXERCISE 4.8.6 Prove that S is a stable extension iff $S = \{A \mid A \text{ is not defeated by } S\}$.

EXERCISE 4.8.7 Determine all status assignments in Examples 4.1.3, 4.1.4 and 4.3.8. Which of these assignments are maximal?

EXERCISE 4.8.8 Consider two status assignments $S = (In, Out)$ and $S' = (In', Out')$ to the same argumentation framework such that $In \subset In'$.

1. Does it hold that $Out \subseteq Out'$? If so, give the proof; if not, give a counterexample.

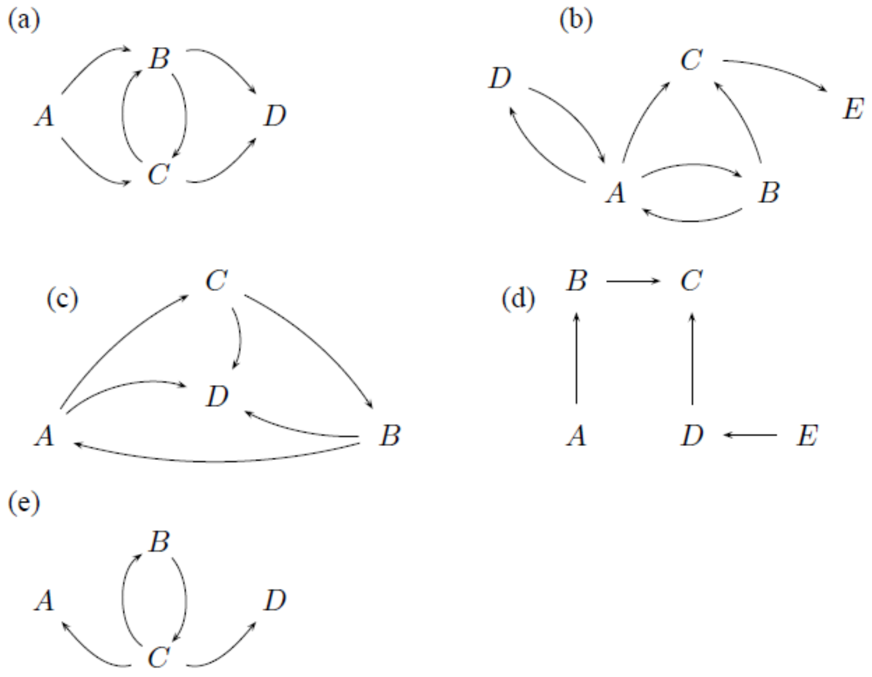
2. Does it hold that $Out \subset Out'$? Again, if so, give the proof; if not, give a counterexample.

EXERCISE 4.8.9 Give one or more alternative definitions of the notions of defensible and overruled arguments in preferred semantics. Verify for each definition whether it implies that each argument is either justified, or defensible, or overruled. If not, do you regard this as a flaw of your definition?

EXERCISE 4.8.10 Determine the admissible sets in Example 4.3.8. Which of these is or are maximally admissible?

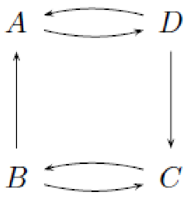
EXERCISE 4.8.11

1. Determine the preferred and stable extension(s) of the following defeat graphs.



2. Determine for each of the above defeat graphs, and with respect to each semantics, which arguments are justified, which are defensible and which are overruled.

EXERCISE 4.8.12 Consider four arguments A, B, C and D such that B strictly defeats A, D strictly defeats C, A and D defeat each other and B and C defeat each other.



Here is a natural-language version, in which the defeat relations are based on which argument uses the more specific of two conflicting defaults.

$A =$ Larry is rich because he is a public defender, public defenders are lawyers, and lawyers are rich;
$B =$ Larry is not rich because he is a public defender, and public defenders are not rich;
$C =$ Larry is rich because he lives in Hollywood, and people who live in Hollywood are rich;
$D =$ Larry is not rich because he rents in Hollywood, and people who rent in Hollywood are not rich.

1. Determine the grounded extension and the preferred extension(s) of this argumentation framework.

2. Determine in both cases which conclusions about Larry's richness are justified. Does the result agree with your intuitions?

EXERCISE 4.8.13 This exercise builds on Example 4.6.5. To see why preferred semantics can improve default logic, consider the default theory Δ_3 which is Δ_2 plus an extra default $\frac{:q}{q}$.

1. Determine the stable and preferred extensions of $AF(\Delta_3)$.
2. Explain why preferred semantics gives the better outcome.

EXERCISE 4.8.14

1. Consider a default theory $\Delta = (W, D)$ with

$$W = \varnothing$$

$$D = \left\{ \frac{:b}{a}, \frac{:e}{e}, \frac{a : c \wedge d}{c}, \frac{c : b}{b}, \frac{e : \neg a}{\neg d}, \frac{: \neg a}{\neg a} \right\}$$

and answer the following questions on the basis of the argumentation framework $AF(\Delta)$.

(a) Construct an argument for the conclusion b.
(b) Construct all minimal arguments that defeat the argument found under (a).
(c) Is the argument found under (a) element of an admissible set?
(d) Is it in a preferred extension?
(e) Is it in the grounded extension?

EXERCISE 4.8.15 Verify that any failed finite process is a selfdefeating argument.

Chapter 5

Games for abstract argumentation

So far mainly semantical aspects have been discussed, where the main focus was on characterising properties of *sets* of arguments, without specifying procedures for determining whether a given argument is a member of the set. In this chapter we shall go deeper into proof-theoretical, or procedural aspects of argumentation, where the chief concern is to investigate the status of *individual* arguments. This aspect of argumentation logics is less well-developed than its semantics; much research is ongoing or still to be carried out.

5.1 General ideas

The main question of this chapter is: given an argument from an abstract argumentation framework, how can its status be investigated? Several argumentation systems have tackled this problem in dialectical style. The common idea can be explained in terms of an argument game between two players, a proponent and an opponent of an argument. A dispute is an alternating series of moves by the two players. The proponent starts with an argument to be tested, and each following move consists of an argument that defeats (or in some cases strictly defeats) a move of the other party. The initial argument provably has a certain dialectical status if the proponent has a winning strategy, i.e., if he can win whatever moves the opponent makes.

The precise rules of the game depend on the semantics the game is meant to capture. A common winning criterion is that a player has won if s/he has made the other player run out of moves. However, other criteria are also possible. Other aspects on which choices have to be made are:

- Must moves strictly defeat their target or can they be weakly defeating?
- May moves be repeated?
- May players backtrack?
- May players defeat or be defeated by their own earlier moves?

These choices have to be made independently for both sides.

A natural idea in dialectical proof theories is that of dialectical asymmetry. The players of an argument game have different objectives: proponent wants to build a (dialectical) proof, while opponent wants to prevent proponent from doing so. In other words, while proponent is constructive, opponent is destructive, and this leads to different rules for the two players. Moreover, the burden induced by these rules will be heavier for one player than for the other. Which player has the heavier burden depends on whether the reasoning is credulous or skeptical: in skeptical reasoning the heavier burden is on proponent, while in credulous reasoning it is on opponent.

Let us now make these informal observations more precise. A dialectical proof theory takes the form of an argument game regulating a *dispute* between two *players*, the proponent P and opponent O of an argument. If p is a player, then $\overline{p}$ denotes the other player. The players *move* alternatingly, moving one argument at each turn. The game has a *protocol* function for determining *legality* of moves, by defining at each point in a dispute which arguments can be moved. Finally, a *winning criterion* is a partial function that determines the winner of a dispute, if any. If one player wins, the other player loses, so the argument game is a so-called zero-sum game.

These notions are formally defined as follows (recall that, unless stated otherwise, we implicitly assume an arbitrary but fixed argumentation framework).

Definition 5.1.1 [Moves, disputes and protocols.] Given an argumentation framework $AF = \langle Args, defeat \rangle$ we define the following notions.

- The set M of *moves* consists of all pairs (p, A) such that $p \in \{P, O\}$ and $A \in Args$; for any move (p, A) in M we denote p by $pl(m)$ and A by $s(m)$.
- The set of $M^{\leq\infty}$ of *disputes* is the set of all sequences from M and the set $M^{<\infty}$of *finite disputes* is the set of all finite sequences from M.
- A *protocol* is a function that specifies the *legal moves* at each stage of a dispute. Formally, protocol is a function Pr with domain a nonempty subset D of $M^{<\infty}$ taking subsets of M as values. That is:

  - $Pr : D \longrightarrow Pow(M)$

  such that $D \subseteq M^{<\infty}$. The elements of D are called the *legal finite disputes*. The elements of $Pr(d)$ are called the moves allowed after d. If d is a legal dispute and $Pr(d) = \varnothing$, then d is said to be a *terminated* dispute. Pr must satisfy the following conditions for all finite disputes d and moves m:

  1. $d \in D$ and $m \in Pr(d)$ iff $d, m \in D$;
  2. if $m \in Pr(d)$ then $pl(m) = P$ if d is of even length, otherwise $pl(m) = O$.

- A *winning function* is a partial function of type $W : D \longrightarrow \{P, O\}$.

The crucial elements of this definition are the protocol and the winning criterion. Dialectical proof theories differ only on these two elements.

We now define an abstract game-theoretic notion of defeasible provability, which is the same for all dialectical proof theories. It is defined in terms of the notion of a strategy. A strategy for a player in a dispute game has the form of a tree of disputes that for each possible move of the other player specifies a unique reply.

Definition 5.1.2 [Strategies.]

1. A *strategy* for player p is a tree of disputes only branching after p's moves, and containing all legal replies of $\overline{p}$.
2. A strategy for p is *winning* iff p wins all disputes in the strategy.

If the winning criterion is that the other player has no legal moves, then it is easy to see that a winning strategy for a player is a strategy in which all branches end with a move by that player.

Defeasible provability is now defined as follows, parametrised by a protocol X.

Definition 5.1.3 [Provability.] An argument A is *defeasibly provable in the X-game* iff the proponent has a winning strategy in a dispute with as root the argument A that satisfies protocol X.

5.2 Dialectics for grounded semantics

In this section we discuss a proof theory for determining whether an argument is in the grounded extension of a given argumentation framework. Since a grounded extension only contains justified arguments, the dialectical asymmetry favours the opponent: her moves are allowed to be simply defeating[1], while proponent's moves must be strictly defeating. Moreover, the proponent is not allowed to repeat his arguments. Finally, backtracking is not allowed for both players.

Definition 5.2.1 [Proof theory for grounded semantics.] A dispute satisfies the *G-game* protocol iff it satisfies the following conditions.

1. Moves are legal iff in addition to Definition 5.1.1 they satisfy the following conditions.
   (a) Proponent does not repeat his moves; and
   (b) Proponent's moves (except the first) strictly defeat opponent's last move; and
   (c) Opponent's moves defeat proponent's last move.
2. A player wins a dispute iff the other player has no legal moves.

A dispute satisfying the protocol of the G-game is called a G-dispute.

Example 5.2.2 Let A, B, C and D be arguments such that B and D defeat A, and C defeats B. Then a G-dispute on A may run as follows:

P: A, O: B, P: C

In this dispute P attempts to show A justified. Both B and D defeat A, which means that O has two choices in response to A. O chooses to respond with B in the second move. Then C is the only argument defeating B, so that P has no choice than to respond with C in the third move. There are no arguments against C, so that O cannot move and loses the dispute.

However, this outcome is not inevitable for O; her loss was merely caused by her weak play. A dispute in which O follows an optimal strategy is

P: A, O: D

[1] When below we say that move m defeats move m' we mean that $s(m)$ defeats $s(m')$.

And P has no reply, so O wins. Concluding, in this example P has no winning strategy. The only reason why P wins the first dispute is that O chooses the wrong argument, viz. B, in response to A. In fact, O is in the position to win every game, provided it chooses the right moves. In other words, O possesses a winning strategy.

Example 5.2.3 To give an another example, consider two strategies for P as depicted in Figure 5.1. The tree on the left is based on an argumentation framework AF_1 with $Args = \{A, B, C, D, E, F, G\}$ and *defeat* as shown by the arrows. Here P has a winning strategy, since in all disputes O eventually runs out of moves; so argument A is provable on the basis of AF_1. The tree on the right is based on an extension of AF_1 into AF_2 by adding H, I and J to $Args$ and adding new defeat relations corresponding to the new arrows (the extension is shown inside the dotted box). This is not a winning strategy for P, since one dispute ends with a move by O; so (assuming P has no better strategy) A is not provable on the basis of AF_2.

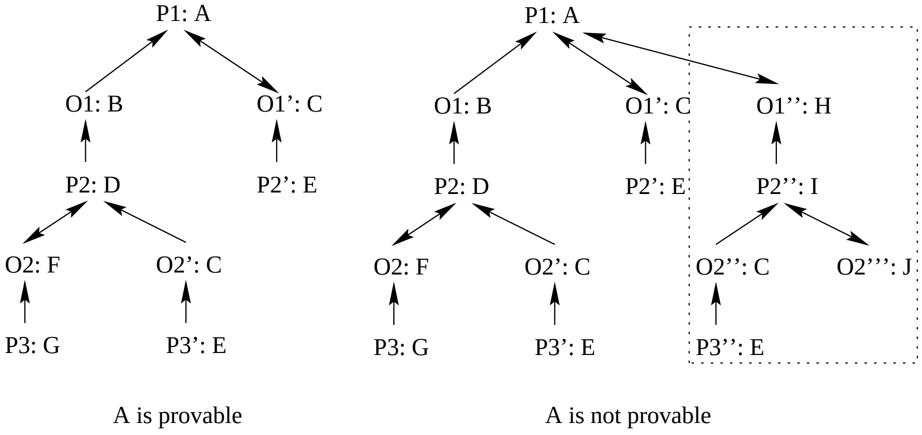


Figure 5.1: Two trees of proof-theoretical disputes.

Some words are in order on the non-repetition requirement of Definition 5.2.1 (condition 1a). This requirement does not change provability of any argument, since O will have a reply the second time iff she had a reply the first time. However, it avoids infinite disputes if $Args$ is finite, which is especially convenient for computational purposes. The same holds for the condition that P's arguments are strictly defeating; allowing them to be simply defeating does not change provability, but it avoids certain infinite disputes.

As for the relation between grounded semantics and its proof theory, the following proposition holds.

Proposition 5.2.4 [Soundness and completeness of the G-game.] An argument is in the grounded extension of an AF iff it is defeasibly provable on the basis of AF in the G-game.

Proof: (Sketch). We give a sketch of the proof for finitary AF's. Without this restriction the proof is more complicated. The restriction makes sense for computational purposes, since saying that an AF is finitary is equivalent to saying that each strategy based on AF has at most a finite number of branches.

$\Leftarrow$ (soundness):
Assume that P has a winning strategy W for A. Clearly, all of W's leaves A_n are in F^1, since they have no defeaters. But then in every branch of W, A_{n-2} is acceptable with respect to F^1 and so is in F^2. This can be repeated until the root of W is reached. □

$\Rightarrow$ (completeness):
Suppose A is in the grounded extension of AF. Then, since AF is finitary, there is a least number i such that $A \in F^i$. Then P has the following winning strategy if he begins a dispute with A. For each argument B defeating A moved by O, P can choose one argument C from F^{i-1} that strictly defeats B. This can be repeated for each argument defeating C, and so on, until P can choose an argument from F^1, which has no defeaters, so O has no legal reply. □

Note that completeness here does not imply semi-decidability (a logic is semi-decidable iff there exists an algorithm that can produce any provable formula): if the logic for constructing individual arguments is not decidable, then the search for counterarguments is in general not even semi-decidable, since this search is essentially a consistency check.

This completes the discussion of the dialectical proof theory for grounded semantics. We now turn to a dialectical proof theory for credulous reasoning, in particular for preferred semantics.

5.3 Dialectics for preferred semantics

In this section we present the so-called P- game[2], which serves as a credulous proof theory for preferred semantics, and was developed by Vreeswijk and Prakken (2000). For notational convenience we now denote defeat relations with $\leftarrow$. Throughout this section we will use the following example.

Example 5.3.1 The pair $\mathcal{A} = \langle X, \leftarrow \rangle$ with arguments

$$X = \{a, b, c, d, e, f, g, h, i, j, k, l, m, n, p, q\}$$

and $\leftarrow$ as indicated in Figure 5.2 is an example of an abstract argumentation framework. It accommodates a number of interesting cases, and will therefore be used as a running example throughout this chapter.

5.3.1 The basic ideas illustrated

Example 5.3.1 gives us some useful clues as to which features the argument game for preferred semantics should have. We are interested in credulous reasoning, so in testing membership of *some* extension. The argument game is based on the following idea. By definition, a preferred extension is a $\subseteq$-maximal admissible set. It is known that each admissible set is contained in a maximal admissible set (see the proof of Proposition 4.3.13), so the procedure comes down to trying to construct an admissible set 'around' the argument in question. If this succeeds, we know that the admissible set and hence the argument in question is contained in a preferred extension.

[2] The P in 'P-game should not be confused with the P denoting proponent.

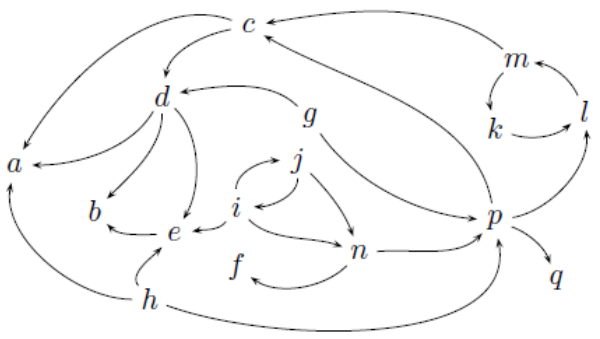
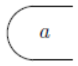


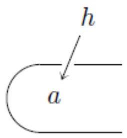
Figure 5.2: Defeat relations in the running example.

Suppose now we wish to investigate whether a is preferred, i.e., belongs to a preferred extension. We know that it suffices to show that the argument in question is admissible. The idea is to start with $S = \{a\}$ and, if a has defeaters, to find other arguments in order to complete S into an admissible set.

Example 5.3.2 (Straight failure). Consider the argument system of Figure 5.2, and suppose that P's task is to show that a is preferred. The first action of P is simply putting forward a:

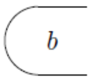


If a cannot be defeated, then $S = \{a\}$ is admissible, and P succeeds. However, since $a \leftarrow h$, O forwards h:

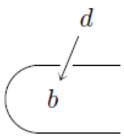


Now it is up to P to defend a by finding arguments against h. There are no such arguments, so that P fails to construct an admissible set 'around' a. So a is not admissible, hence not preferred.

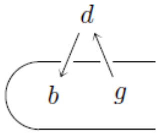
Example 5.3.3 (Straight success). Suppose that P wants to show that b is admissible. The first action of P is putting forward b:



O defeats b with d:

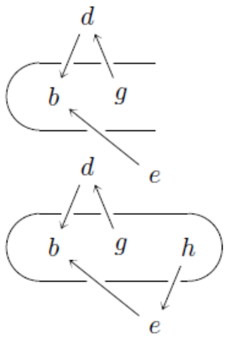


P defends this attack with g:



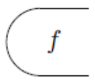
Since O 's attack on b with d has failed, O returns to b and defeats it again, this time with e:

P defends b again, this time with h. Since O is unable to find other arguments against b, g or h, P may now close S:

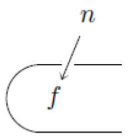


Example 5.3.4 (Even loop success). Suppose that P wants to show that f is admissible.

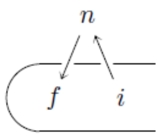
The first action of P is putting forward f:



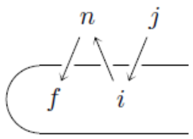
O defeats f with n:



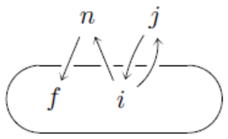
P defends this attack with i:



O defeats i with j:



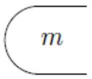
P defends i with i itself (so that i is self-defending). O is unable to put forward other arguments that defeat f or i so that P closes S:



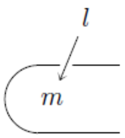
This example shows that P must be allowed to repeat his arguments, while O must be forbidden to repeat O 's arguments (at least in the same 'line of dispute'; see further below).

Example 5.3.5 (Odd loop failure). Suppose that P wants to show that m is admissible.

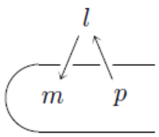
The first action of P is putting forward m:



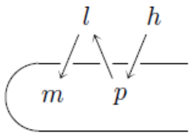
O defeats m with l:



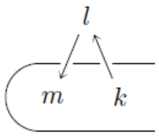
P defends this attack with p:



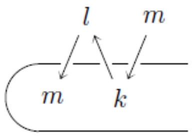
O defeats p with h:



P backtracks and removes p from S. He then tries to defend l with k instead:

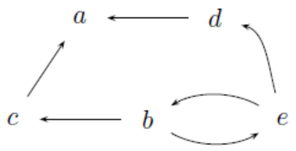


O defeats k with m (and, as a bonus, introduces an inconsistency in S):



P has no other arguments in response to l and m, so that he is unable to close S into an admissible set. So m is not contained in an admissible set. Note that we cannot allow P to reply to m with l, since otherwise the set that P is constructing 'around' m is not conflict-free, hence not admissible. So we must forbid P to repeat O 's moves. On the other hand, this example also shows that O should be allowed to repeat P 's moves, since such a repetition reveals a conflict in P 's position.

Example 5.3.6 (The need for backtracking). The next feature of our argument game is not illustrated by Figure 5.2 so we need a new example. Consider an argument system with five arguments a, b, c, d and e and defeat relations as shown in the graph.



This example shows that we must allow O to backtrack. Suppose P starts with a, O defeats a with d, and P defends a with e. If O now defeats e with b, P can defend e by repeating e itself. However, O can backtrack to a, this time defeating it with c, after which P can only defend a with b which repeats O, and in Example 5.3.5 we concluded that P must be forbidden to do so. So by backtracking O can reveal that P's position is not conflict-free.

Repetition

Let us summarise our observations about repetition of moves.

i. It makes sense for P to repeat himself (if possible), because O might fail to find or produce a new defeater of P's repeated argument. If so, then P's repetition closes a cycle of even length, of which P's arguments are admissible.

ii. It makes sense for O to repeat P (if possible), because thus she shows that P's collection of arguments is not conflict-free.

iii. P must not repeat O, because doing so would introduce a conflict into P's own collection of arguments.

iv. O must not repeat herself, because P has already shown to have adequate defense for O's previous arguments.

5.3.2 The P-game defined

We now turn to the formal definition of the argument game for preferred semantics. Let us fix some terminology.

- A *dispute line* is a dispute without backtracking moves.
- An *eo ipso* (meaning: "you said it yourself") is a move that uses a previous argument of the other player.

Definition 5.3.7 [A proof theory for preferred semantics.] A dispute satisfies the P-*game* protocol iff satisfies the following conditions.

1. Moves are legal iff in addition to Definition 5.1.1 they satisfy the following conditions.

   (a) A move by P responds to the previous move by O.

   (b) A move by O responds to some earlier move by P.

   (c) A move defeats the argument to which it responds.

   (d) P does not repeat O's moves.

   (e) O does not repeat O's moves in the same dispute line.

   (f) No two responses to the same move have the same content.

2. O wins a dispute iff she does an *eo ipso* or makes P run out of legal moves; otherwise P wins.

A dispute satisfying the rules of the P-game is called a P-dispute.

Note that an infinite dispute is won by P.

Since the P-game allows O to backtrack, during a P-dispute a tree of dispute lines is constructed. (By contrast, a G-dispute consists of only one dispute line, since in a G-dispute each argument replies to the immediately preceding move in the dispute.) Accordingly, there are two ways to display a P-dispute: as a *linear* structure, in the order in which the arguments are moved, and as a *tree* structure, where the edges indicate to which argument an argument replies. The reader should not confuse the tree form of a single dispute with the tree form of a strategy: in the latter tree (cf. Definition 5.1.2) an edge between two arguments indicates that the child argument is moved immediately after the parent argument; in other words, each branch of a strategy tree is a complete dispute, possibly with backtracking moves, but displayed in linear form.

Proposition 5.3.8 [Soundness and completeness of the P-game.] An argument is in some preferred extension of an AF iff it is defeasibly provable on the basis of AF in the P-game

Proof: (Below we say that an argument a is *defended* in a dispute iff the dispute begins with a and is won by P.) By definition of preferred extensions it suffices to show that an argument is admissible iff it can be defended in every dispute.

First suppose that a can be defended in every dispute. This includes disputes in which O has opposed optimally. Let us consider such a dispute. Let A be the arguments that P used to defend a. (in particular $a \in A$.) If A is not conflict-free then $a_i \leftarrow a_j$ for some $a_i, a_j \in A$, and O would have done an *eo ipso*, which is not the case. If A is not admissible, then $a_i \leftarrow b$ for some $a_i \in A$ while $b \not\leftarrow A$. In that case, O would have used b as a winning argument, which is also not the case. Hence A is admissible.

Conversely, suppose that $a \in A$ with A admissible. Now P can win every dispute by starting with a, and replying with arguments from A only. (P can do this, because all arguments in A are acceptable wrt A.) As long as P picks his arguments from A, O cannot win by *eo ipso*, because A is conflict-free. So a can be defended in dispute.□

Finally, a drawback of the P-game is that in some cases proofs have to be infinite. This is obvious when an argument has an infinite number of defeaters, but even otherwise some proofs are infinite, as in the case of Example 4.2.5. Nevertheless, it is easy to verify that with a finite set of arguments all proofs are finite.

5.4 A simplification of the P-game

Applying the P-game as defined above can be quite complex, since it combines two kinds of trees: the tree of reply relations within a single P- game and the game tree in the game-theoretical sense, that is, the tree of all possible ways in which a game about a given argument can be played. Fortunately, a simplification is possible, since Wu (2012) has proved that the proponent has a winning strategy in the P-game just in case there exists a terminated game won by the proponent. Here 'terminated' means that the player to move cannot move any further legal move. Note that infinite games can also be terminated in this sense. The intuition behind this result is that since the opponent can freely backtrack in a single game, a single terminated game will already contain all possible ways the opponent can attack the proponent's arguments.

5.5 Exercises

EXERCISE 5.5.1 Consider an argumentation framework with the arguments $\{A-G\}$ and the following defeat relations: A and B defeat each other, E and G defeat each other, C defeats B, D defeats A, E defeats D, and F defeats D.

1. Draw the defeat graph.

2. Determine all strategies for P and O in a game for A according to grounded semantics. Indicate which of these strategies are winning.

EXERCISE 5.5.2

1. Change Definition 5.2.1 to the effect that the non-repetition rule is dropped, and P's arguments are allowed to be simply defeating. Give a dispute that is finite under the original definition but infinite under the new definition.

2. Answer the same question for the case that only the non-repetition rule is dropped.

3. Give a dispute that is infinite under the original definition.

EXERCISE 5.5.3

1. Investigate for the following arguments in Exercise 4.8.3 whether they can be proven justified with respect to grounded semantics. For each provable argument, give a winning strategy for P. For each argument that is not provable, show why P's strategies fail.

   (a) In (a): investigate A, B and D.
   (b) In (b): investigate C and E.
   (c) In (c): investigate A, B and C.
   (d) In (d): investigate C.

2. Answer the same question about defeat graph (e) of Exercise 4.8.11, for the arguments C and D.

3. For each argument under 1 that is provable, compare the structure of P's winning strategy with the construction of the grounded extension that you found in Exercise 4.8.3. How are they related?

EXERCISE 5.5.4 This exercise is a continuation of Exercise 4.8.14. Investigate whether the argument for b that you constructed in that exercise, is defeasibly provable in the G-game. If so, give a winning strategy for P.

EXERCISE 5.5.5 Verify that a proof in the P-game of A_1 in Example 4.2.5 has to be infinite.

EXERCISE 5.5.6 Show with an example that the P-game is incorrect as a proof theory for stable semantics.

EXERCISE 5.5.7

1. Investigate for the following arguments in Exercise 4.8.11 whether they can be proven to be in some preferred extension. For each provable argument, give a winning strategy for P. For each argument that is not provable, show why P's strategies fail.

   (a) All arguments in (b);
   (b) All arguments in (c);
   (c) Argument c in (d).

2. Answer the same question for argument c in Figure 5.2.

Chapter 6

A framework for argumentation with structured arguments

6.1 Introduction

As explained above, Dung's (1995) abstract framework was an important advance in the formal study of argumentation. However, its fully abstract nature makes it less suitable for directly representing specific argumentation problems. It is best used as a tool for analysing particular argumentation formalisms and for developing a metatheory of such systems. When actual applications of argumentation-based inference have to be modelled, Dung's framework should be refined with accounts of the structure of arguments and the nature of the defeat relation. However, here too abstraction is still possible and worthwhile. This chapter instantiates Dung's abstract approach by assuming an unspecified logical language and by defining arguments as (directed acyclic) inference graphs formed by applying two kinds of inference rules, deductive (or 'strict') and defeasible rules'. As explained in Section 3.2, the notion of an argument as an inference graph naturally leads to three ways of attacking an argument: attacking a premise, attacking a conclusion and attacking an inference. To resolve such conflicts, preferences may be used, which leads to three corresponding kinds of defeat: undermining, rebutting and undercutting defeat. To characterise them, some minimal assumptions on the logical object language must be made, namely that certain well-formed formulas are a contrary or contradictory of certain other well-formed formulas. Apart from this the framework is still abstract: it applies to any set of inference rules, as long as it is divided into strict and defeasible ones, and to any logical language with a (possibly non-symmetric) negation connective.

The resulting framework unifies two ways to capture the fallibility of reasoning. Some, e.g. Bondarenko *et al.* (1997), locate the fallibility of arguments in the uncertainty of their premises, so that arguments can only be attacked on their premises. Others, e.g. Pollock (1994); Vreeswijk (1997), instead locate the fallibility of arguments in the riskiness of their inference rules: in these logics inference rules are of two kinds, being either deductive or defeasible, and arguments can only be attacked on their applications of defeasible inference rules. Vreeswijk (1993, Ch. 8) called these two approaches *plausible* and *defeasible* reasoning: he described plausible reasoning as sound (i.e, deductive) reasoning on an uncertain basis, and defeasible reasoning as unsound (but still rational) reasoning on a solid basis. In his chapter 8, Vreeswijk attempted to combine both forms of reasoning in a single formalism, but since then most formal accounts of argumentation have modelled either only plausible or only defeasible reasoning. The present framework again combines the two forms of reasoning but this time within the abstract setting of Dung (1995).

The account offered in this chapter further develops work undertaken in the European ASPIC project (Amgoud *et al.*, 2006; Caminada and Amgoud, 2007) and is more fully reported in (Prakken, 2010; Modgil and Prakken, 2013). It is based on work of John Pollock (1987; 1994) and Gerard Vreeswijk (1993; 1997) on the structure of arguments, work of Pollock (1974; 1987) on notions of defeat and work of Prakken and Sartor (1997) and others on argumentation with prioritised rules. The proofs of the formal results stated in this chapter can be found in (Prakken, 2010; Modgil and Prakken, 2013). The text of this chapter is largely based on Modgil and Prakken (2014), which gives a tutorial introduction to the $ASPIC^+$ framework.

6.2 Design choices and Overview

People argue to remove doubt about a claim (Walton, 2006, p. 1), by giving reasons why one should accept the claim and by defending these reasons against criticism. The strongest way to remove doubt is to show that the claim deductively follows from indisputable grounds. A mathematical proof from the axioms of arithmetic is like this: its grounds are mathematical axioms, while its inferences are deductively sound. So such a proof cannot be attacked in any way: not on its grounds and not on its inferences. However, such perfection is not attainable in real life: our grounds may not be indisputable or they may provide less than conclusive support for their claim.

Suppose we believe that John was in Holland Park some morning and that Holland Park is in London. Then we can deductively reason from these beliefs, to conclude that John was in London that morning. So the reasoning cannot be attacked. However, perfection remains unattainable since the argument is still fallible: its grounds may turn out to be wrong. For instance, Jan may tell us that he met John in Amsterdam that morning around the same time. We now have a reason against our belief that John was in Holland Park that morning, since witnesses usually speak the truth. Can we retain our belief or must we give it up? The answer to this question determines whether we can accept that John was in London that morning.

Maybe we originally believed that John was in Holland Park for a reason. Maybe we went jogging in Holland Park and we saw John. We then have a reason supporting our belief that John was in Holland Park that morning, since we know that our senses are usually accurate. But we cannot be sure, since Jan told us that he met John in Amsterdam that morning around the same time. Perhaps our senses betrayed us this morning? But then we hear that Jan has a reason to lie, since John is a suspect in a robbery in Holland Park that morning and Jan and John are friends. We then conclude that the basis for questioning our belief that John was in Holland Park that morning (namely, that witnesses usually speak the truth and Jan witnesses John in Amsterdam) does not apply to witnesses who have a reason to lie. So our reason in support of our belief is undefeated and we accept it.

If we want to formalise a logic for argumentation, then this simple example (displayed in Figure 6.1) already suggests a number of issues we have to deal with. At least two further important design decisions have to be made: how can arguments be built, i.e., how can claims be supported with grounds, and how can arguments be attacked? We shall see that the answers to these two questions are related.

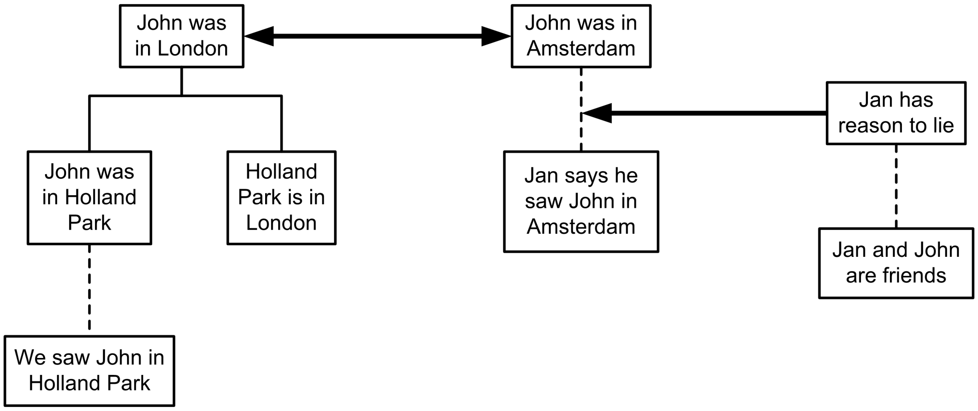


Figure 6.1: An informal example

First, the claims and beliefs in our example were supported in various ways: in the first case we appealed to the principles of deductive inference when concluding that John was in London (visualised in Figure 6.1 with solid links). *ASPIC*$^+$ is therefore designed so that arguments can be constructed using deductive or *strict* inference rules that license deductive inferences from premises to conclusions. However, in the other two cases the reasoning from grounds to claim appealed to the reliability of, respectively, our senses and witnesses as sources of information. Should these kinds of support (inferences) from grounds to claims be modelled as deductive?

To help answer this question, consider that our informal example contains three ways of attacking an argument: 1) Our initial argument that John was in London was attacked by the witness argument on its ground, or *premise*, that John was in Holland Park that morning; 2) We then modified our initial argument by extending it with an additional argument for the attacked premise, but the extended argument was still attacked (by the witness argument) on the (now) intermediate conclusion that John was in Holland Park that morning; 3) Finally, we counterattacked the witness argument not on a premise or conclusion but on the reasoning from the grounds to the claim: namely, the inference step from the premise that Jan said he met John in Amsterdam that morning to the claim that John was in Amsterdam that morning (note that here we regard the principle that witnesses usually speak the truth as an inference rule).

Now, returning to the question whether all kinds of inference should be deductive, the second type of attack would not be possible on the deductively inferred intermediate conclusion since the nature of deductive support is that it is absolutely watertight: if one accepts all antecedents of a deductively valid inference rule, then one must also accept its consequent no matter what, on the penalty of being irrational. If the antecedents of a deductively valid inference rule are true, then its consequent must also be true. So if we have reason to believe that the conclusion of a deductive inference is not true, then there must be something wrong with its premises (which may in turn be the conclusions of subarguments). It is for this very same reason that the third type of attack, on the deductive inferential step itself, is also not possible.

ASPIC$^+$ is therefore designed to comply with the common-sense and philosophically argued position (Pollock (1995, p.41); Pollock (2009, p. 173)) advocating the rationality of supporting claims with grounds that do not deductively entail them. In other words, the fallibility of an argument need not only be located in its premises, but can also be located in the inference steps from premises to conclusion (visualised in

Figure 6.1 with dashed links). Thus, arguments in *ASPIC*$^+$ can be constructed using *defeasible* inference rules, and arguments can be attacked on the application of such defeasible inference rules, in keeping with the interpretation that the premises of such a rule presumptively, rather than deductively, support their conclusions,

However, some would argue that the second and third type of attacks can be simulated using only deductive rules (specifically the deductive rules of classical logic) by augmenting the antecedents of these rules with normality premises. For example, with regard to the second type of attack, could we not say that our argument claiming that John was in Holland Park that morning since we saw him there has an implicit premise *our senses functioned normally*, and that the argument that John was in Amsterdam that morning in fact attacks this implicit premise, rather than its claim, thus reducing attacks on conclusions to attacks on premises? With regard to the third type of attack, could we not say that instead of attacking the defeasible inference step from Jan's testimony to the claim that John was in Amsterdam, we could model this step as deductive, and then add the premise that normally witnesses speak the truth, and then direct the attack at this premise? In other words, can we reduce attacks on inferences to attacks on premises?

In answer to these questions, we first note that some have argued that such deductive simulations are prone to yielding counterintuitive results. This is a topic that we will return to and examine in more detail in Section 6.4.5. Second, we claim that there is some merit in modelling the everyday practice of 'jumping to defeasible conclusions' and of considering arguments for contradictary conclusions. This is especially important given that one of the argumentation paradigm's key strengths is its characterisation of formal logical modes of reasoning in a way that corresponds with human modes of reasoning and debate.

The above discussion introduced the notion of *fallible* premises that can be attacked. However *ASPIC*$^+$ also wants to allow you to distinguish premises that are axiomatic and so cannot be attacked. We discuss the uses of such premises in Section 6.4, but for the moment we can summarise by saying that *ASPIC*$^+$ arguments can be constructed from fallible and infallible premises (respectively called *ordinary* and *axiom* premises in Section 6.3), and strict and defeasible inference rules, and that arguments can be attacked on their ordinary premises, the conclusions of defeasible inference rules, and the defeasible inference steps themselves. Finally, a key feature of the *ASPIC*$^+$ framework is that it accommodates the use of preferences over arguments, so that an attack from one argument to another only succeeds (as a defeat) if the attacked argument is not stronger than (strictly preferred to) the attacking argument, according to some given preference relation. The justified *ASPIC*$^+$ arguments are then evaluated with respect to the Dung framework relating *ASPIC*$^+$ arguments by the defeat relation.

6.3 The framework defined: Special case with 'ordinary' negation

In this section we present the basis definitions of the *ASPIC*$^+$ framework. Note that in this section we present a special case of *ASPIC*$^+$, in which conflict is based on the standard classical notion of negation, and then in Section 6.5 we replace negation by a more general notion of conflict between formulae.

6.3.1 Argumentation systems, knowledge bases, and arguments

To use *ASPIC*$^+$, you need to provide the following information. You must choose a *logical language* $\mathcal{L}$ closed under negation $\neg$ (which we later replace with a more general notion of conflict). You must then provide two (possibly empty) sets of *strict* ($\mathcal{R}_s$) and *defeasible* ($\mathcal{R}_d$) inference rules. If you provide a non-empty set of defeasible rules, you then need to also specify which well-formed formulas in $\mathcal{L}$ correspond to (i.e., name) which defeasible rule in $\mathcal{R}_d$. To do the latter requires specifying a partial function n from $\mathcal{R}_d$ to $\mathcal{L}$. These names can then by used when attacking arguments on defeasible inference steps. Informally, $n(r)$ is a wff in $\mathcal{L}$ which says that the defeasible rule $r \in \mathcal{R}$ is applicable, so that an argument claiming $\neg n(r)$ attacks the inference step in the corresponding rule[1].

The above is summarised in the following formal definition:

Definition 6.3.1 [**Argumentation systems**] An *argumentation system* is a triple $AS = (\mathcal{L}, \mathcal{R}, n)$ where:

- $\mathcal{L}$ is a nonempty logical language with a unary negation symbol $\neg$.
- $\mathcal{R} = \mathcal{R}_s \cup \mathcal{R}_d$ is a set of strict ($\mathcal{R}_s$) and defeasible ($\mathcal{R}_d$) inference rules of the form $\varphi_1, \ldots, \varphi_n \rightarrow \varphi$ and $\varphi_1, \ldots, \varphi_n \Rightarrow \varphi$ respectively (where φ_i, φ are meta-variables ranging over wff in $\mathcal{L}$), and $\mathcal{R}_s \cap \mathcal{R}_d = \varnothing$.
- n is a partial function such that $n : \mathcal{R}_d \longrightarrow \mathcal{L}$.

If there is no danger for confusion, we will sometimes write the sequence of antecedents of a strict or defeasible rule as a set. Furthermore, we write $\psi = -\varphi$ just in case $\psi = \neg\varphi$ or $\varphi = \neg\psi$ (we will sometimes informally say that formulas φ and $-\varphi$ are each other's negation). Note that $-$ is not part of the logical language $\mathcal{L}$ but a metalinguistic function symbol to obtain more concise definitions.

It is important to stress here that *ASPIC*$^+$'s strict and defeasible inference rules are *not* object-level formulae in the language $\mathcal{L}$, but are meta to the language, allowing one to deductively, respectively defeasibly, infer the rule's consequent from the rule's antecedents. Such inference rules may range over arbitrary formulae in the language, in which case they will, as usual in logic, be specified as *schemes*. For example, a scheme for strict inference rules capturing modus ponens for the material implication of classical logic can be written as $\alpha, \alpha \supset \beta \rightarrow \beta$[2], where α and β are metavariables for wff in $\mathcal{L}$. Alternatively, strict or defeasible inference rules may be domain-specific in that they reference specific formulae, as in the defeasible inference rule concluding that an individual flies if that individual is a bird: $Bird \Rightarrow Flies$. We will further discuss these distinct uses of inference rules in Section 6.4.

If you want to use *ASPIC*$^+$, then an argumentation system is not all you have to specify: you must also specify from which body of information the premises of an argument can be taken. We call this a knowledge base, and as discussed in Section 6.2, distinguish ordinary premises, which are uncertain and so can be attacked, and premises that are axioms, hence certain, and so cannot be attacked.

[1] n is a partial function since you may want to enforce that some defeasible inference steps cannot be attacked.

[2] In this chapter we use $\supset$ to denote the material implication connective of classical logic.

Definition 6.3.2 [**Knowledge bases**] A *knowledge base* in an $AS = (\mathcal{L}, \mathcal{R}, n)$ is a set $\mathcal{K} \subseteq \mathcal{L}$ consisting of two disjoint subsets $\mathcal{K}_n$ (the *axioms*) and $\mathcal{K}_p$ (the *ordinary premises*).

ASPIC$^+$ leaves you fully free to choose your language, what is an axiom and what is an ordinary premise and how you specify your strict and defeasible rules. However some care needs to be taken in making these choices, to ensure that the result of argumentation is guaranteed to be well-behaved. By 'well-behaved' we mean that the desirable properties proposed by Caminada and Amgoud (2007) are satisfied; for example, that the conclusions of arguments in the same extension are mutually consistent (we will define below what this means) and are closed under application of strict inference rules (whatever you can derive from your conclusions of arguments in an extension, with strict rules alone, is already a conclusion of an argument in that extension). In Section 6.4 we present some theorems which tell you how you can make your choices in such a way that the result is guaranteed to be well-behaved. These theorems will talk about two notions of consistency, namely, direct and indirect consistency. Indirect consistency is defined in terms of the closure of a set of well-formed formulas under application of strict inference rules. Informally, the strict closure of a set of wff is the set itself plus everything that can be derived from it when only applying strict rules.

Definition 6.3.3 [**Consistency and strict closure**] For any $S \subseteq \mathcal{L}$, let the closure of S under strict rules, denoted $Cl(S)$, be the smallest set containing S and the consequent of any strict rule in $\mathcal{R}_s$ whose antecedents are in $Cl(S)$. Then a set $S \subseteq \mathcal{L}$ is

- *directly consistent* iff $\nexists\ \psi, \varphi \in S$ such that $\psi = -\varphi$
- *indirectly consistent* iff $Cl(S)$ is directly consistent.

We call the combination of an argumentation system and a knowledge base an argumentation theory:

Definition 6.3.4 [**Argumentation theory**] An *argumentation theory* is a tuple AT = $(AS, \mathcal{K})$ where AS is an argumentation system and $\mathcal{K}$ is a knowledge base in AS.

ASPIC$^+$ arguments are now defined relative to an argumentation theory $AT = (AS, \mathcal{K})$, and chain applications of the inference rules from AS into directed acyclic inference graphs, starting with elements from the knowledge base $\mathcal{K}$ (if no premise is used more than once, then the graph will be a tree). In what follows, for a given argument, the function `Prem` returns all the formulas of $\mathcal{K}$ (called *premises*) used to build the argument, `Conc` returns its conclusion, `Sub` returns all its sub-arguments, `DefRules` returns all the defeasible rules of the argument and `TopRule` returns the last inference rule used in the argument.

Definition 6.3.5 [Argument] An *argument* A on the basis of an argumentation theory with a knowledge base $\mathcal{K}$ and an argumentation system $(\mathcal{L}, \mathcal{R}, n)$ is:

1. φ if $\varphi \in \mathcal{K}$ with: $\texttt{Prem}(A) = \{\varphi\}$, $\texttt{Conc}(A) = \varphi$, $\texttt{Sub}(A) = \{\varphi\}$, $\texttt{DefRules}(A)$ = $\varnothing$, $\texttt{TopRule}(A)$ = undefined.

2. $A_1, \dots A_n \rightarrow \psi$ if $A_1, \dots, A_n$ are arguments such that there exists a strict rule $\texttt{Conc}(A_1), \dots, \texttt{Conc}(A_n) \rightarrow \psi$ in $\mathcal{R}_s$.
   $\texttt{Prem}(A) = \texttt{Prem}(A_1) \cup \dots \cup \texttt{Prem}(A_n)$,
   $\texttt{Conc}(A) = \psi$,
   $\texttt{Sub}(A) = \texttt{Sub}(A_1) \cup \dots \cup \texttt{Sub}(A_n) \cup \{A\}$.
   $\texttt{DefRules}(A) = \texttt{DefRules}(A_1) \cup \dots \cup \texttt{DefRules}(A_n)$,
   $\texttt{TopRule}(A) = \texttt{Conc}(A_1), \dots \texttt{Conc}(A_n) \rightarrow \psi$

3. $A_1, \dots A_n \Rightarrow \psi$ if $A_1, \dots, A_n$ are arguments such that there exists a defeasible rule $\texttt{Conc}(A_1), \dots, \texttt{Conc}(A_n) \Rightarrow \psi$ in $\mathcal{R}_d$.
   $\texttt{Prem}(A) = \texttt{Prem}(A_1) \cup \dots \cup \texttt{Prem}(A_n)$,
   $\texttt{Conc}(A) = \psi$,
   $\texttt{Sub}(A) = \texttt{Sub}(A_1) \cup \dots \cup \texttt{Sub}(A_n) \cup \{A\}$,
   $\texttt{DefRules}(A) = \texttt{DefRules}(A_1) \cup \dots \cup \texttt{DefRules}(A_n) \cup \{\texttt{Conc}(A_1), \dots \texttt{Conc}(A_n) \Rightarrow \psi\}$,
   $\texttt{TopRule}(A) = \texttt{Conc}(A_1), \dots \texttt{Conc}(A_n) \Rightarrow \psi$.

For any argument A we define $\texttt{Prem}_n(A) = \texttt{Prem}(A) \cap \mathcal{K}_n$ and $\texttt{Prem}_p(A) = \texttt{Prem}(A) \cap \mathcal{K}_p$.

Example 6.3.6 Consider a knowledge base in an argumentation system with $\mathcal{L}$ consisting of $p, q, r, s, t, u, v, w, x, d_1, d_2, d_3, d_4, d_5, d_6$ and their negations, with $\mathcal{R}_s = \{s_1, s_2\}$ and $\mathcal{R}_d = \{d_1, d_2, d_3, d_4, d_5, d_6\}$, where

$$\begin{array}{llllll}
d_1: & p \Rightarrow q & d_4: & u \Rightarrow v & s_1: & p, q \rightarrow r \\
d_2: & s \Rightarrow t & d_5: & v, x \Rightarrow \neg t & s_2: & v \rightarrow \neg s \\
d_3: & t \Rightarrow \neg d_1 & d_6: & s \Rightarrow \neg p & &
\end{array}$$

Moreover, $\mathcal{K}_n = \{p\}$ and $\mathcal{K}_p = \{s, u, x\}$. Note that in presenting the example, we have informally used names d_i to refer to defeasible inference rules. We now define the n function that formally assigns wff d_i to such rules, i.e., for any rule informally referred to as d_i, we have that $n(d_i) = d_i$, so that '$n(d_1) = d_1$' is a shorthand for $n(p \Rightarrow q) = d_1$. In further examples we will often specify the n function in the same way.[3]

An argument for r (i.e., with conclusion r) is displayed in Figure 6.2, with the premises at the bottom and the conclusion at the top of the argument graph (which in this case is a tree). In this and the next figure, the type of a premise is indicated with a superscript and defeasible inferences, underminable premises and rebuttable conclusions are displayed with dotted lines. The figure also displays the formal structure of the argument. We have that

$$\begin{array}{llll}
\texttt{Prem}(A_3) = & \{p\} & \texttt{DefRules}(A_3) = & \{d_1\} \\
\texttt{Conc}(A_3) = & r & \texttt{TopRule}(A_3) = & s_1 \\
\texttt{Sub}(A_3) = & \{A_1, A_2, A_3\} & &
\end{array}$$

The distinction between two kinds of inference rules and two kinds of premises motivates a distinction into four kinds of arguments.

[3] In our further examples we will often leave the logical language $\mathcal{L}$ and the n function implicit, trusting that they will be obvious.

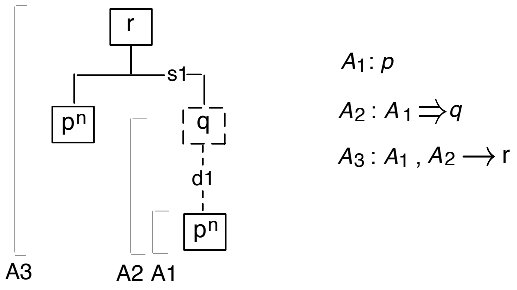


Figure 6.2: An argument

Definition 6.3.7 [Argument properties] An argument A is *strict* if $\texttt{DefRules}(A) = \emptyset$; *defeasible* if $\texttt{DefRules}(A) \neq \emptyset$; *firm* if $\texttt{Prem}(A) \subseteq \mathcal{K}_n$; *plausible* if $\texttt{Prem}(A) \cap \mathcal{K}_p \neq \emptyset$. We write $S \vdash \varphi$ if there exists a strict argument for φ with all premises taken from S, and $S \mid\!\sim \varphi$ if there exists a defeasible argument for φ with all premises taken from S.

Example 6.3.8 In Example 6.3.6 the argument A_1 is both strict and firm, while A_2 and A_3 are defeasible and firm. Furthermore, we have that $\mathcal{K} \vdash p$, $\mathcal{K} \mid\!\sim q$ and $\mathcal{K} \mid\!\sim r$.

In logic-based approaches to argumentation (see Section 6.4.4 below) arguments are often required to be minimal in that no proper subset of their premises should logically (according to the adopted base logic) imply the conclusion. In the *ASPIC*$^+$ context such a constraint would be fine for applications of strict rules. However, minimality cannot be required for application of defeasible inference rules, since defeasible rules that are based on more information may well make an argument stronger. For example, *Observations done in ideal circumstances are usually correct* is stronger than *Observations are usually correct.*

Another requirement of logic-based approaches, namely, that an argument's premises have to be consistent, can optionally be imposed in basic *ASPIC*$^+$, leading to two variants of the basic framework. We define a special class of arguments whose premises are indirectly consistent. In this way *ASPIC*$^+$ can be used as a framework for reconstructing logic-based argumentation formalisms, as we will further discuss in Section 6.4.4.

Definition 6.3.9 [consistent arguments] An argument A is *consistent* iff $\texttt{Prem}(A)$ is indirectly consistent.

6.3.2 Attack and defeat

Recall that *ASPIC*$^+$ is meant to generate Dung-style abstract argumentation frameworks, that is, a set of arguments with a binary relation of defeat. Having defined arguments above, we now define the attack relation and then, as discussed in Section 6.2, we apply preferences to determine the defeat relation (in fact Dung called his relation "attack" but we reserve this term for the basic notion of conflict, to which we then apply preferences).

Attack

We now first present the three ways in which arguments in *ASPIC*$^+$ can be in conflict, that is, three kinds of attack. In short, arguments can be attacked on a conclusion of a defeasible inference (rebutting attack), on a defeasible inference step itself (undercutting attack), or on an ordinary premise (undermining attack). As discussed in Section 6.2, that arguments cannot be attacked on their strict inferences goes without saying. We also discussed why arguments cannot be attacked on the conclusions of strict inferences: if the antecedents of a deductively valid inference rule are true, then its consequent must also be true no matter what. So if we have reason to believe that the conclusion of a deductive inference is not true, then there must be something wrong with the claims from which it is drawn. In Section 6.4.2 we will give a second reason why arguments cannot be attacked on conclusions of strict inferences. In short, this is because if we allow such attacks, then consistency and strict closure of conclusions cannot be guaranteed.

To define undercutting attack, the function n of an AS is used, which assigns to elements of $\mathcal{R}_d$ a well-formed formula in $\mathcal{L}$. Recall that informally, $n(r)$ (where $r \in R_d$) means that r is applicable. Then an argument using r is undercut by any argument with conclusion $-n(r)$.

Definition 6.3.10 [attacks] *A attacks B* iff *A undercuts*, *rebuts* or *undermines* B, where:

- *A undercuts* argument B (on B') iff $\texttt{Conc}(A) = -n(r)$ for some $B' \in \texttt{Sub}(B)$ such that B''s top rule r is defeasible.
- *A rebuts* argument B (on B') iff $\texttt{Conc}(A) = -\varphi$ for some $B' \in \texttt{Sub}(B)$ of the form $B''_1, \ldots, B''_n \Rightarrow \varphi$.
- Argument *A undermines* B (on φ) iff $\texttt{Conc}(A) = -\varphi$ for an ordinary premise φ of B.

This definition allows for a distinction between direct and indirect attack: an argument can be indirectly attacked by directly attacking one of its proper subarguments. This distinction will turn out to be crucial for a proper application of preferences to resolve attacks.

Example 6.3.11 In our running example argument A_3 cannot be undermined, since all its premises are axioms. A_3 can potentially be rebutted on A_2, with an argument for $\neg q$. However, the argumentaton theory of our example does not allow the construction of such a rebuttal. Likewise, A_3 can potentially be undercut on A_2, with an argument for $\neg d_1$. Our example does allow the construction of such an undercutter, namely:

B_1: s
B_2: $B_1 \Rightarrow t$
B_3: $B_2 \Rightarrow \neg d_1$

Argument B_3 has an ordinary premise s, so it can be undermined on B_1 with an argument for $\neg s$:

C_1: u
C_2: $C_1 \Rightarrow v$
C_3: $C_2 \rightarrow \neg s$

Note that since C_3 has a strict top rule, argument B_1 *does not* in turn rebut C_3.

Argument B_3 can potentially be rebut or undercut on either B_2 or B_3, since both of these subarguments of B_3 have a defeasible top rule. Our argumentation theory only allows for a rebutting attack on B_2:

C_1: u
C_2: $C_1 \Rightarrow v$
D_3: x
D_4: $C_2, D_3 \rightarrow \neg t$

All relevant arguments and attacks are displayed in Figure 6.3.

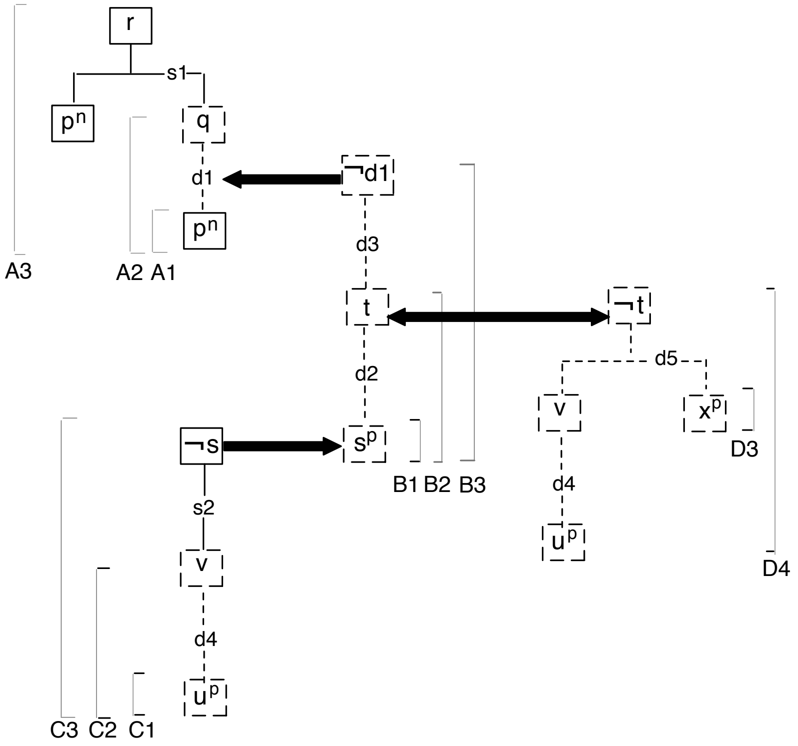


Figure 6.3: Attacks

Defeat

The attack relation tells us which arguments are in conflict with each other: if two arguments are in conflict then they cannot both be justified. However, Definition 4.2.1's notion of the acceptability of arguments is based on the notion that one argument can be used as a counter-argument to another. In general, an argument A can be used as a counter-argument to B, if A *successfully attacks*, i.e., defeats, B. Whether an attack from A to B (on its sub-argument B') succeeds as a defeat, may depend on the relative strength of A and B', i.e., whether B' is *strictly stronger than, or strictly preferred* to A. Note that only the success of undermining and rebutting attacks is contingent on preferences; undercutting attacks succeed as defeats independently of any preferences (see Modgil and Prakken (2013) for a discussion as to why this is the case).

Where do these preferences come from? Again, *ASPIC*$^+$ allows you to make any choice you like. All that *ASPIC*$^+$ as a framework wants is that you as a user give a binary ordering $\preceq$ on the set of all arguments that can be constructed on the basis of an argumentation theory. Then, as usual, if $A \preceq B$ and $B \npreceq A$ then B is strictly preferred to A (denoted $A \prec B$). Also, if $A \preceq B$ and $B \preceq A$ then $A \approx B$. We will later identify some conditions under which argument orderings are well-behaved in that they promote consistency and strict closure of conclusions. We will also define two example argument orderings that satisfy these conditions. However, for now all we need for defining *ASPIC*$^+$'s defeat relation is the attack relation and a preference ordering over arguments.

How should the preference ordering be applied to resolve attacks? At first sight, it would seem that *ASPIC*$^+$ can be taken to generate a so-called *preference-based argumentation framework* (PAF) in the sense of Amgoud and Cayrol (2002), that is, a triple consisting of the set of arguments, the attack relation and the argument ordering. That A defeats B could then be defined as A attacks B and $A \nprec B$. However, this does not work, for two reasons. First, PAFs do not recognise that undercutting attacks succeed irrespective of preferences. More seriously, PAFs cannot express how and at which points arguments attack each other, and yet this is crucial for a proper application of preferences to attack relations. Prakken (2012); Modgil and Prakken (2013) have shown that the use of PAFs leads to violation of the rationality postulates of subargument closure and consistency (see further Section 6.4.2 below) in cases where *ASPIC*$^+$ with the following definition satisfies these postulates.

Definition 6.3.12 [Successful rebuttal, undermining and defeat]

- *A successfully rebuts* B if A rebuts B on B' and $A \nprec B'$.
- *A successfully undermines* B if A undermines B on φ and $A \nprec \varphi$.
- *A defeats* B iff A undercuts or successfully rebuts or successfully undermines B.

The success of rebutting and undermining attacks thus involves comparing the conflicting arguments at the points where they conflict; that is, by comparing those arguments that are in a *direct* rebutting or undermining relation with each other. The definition of successful undermining exploits the fact that an argument premise is also a subargument.

Example 6.3.13 In our running example two argument orderings are relevant for whether attacks are successful: between B_1 and C_3 and and between B_2 and D_4. Note that the undercutting attack of B_3 on A_2 (and thereby on A_3) succeeds as a defeat irrespective of the argument ordering between B_3 and A_2. The undermining attack of C_3 on B_1 succeeds if $C_3 \nprec B_1$. If $B_2 \approx D_4$ or their relation is undefined then these two arguments defeat each other, while D_4 strictly defeats B_3. If $D_4 \prec B_2$ then B_2 strictly defeats D_4 while if $B_2 \prec D_4$ then D_4 strictly defeats both B_2 and B_3.

Let us now put all these elements together; that is the arguments and attacks defined on the basis of an argumentation theory, and a preference ordering over the arguments:

Definition 6.3.14 Let AT be an *argumentation theory* (AS, KB). A *(c-)structured argumentation framework ((*c-)SAF*)* defined by AT, is a triple $\langle \mathit{Args},\ \mathit{attack},\ \preceq\ \rangle$ where

- In a SAF, $Args$ is the smallest set of all finite arguments constructed from KB in AS satisfying Definition 6.3.5;
- In a c-SAF, $Args$ is the smallest set of all finite consistent arguments constructed from KB in AS satisfying Definition 6.3.5;
- $\preceq$ is a preference ordering on $Args$;
- $(X, Y) \in attack$ iff X attacks Y.

Example 6.3.15 In our running example $Args = \{A_1, A_2, A_3, B_1, B_2, B_3, C_1, C_2, C_3, D_3, D_4\}$, while $attack$ is such that B_3 attacks both A_2 and A_3, argument C_3 attacks all of B_1, B_2, B_3, argument D_4 attacks both B_2 and B_3 and, finally, B_2 attacks D_4.

6.3.3 Generating Dung-style abstract argumentation frameworks

We are now ready to instantiate a Dung framework with *ASPIC*$^+$ arguments and the *ASPIC*$^+$ defeat relation.

Definition 6.3.16 [**Argumentation frameworks**] An *abstract argumentation framework (AF) corresponding to a (c-)SAF* $= \langle Args,\ attack, \preceq \rangle$ is a pair $\langle Args,\ defeat \rangle$ such that $defeat$ is the defeat relation on $Args$ determined by $\langle Args,\ attack, \preceq \rangle$.

The justified arguments of the above defined AF are then defined under the various semantics of Chapter 4.

It is now also possible to define a consequence notion for well-formed formulas. Several definitions are possible. The following definition directly uses the notions of justified, defensible and overruled arguments from Chapter 4: (here an S-justified (S-defensible, S-overruled) argument is an argument that is justified (defensible, overruled) according to semantics S):

Definition 6.3.17 [The status of conclusions] For every semantics S and for every (c-)structured argumentation framework *(c-)SAF* with corresponding abstract argumentation framework AF, and every formula $\varphi \in \mathcal{L}_{AT}$:

1. φ is *S-justified* in *(c-)SAF* if and only if there exists an S-justified argument on the basis of AF with conclusion φ;
2. φ is *S-defensible* in *(c-)SAF* if and only if φ is not S-justified in SAF and there exists an S-defensible argument on the basis of AF with conclusion φ;
3. φ is *S-overruled* in *(c-)SAF* if and only if it is not S-justified or S-defensible in SAF and there exists an S-overruled argument on the basis of AF with conclusion φ.

Example 6.3.18 In our running example, if D_4 strictly defeats B_2, then we have a unique extension in all semantics which at least contains the set $S = \{A_1, A_2, A_3, C_1, C_2, C_3, D_3, D_4\}$. If in addition C_3 does not defeat B_1, then the extension also contains B_1. In both cases this yields that wff r is sceptically justified.

Alternatively, if B_2 strictly defeats D_4, then the status of r depends on whether C_3 defeats B_1. If it does, then we again have a unique extension in all semantics consisting of the set S, so r is sceptically justified. By contrast, if C_3 does not defeat B_1, we obtain a unique extension with A_1, B_1, B_2, B_3, C_1, C_2, C_3 and D_3, so r is neither sceptically nor credulously justified.

Finally, if B_2 and D_4 defeat each other, then the outcome again depends on whether C_3 defeats B_1. If it does, then the situation is as in the previous case – a unique extension S – but if C_3 does not defeat B_1, then the grounded extension consists of A_1, B_1, C_1-C_3, D_3. So in the latter case, in grounded semantics r is neither sceptically nor credulously justified. However, in preferred and stable semantics we then obtain two alternative extensions: the first contains D_4 while the second instead contains B_2 and B_3 and so excludes A_2 and A_3. So in the latter case r is credulously, but not sceptically justified under stable and preferred semantics.

Note that the first condition of Definition 6.3.17 is equivalent to

1. φ is *S-justified* in *(c-)SAF* if and only if there exists an argument with conclusion φ that is contained in all S-extensions of AF.

Thus this definition does not allow that different extensions contain different arguments for a skeptical conclusion and therefore does not capture floating conclusions (see Section 4.2). The following alternative definition does capture floating conclusions.

Definition 6.3.19 [Justified conclusions (possibly floating)]

1. φ is *S-f-justified* in *(c-)SAF* if and only if all S-extensions of AF contain an argument with conclusion φ.

6.3.4 More on argument orderings

A well studied use of preferences in the non-monotonic logic literature is based on the use of priority orderings over formulae in the language or defeasible inference rules. If *ASPIC*$^+$ is to be used as a framework for giving argumentation-based characterizations of non-monotonic formalisms augmented with priorities, then it needs to provide an account of how these priority orderings can be 'lifted' to preferences over arguments. Now the first thing to note is that if your use of *ASPIC*$^+$ involves using defeasible inference rules and ordinary premises, then both may come equipped with priority orderings $\leq$ on $\mathcal{R}_d$ and $\leq'$ on $\mathcal{K}_p$. We assume that these priority orderings are distinct to allow for the ontological nature of the rules and premises to be distinct. For example, the ordinary premises may represent the content of percepts from sensors or of witness testimonies, whose priority ordering reflects the relative reliability of the sensors, respectively witnesses. The defeasible rules may, for example, be prioritized based on probabilistic strength, on temporal precedence (defeasible rules acquired later are preferred to those acquired earlier), on the basis of principles of legal precedence, and so on. The challenge is to then define a preference over arguments A and B based on the priorities over their constituent ordinary premises *and* defeasible rules.

We now define two argument preference orderings, called the weakest-link and last-link orderings. These orderings are in turn based on priority orderings $\leq$ on $\mathcal{R}_d$ and $\leq'$ on $\mathcal{K}_p$, where as usual, $X <^{(')} Y$ iff $X \leq^{(')} Y$ and $Y \nleq^{(')} X$ (note that we may represent orderings in terms of the strict counterpart they define). However, these priorities relate individual defeasible rules, respectively ordinary premises, whereas when comparing two arguments, we want to compare them on the (possibly non-singleton) *sets of* rules/premises that these arguments are constructed from. So, to define these argument preferences, we need to first define an ordering over *sets* of rules/premises. We will denote this ordering with $\lhd_s$. For technical reasons we interpret it as strict preference; that is, $\Gamma \lhd_s \Gamma'$ means that Γ' is strictly preferred over Γ.

Note that for any sets of defeasible rules/ordinary premises Γ and Γ', we intuitively want that:
1) if Γ is the empty set, it cannot be that $\Gamma \lhd_{\mathtt{s}} \Gamma'$;
2) if Γ' is the empty set, it must be for any non-empty Γ that $\Gamma \lhd_{\mathtt{s}} \Gamma'$.
In other words, arguments that have no defeasible rules (ordinary premises) are, modulo the premises (rules), strictly stronger than (preferred to) arguments that have defeasible rules (ordinary premises). Hence the following definition explicitly imposes these constraints, and then gives two alternative ways of defining $\lhd_{\mathtt{s}}$; the so called `Elitist` and `Democratic` ways (i.e., $\mathtt{s} = \mathtt{Eli}$ and `Dem` respectively). `Eli` compares sets on their minimal and `Dem` on their maximal elements.

Definition 6.3.20 [Orderings $\lhd_{\mathtt{s}}$] Let Γ and Γ' be finite sets[4]. Then $\lhd_{\mathtt{s}}$ is defined as follows:

1. If $\Gamma = \varnothing$ then it cannot be that $\Gamma \lhd_{\mathtt{s}} \Gamma'$;
2. If $\Gamma' = \varnothing$ and $\Gamma \neq \varnothing$ then $\Gamma \lhd_{\mathtt{s}} \Gamma'$;
   else, assuming a preordering $\leq$ over the elements in $\Gamma \cup \Gamma'$, then if :
3. $\mathtt{s} = \mathtt{Eli}$:
   $\Gamma \lhd_{\mathtt{Eli}} \Gamma'$ if $\exists X \in \Gamma$ s.t. $\forall Y \in \Gamma'$, $X < Y$.
   else, if:
4. $\mathtt{s} = \mathtt{Dem}$:
   $\Gamma \lhd_{\mathtt{Dem}} \Gamma'$ if $\forall X \in \Gamma$, $\exists Y \in \Gamma'$, $X < Y$.

Henceforth, we will assume that $\lhd_{\mathtt{Eli}}$ is used to compare sets of rules/premises.

Now the **last-link principle** strictly prefers an argument A over another argument B if the last defeasible rules used in B are strictly less preferred ($\lhd_{\mathtt{s}}$) than the last defeasible rules in A or, in case both arguments are strict, if the premises of B are strictly less preferred than the premises of A. The concept of 'last defeasible rules' is defined as follows.

Definition 6.3.21 [Last defeasible rules] Let A be an argument.

- $\mathtt{LastDefRules}(A) = \varnothing$ iff $\mathtt{DefRules}(A) = \varnothing$.
- If $A = A_1, \ldots, A_n \Rightarrow \phi$, then $\mathtt{LastDefRules}(A) = \{\mathtt{Conc}(A_1), \ldots, \mathtt{Conc}(A_n) \Rightarrow \phi\}$, else $\mathtt{LastDefRules}(A) = \mathtt{LastDefRules}(A_1) \cup \ldots \cup \mathtt{LastDefRules}(A_n)$.

A simple example with more than one last defeasible rule is with $\mathcal{K} = \{p; q\}$, $\mathcal{R}_s = \{r, s \to t\}$ and $\mathcal{R}_d = \{p \Rightarrow r;\ q \Rightarrow s\}$. Then for the argument A for t we have that $\mathtt{LastDefRules}(A) = \{p \Rightarrow r;\ q \Rightarrow s\}$.

The above definition is now used to compare pairs of arguments as follows:

[4] Notice that it suffices to restrict $\lhd$ to finite sets since *ASPIC*$^+$ arguments are assumed to be finite (in Definition 6.3.14) and so their sets of ordinary premises/defeasible rules must be finite.

Definition 6.3.22 [Last link principle] Let A and B be two arguments. Then $A \prec^* B$ iff:

1. $\texttt{LastDefRules}(A) \lhd_{\texttt{s}} \texttt{LastDefRules}(B)$; or

2. $\texttt{LastDefRules}(A)$ and $\texttt{LastDefRules}(B)$ are empty and $\texttt{Prem}_p(A) \lhd_{\texttt{s}} \texttt{Prem}(_pB)$.

Moreover, $A \preceq B$ iff $A \prec B$ or $A = B$.

Because of this definition, the last-link ordering $\preceq$ is in fact a *strict partial ordering*, i.e., it is *transitive* (If $A \preceq B$ and $B \preceq C$ then $A \preceq C$) and *antisymmetric* (if $A \preceq B$ and $B \preceq A$ then $A = B$).

Example 6.3.23 Suppose in our running example that $u <' s$, $x <' s$, $d_2 < d_5$ and $d_4 < d_2$. Applying the last-link ordering, we must, to check whether C_3 defeats B_1, compare $\texttt{LastDefRules}(C_3) = \{d_4\}$ with $\texttt{LastDefRules}(B_1) = \varnothing$. Clearly, $\{d_4\} \lhd_{\texttt{Eli}} \varnothing$, so $C_3 \prec B_1$, so C_3 does not defeat B_1. Next, to check the conflict between B_2 and D_4 we compare $\texttt{LastDefRules}(B_2) = \{d_2\}$ with $\texttt{LastDefRules}(D_4) = \{d_5\}$. Since $d_2 < d_5$ we have that $\texttt{LastDefRules}(B_2) \lhd_{\texttt{Eli}} \texttt{LastDefRules}(D_4)$, so D_4 strictly defeats B_2.

The **weakest-link principle** considers not the last but all uncertain elements in an argument. Recall that in the following definition, $\texttt{Prem}_p(A) = \texttt{Prem}(A) \cap \mathcal{K}_p$.

Definition 6.3.24 [Weakest link principle] Let A and B be two arguments. Then $A \prec B$ iff

1. If both B and A are strict, then $\texttt{Prem}_{\mathrm{p}}(A) \lhd_{\texttt{s}} \texttt{Prem}_{\mathrm{p}}(B)$, else;

2. If both B and A are firm, then $\texttt{DefRules}(A) \lhd_{\texttt{s}} \texttt{DefRules}(B)$, else;

3. $\texttt{Prem}_{\mathrm{p}}(A) \lhd_{\texttt{s}} \texttt{Prem}_{\mathrm{p}}(B)$ and $\texttt{DefRules}(A) \lhd_{\texttt{s}} \texttt{DefRules}(B)$

Moreover, $A \preceq B$ iff $A \prec B$ or $A = B$.

Like the last-link ordering, the weakest-link ordering is also a strict partial ordering.

Example 6.3.25 If in our running example we apply the weakest-link ordering, then we must, to check whether C_3 defeats B_1, first compare $\texttt{Prem}_p(C_3) = \{u\}$ with $\texttt{Prem}_p(B_1) = \{s\}$. Since $u <' s$ we have that $\texttt{Prem}_p(C_3) \lhd_{\texttt{Eli}} \texttt{Prem}_p(B_1)$. Then we must compare $\texttt{DefRules}(C_3) = \{d_4\}$ with $\texttt{DefRules}(B_1) = \varnothing$. We have as above that $\{d_4\} \lhd_{\texttt{Eli}} \varnothing$. So $C_3 \prec B_1$ and so C_3 does not defeat B_1. Next, to check the conflict between B_2 and D_4 we must first compare $\texttt{Prem}_p(B_2) = \{s\}$ with $\texttt{Prem}_p(D_4) = \{u, x\}$. Since both $u <' s$ and $x <' s$ we have that $\texttt{Prem}_p(D_4) \lhd_{\texttt{Eli}} \texttt{Prem}_p(B_2)$. We must then compare $\texttt{DefRules}(B_2) = \{d_2\}$ with $\texttt{DefRules}(D_4) = \{d_4, d_5\}$. Since $d_4 < d_2$ we now have that $\texttt{DefRules}(D_4) \lhd_{\texttt{Eli}} \texttt{DefRules}(B_2)$. So $D_4 \prec B_2$ and B_2 strictly defeats D_4.

We next discuss with two examples when the last-, respectively, weakest-link ordering may be more suitable. Consider first the following example on whether people misbehaving in a university library may be denied access to the library.[5]

[5] In all examples below, sets that are not specified are assumed to be empty.

Example 6.3.26 Let $\mathcal{K}_p = \{Snores;\ Professor\}$, $\mathcal{R}_d =$

$$\begin{aligned}&\{Snores \Rightarrow_{d_1} Misbehaves;\\ &Misbehaves \Rightarrow_{d_2} AccessDenied;\\ &Professor \Rightarrow_{d_3} \neg AccessDenied\}.\end{aligned}$$

Assume that $Snores <' Professor$ and $d_1 < d_2,\ d_1 < d_3,\ d_3 < d_2$, and consider the following arguments.

A_1:	$Snores$	B_1:	$Professor$
A_2:	$A_1 \Rightarrow Misbehaves$	B_2:	$B_1 \Rightarrow \neg AccessDenied$
A_3:	$A_2 \Rightarrow AccessDenied$		

Let us apply the ordering to the arguments A_3 and B_2. The rule sets to be compared are $\texttt{LastDefRules}(A_3) = \{d_2\}$ and $\texttt{LastDefRules}(B_2) = \{d_3\}$. Since $d_3 < d_2$ we have that $\texttt{LastDefRules}(B_2) \lhd_{\texttt{Eli}} \texttt{LastDefRules}(A_3)$, hence $B_2 \prec A_3$. So A_3 *strictly* defeats B_2 (i.e., A_3 defeats B_2 but B_2 does not defeat A_3). We therefore have that A_3 is justified in any semantics, so we conclude $AccessDenied$.

With the weakest-link principle the ordering between A_3 and B_2 is different. Both A and B are plausible and defeasible so we are in case (3) of Definition 6.3.24. Since $Snores <' Professor$, we have that $\texttt{Prem}_p(A_3) \lhd_{\texttt{Eli}} \texttt{Prem}_p(B_2)$. Furthermore, the rule sets to be compared are now $\texttt{DefRules}(A_3) = \{d_1, d_2\}$ and $\texttt{DefRules}(B_2) = \{d_3\}$. Since $d_1 < d_3$ we have that $\texttt{DefRules}(A_3) \lhd_{\texttt{Eli}} \texttt{DefRules}(B_2)$. So now we have that $A_3 \prec B_2$. Hence B_2 now strictly defeats A_3 and we conclude instead that $\neg AccessDenied$.

Which outcome in this example is better? Some have argued that the last-link ordering gives the better outcome since the conflict really is between the two legal rules about whether someone may be denied access to the library, while d_1 just provides a sufficient condition for when a person can be said to misbehave. The existence of a conflict on whether someone may be denied access to the library is in no way relevant for the issue of whether a person misbehaves when snoring. More generally, it has been argued that for reasoning with legal (and other normative) rules the last-link ordering is appropriate.

However, an example can be given of exactly the same form but with the legal rules replaced by empirical generalisations, and in that case intuitions seem to favour the weakest-link ordering:

Example 6.3.27 Let $\mathcal{K}_p = \{BornInScotland;\ FitnessLover\}$, $\mathcal{R}_d =$

$$\begin{aligned}&\{BornInScotland \Rightarrow_{d_1} Scottish;\\ &Scottish \Rightarrow_{d_2} LikesWhisky;\\ &FitnessLover \Rightarrow_{d_3} \neg LikesWhisky\}.\end{aligned}$$

Assume that $BornInScotland <' FitnessLover$ and $d_1 < d_2,\ d_1 < d_3,\ d_3 < d_2$, and consider the following arguments.

A_1:	$BornInScotland$	B_1:	$FitnessLover$
A_2:	$A_1 \Rightarrow Scottish$	B_2:	$B_1 \Rightarrow \neg LikesWhisky$
A_3:	$A_2 \Rightarrow LikesWhisky$		

This time it seems reasonable to conclude $\neg LikesWhisky$, since the epistemic uncertainty of the premise and d_1 of A_3 should propagate to weaken A_3. And this is the outcome given by the weakest-link ordering. So it could be argued that for epistemic reasoning the weakest-link ordering is appropriate.

6.4 Ways to use the framework

As should be clear by now, *ASPIC*$^+$ is not a system but a framework for specifying systems. *ASPIC*$^+$ leaves you fully free to make choices as to the logical language, the strict and defeasible inference rules, the axioms and ordinary premises in your knowledge base, and the argument preference ordering. In this section we discuss various more or less principled ways to make your choices, and then show specific uses of *ASPIC*$^+$.

6.4.1 Choosing strict rules, axioms and defeasible rules

Domain specific strict inference rules

When designing your *ASPIC*$^+$ system, you can specify domain specific strict inference rules, as illustrated by the following example (based on Example 4 of Caminada and Amgoud 2007) in which the strict inference rules capture definitional knowledge, namely, that bachelors are not married.[6]

Example 6.4.1 Let $\mathcal{R}_d = \{d_1, d_2\}$ and $\mathcal{R}_s = \{s_1, s_2\}$, where:

$$\begin{array}{llcll} d_1 = & \mathit{WearsRing} \Rightarrow \mathit{Married} & \quad & s_1 = & \mathit{Married} \rightarrow \neg \mathit{Bachelor} \\ d_2 = & \mathit{PartyAnimal} \Rightarrow \mathit{Bachelor} & & s_2 = & \mathit{Bachelor} \rightarrow \neg \mathit{Married} \end{array}$$

Finally, let $\mathcal{K}_p = \{\mathit{WearsRing}, \mathit{PartyAnimal}\}$. Consider the following arguments.

$$\begin{array}{llcll} A_1: & \mathit{WearsRing} & \quad & B_1: & \mathit{PartyAnimal} \\ A_2: & A_1 \Rightarrow \mathit{Married} & & B_2: & B_1 \Rightarrow \mathit{Bachelor} \\ A_3: & A_2 \rightarrow \neg \mathit{Bachelor} & & B_3: & B_2 \rightarrow \neg \mathit{Married} \end{array}$$

We have that A_3 rebuts B_3 on its subargument B_2 while B_3 rebuts A_3 on its subargument A_2. Note that A_2 does not rebut B_3, since B_3 applies a strict rule; likewise for B_2 and A_3.

Notice that in the above example, the rules s_1 and s_2 are 'transpositions' of each other, and $\mathcal{R}_s$ is 'closed under transposition', in the following sense:

Definition 6.4.2 [Closure under transposition] A strict rule s is a *transposition* of φ_1, ..., $\varphi_n \rightarrow \psi$ iff s = $\varphi_1, \ldots, \varphi_{i-1}$, $-\psi$, $\varphi_{i+1}, \ldots, \varphi_n \rightarrow -\varphi_i$ for some $1 \leq i \leq n$.

An argumentation theory is *closed under transposition* iff for all rules r in $\mathcal{R}_s$ the transposition of r is also in R_s.

In general it is a good idea to ensure that your theory is closed under transposition. Proponents of this idea argue that this follows from the intuitive meaning of a strict rule as capturing deductive, that is, perfect inference: a strict rule $q \rightarrow \neg s$ expresses that if q is true, then this guarantees the truth of $\neg s$, no matter what. Hence, if we have s, then q cannot hold, otherwise we would have $\neg s$. In general, if the negation of the consequent of a strict rule holds, then we cannot have all its antecedents, since if we had all of them, then its consequent would hold. This is the very meaning of a strict rule. So it is very reasonable to include in $\mathcal{R}_s$ the transposition of a strict rule that is in $\mathcal{R}_s$.

[6] In the examples that follow we may use terms of the form s_i, d_i or f_i, to identify strict or defeasible inference rules or items from the knowledge base. We will assume that the d_i names are those assigned by the n function of Definition 6.3.1; sometimes we will attach these names to the $\Rightarrow$ symbol. Note that the s_i and f_i names have no formal meaning and are for ease of reference only.

A second reason for ensuring closure under transposition is that it ensures satisfaction of Caminada and Amgoud (2007)'s rationality postulates, as illustrated later in Section 6.4.2.

Strict inference rules and axioms based on deductive logics

Some find the use of domain-specific strict inference rules rather odd: why not instead express them as material implications in $\mathcal{L}$ and put them in the knowledge base as axiom premises? These people want to reserve the strict inference rules for general patterns of deductive inference, since they say that this is what inference rules are meant for in logic. (Below we will see that the same issue arises with regard to the choice of defeasible rules, but we ignore that issue for the moment). *ASPIC*$^+$ allows you to do this by basing your strict inference rules (and axioms) on a deductive logic of your choice. You can do so by choosing a semantics for your choice of $\mathcal{L}$ with an associated monotonic notion of semantic consequence, and then filling $\mathcal{R}_s$ with rules that are sound with respect to that semantics. For example, suppose you want it to conform to classical logic: you want to choose a standard propositional (or first-order) language, and you want that arguments can contain any classically valid inference step over this language. In *ASPIC*$^+$ you can achieve this in two ways, a crude way and a sophisticated way.

A crude way is to simply put all valid propositional (or first-order) inferences over your language of choice in $\mathcal{R}_s$. So if you have chosen a propositional language, then you define the content of $\mathcal{R}_s$ as follows. (where $\vdash_{PL}$ denotes standard propositional-logic consequence). For any finite $S \subseteq \mathcal{L}$ and any $\varphi \in \mathcal{L}$:[7]

$$S \rightarrow \varphi \in \mathcal{R}_s \text{ if and only if } S \vdash_{PL} \varphi$$

In fact, with this choice of $\mathcal{R}_s$, strict parts of an argument don't need to be more than one step long. For example, if rules $S \rightarrow \varphi$ and $\varphi \rightarrow \psi$ are in $\mathcal{R}_s$, then $S \cup \{\varphi\} \rightarrow \psi$ will also be in $\mathcal{R}_s$. Note also that using this method your strict rules will be closed under transposition, because of the properties of classical logic. The proof is easy: suppose $p \rightarrow q$ is in $\mathcal{R}_s$ for some p and q. Then we know that $p \vdash_{PL} q$, so (by the deduction theorem for classical logic) $\vdash_{PL} p \supset q$ so (by the properties of $\vdash_{PL}$) we have $\vdash_{PL} -q \supset -p$ so (by the other half of the deduction theorem) we have $-q \vdash_{PL} -p$, so (by choice of $\mathcal{R}_s$) $-q \rightarrow -p \in \mathcal{R}_s$.

Let us illustrate the crude approach with a variation on Example 6.4.1. We retain the defeasible rules d_1 and d_2 but we replace the domain-specific strict rules s_1 and s_2 with a single material implication $\mathit{Married} \supset \neg \mathit{Bachelor}$ in $\mathcal{K}_n$. Moreover, we put all propositionally valid inferences over our language in $\mathcal{R}_s$. Then the arguments change as follows:

A_1:	$\mathit{WearsRing}$	B_1:	$\mathit{PartyAnimal}$
A_2:	$A_1 \Rightarrow \mathit{Married}$	B_2:	$B_1 \Rightarrow \mathit{Bachelor}$
A_3:	$\mathit{Married} \supset \neg \mathit{Bachelor}$	B_3:	$\mathit{Married} \supset \neg \mathit{Bachelor}$
A_4:	$A_2, A_3 \rightarrow \neg \mathit{Bachelor}$	B_4:	$B_2, B_3 \rightarrow \neg \mathit{Married}$

Now A_4 rebuts B_4 on B_2 while B_4 rebuts A_4 on A_2.

A sophisticated way to base the strict part of *ASPIC*$^+$ on a deductive logic of your choice is to build an existing axiomatic system for your logic into *ASPIC*$^+$. You can

[7] Although antecedents of rules formally are sequences of formulas, we will sometimes abuse notation and write them as sets.

include its axiom(s) (typically a handfull) in $\mathcal{K}_n$ and its inference rule(s) (typically just one or a few) in $\mathcal{R}_s$. For example, there are axiomatic systems for classical logic with just four axioms and just one inference rule, namely, modus ponens (i.e, $\varphi \supset \psi, \varphi \rightarrow \psi)^8$. With this choice of $\mathcal{R}_s$ strict parts of an argument could be very long, since in logical axiomatic systems proofs of even trivial validities might be long. However, this difference with the crude way is not very big, since if we want to be crude, we must, to know whether $S \rightarrow \varphi$ is in $\mathcal{R}_s$, first construct a propositional proof of φ from S.

With the sophisticated way of building classical logic into our argumentation system, argument A_4 in our example stays the same, since modus ponens is in $\mathcal{R}_s$. However, argument B_4 will change, since modus tollens is not in $\mathcal{R}_s$. In fact, B_4 will be replaced by a sequence of strict rule applications, together being an axiomatic proof of $\neg Married$ from $Married \supset \neg Bachelor$ and $Bachelor$.

Which approach is more natural? We think that the crude way is more like how people reason: people often summarise chunks of deductive reasoning in one step. But if you want to implement such reasoning on a computer, then the crude and sophisticated way do not differ much.

However, note that in the sophisticated method, closure under transposition may not hold; our example above does not contain modus tollens (that is, $\varphi \supset \psi, -\psi \rightarrow -\varphi$). But we have already argued that the contrapositive reasoning yielded by the inclusion of transpositions is a desirable feature. Is this a problem for this method? No, since this reasoning can also be enforced without explicitly requiring transpositions of rules. Recall that $S \vdash \varphi$ was defined as 'there exists a strict argument for φ with all premises taken from S'. Now it turns out that if $\vdash$ contraposes, then this is just as good as closure of the strict rules under transposition. Contraposition of $\vdash$ means that if $S \vdash \varphi$, then if we replace one element s of S with $-\varphi$, then $-s$ is strictly implied:

Definition 6.4.3 [Closure under contraposition] An argumentation theory is *closed under contraposition* iff for all $S \subseteq \mathcal{L}$, $s \in S$ and ϕ, if $S \vdash \phi$, then $S \backslash \{s\} \cup \{-\phi\} \vdash -s$.

Now the point is that if $\vdash$ corresponds to classical provability (as we have made it by our choice of axioms and inference rules), then $\vdash$ does indeed contrapose. Again, as will be discussed in Section 6.4.2, closure under contraposition also ensures satisfaction of rationality postulates.

We end this section by stating a quite general result on a class of logics that, if embedded in *ASPIC*$^+$, ensures closure of the strict rules under contraposition. In Amgoud and Besnard (2009) the idea was introduced to base argumentation logics on so-called Tarskian abstract logics. Very briefly, abstract logics assume just some unspecified logical language $\mathcal{L}$ and a consequence operator over this language, which to each subset of $\mathcal{L}$ assigns a subset of $\mathcal{L}$ (its logical consequences). Tarski then assumed a number of constraints on Cn, which we need not repeat here. Finally, Tarski defined a set $S \subseteq \mathcal{L}$ as *consistent* iff $Cn(S) \neq \mathcal{L}$.

Now Amgoud and Besnard (2009)'s idea was to define an argument as a pair (S, p) where $S \subseteq \mathcal{L}$ and $p \in \mathcal{L}$, where S is consistent, $p \in Cn(S)$ and S is minimal in satisfying all these conditions. In *ASPIC*$^+$ Tarski's notion of an abstract logic can be used to generate the strict rules, via the following constraint (for any finite S):

$$S \rightarrow p \in \mathcal{R}_s \text{ iff } p \in Cn(S)$$

[8] As explained above, this strictly speaking is not a rule but a scheme or rules, with meta variables ranging over $\mathcal{L}$.

It turns out that any AT with this choice of strict rules satisfies closure under contraposition. Strictly speaking, this only holds under some assumptions on the relation between the Cn function and *ASPIC*$^+$'s negation (note that Tarski did not make any assumption on the syntax of $\mathcal{L}$), but these assumptions are quite natural. For the details we refer the reader to Section 5.2 of Modgil and Prakken (2013).

Choosing defeasible inference rules

Let us return to the question of how to choose the defeasible rules. Can we derive them from a logic of our choice just as we can derive the strict rules from a logic of our choice if we want to? This is controversial. Some philosophers argue that all rule-like structures that we use in daily life are "inference licences" and so cannot be expressed in the logical object language. In this view, all that can be done is apply them to formulas from $\mathcal{L}$ to support new formulas from $\mathcal{L}$. That is, these philosophers see all defeasible generalisations as inference rules, whether they are domain-specific or not.

Others (usually logicians) take a more standard-logic approach (e.g. Kraus *et al.* (1990); Pearl (1992)). They say that all contingent knowledge should be expressed in the object language, so they reject the idea of domain-specific defeasible inference rules (for the same reason they don't like domain-specific strict rules). They would introduce a new connective into $\mathcal{L}$, let us write it as $\rightsquigarrow$, where they informally read $p \rightsquigarrow q$ as something like "If p then normally/typically/usually q". They then want to give a model-theoretic semantics for this connective just as logicians give a model-theoretic semantics for all connectives. The main difference is that such semantics for defeasible conditionals do not look at *all* models of a theory to check whether it entails a formula (as semantics for deductive logics do) but only to a *preferred class* of models of the theory (for example, all models where things are as normal as possible). They would then add a strict inference rule $S \rightarrow \varphi$ to $\mathcal{R}_s$ just in case φ is true in *all* models of S, while they would add a defeasible inference rule $S \Rightarrow \varphi$ to $\mathcal{R}_d$ just in case φ is true in all *preferred* models of S but *not* in all models of S.

Now what inference rules for $\rightsquigarrow$ could result from such an approach? On two things there is consensus between logicans: modus ponens for $\rightsquigarrow$ is defeasibly but not deductively valid, so the rule $\varphi \rightsquigarrow \psi, \varphi \Rightarrow \psi$ should go into $\mathcal{R}_d$. There is also consensus that contraposition for $\rightsquigarrow$ is deductively invalid, so the rule $\varphi \rightsquigarrow \psi \rightarrow -\psi \rightsquigarrow -\varphi$ should *not* go into $\mathcal{R}_s$. However, here the consensus ends. Should the defeasible analogue of this rule go into $\mathcal{R}_d$ or not? Opinions differ at this point[9].

Let us illustrate the difference between the two approaches with a further variation on Example 6.4.1. Above we used the approach where all defeasible generalisations are inference rules. We now replace the two domain-specific defeasible inference rules d_1 and d_2 with two object-level conditionals expressed in $\mathcal{L}$ and now add them to $\mathcal{K}_p$:

$$WearsRing \rightsquigarrow Married$$
$$PartyAnimal \rightsquigarrow Bachelor$$

Moreover, we add defeasible modus ponens for $\rightsquigarrow$ to $\mathcal{R}_d$:

$$\mathcal{R}_d = \{\varphi \rightsquigarrow \psi, \varphi \Rightarrow \psi\}$$

The arguments then change as follows (assuming the crude way of incorporating classical logic):

[9] See Chapter 4 of Caminada (2004) for a very readable overview of the discussion.

$$\begin{aligned}
A_1&: \quad \mathit{WearsRing}\\
A_2&: \quad \mathit{WearsRing} \rightsquigarrow \mathit{Married}\\
A_3&: \quad A_1, A_2 \Rightarrow \mathit{Married}\\
A_4&: \quad \mathit{Married} \supset \neg \mathit{Bachelor}\\
A_5&: \quad A_3, A_4 \rightarrow \neg \mathit{Bachelor}
\end{aligned}$$

$$\begin{aligned}
B_1&: \quad \mathit{PartyAnimal}\\
B_2&: \quad \mathit{PartyAnimal} \rightsquigarrow \mathit{Bachelor}\\
B_3&: \quad B_1, B_2 \Rightarrow \mathit{Bachelor}\\
B_4&: \quad \mathit{Married} \supset \neg \mathit{Bachelor}\\
B_5&: \quad B_3, B_4 \rightarrow \neg \mathit{Married}
\end{aligned}$$

Now A_5 rebuts B_5 on B_3 while B_5 rebuts A_5 on A_3.

Concluding, if you want, you can base at least some of your choices concerning defeasible inference rules on model-theoretic semantics for nonmonotonic logics. However, it is an open question whether a model-theoretic semantics is the *only* criterion by which we can choose our defeasible rules. Some have based their choice on other criteria, since they do not primarily see defeasible rules as logical inference rules but as principles of human cognition or rational action, so that they should be based on foundations other than semantics. For example, John Pollock based his defeasible reasons on his account of epistemology (the part of philosophy that studies how we can obtain knowledge). Others have based their choice of defeasible reasons on the study of argument schemes in informal argumentation theory. We give examples of both these approaches in Section 6.4.3.

Naming defaults in first-order languages

We finally illustrate some subtleties of the naming convention for defeasible rules. If domain-specific defeasible rules are defined over a first-order language, then the same notational naming convention is often used as for defaults in default logic. A rule with free variables is used as a scheme for all its ground instances, that is, for all its instances in which the variable x is replaced by a ground term from $\mathcal{L}$. Moreover, the scheme is often given a name $d(x_1, \ldots, x_n)$, where $x_1, \ldots, x_n$ are all free variables that occur in the scheme. Such a name allows the formulation of undercutters to a rule. Consider, for example:

$$d(x): \quad \texttt{Bird}(x) \Rightarrow \texttt{Flies}(x)$$

Then schemes for undercutters can be written as follows:

$$u(x): \quad \texttt{Penguin}(x) \Rightarrow \neg d(x)$$

To see how this naming convention can be used, consider the following knowledge base:

$$\begin{aligned}
K_n &= \quad \{\forall x(\texttt{Penguin}(x) \supset \texttt{Bird}(x))\}\\
K_p &= \quad \{\texttt{Penguin}(\mathit{Tweety}), \texttt{Bird}(\mathit{Polly})\}
\end{aligned}$$

Then two arguments can be constructed for the conclusions that Tweety and Polly can fly (the strict rules are assumed to be all valid first-order inferences):

$$\begin{aligned}
A_1&: \quad \texttt{Penguin}(\mathit{Tweety})\\
A_2&: \quad \forall x(\texttt{Penguin}(x) \supset \texttt{Bird}(x))\\
A_3&: \quad A_1, A_2 \rightarrow \texttt{Bird}(\mathit{Tweety})\\
A_4&: \quad A_3 \Rightarrow \texttt{Flies}(\mathit{Tweety})
\end{aligned}$$

$$\begin{aligned}
B_1&: \quad \texttt{Bird}(\mathit{Polly})\\
B_2&: \quad B_1 \Rightarrow \texttt{Flies}(\mathit{Polly})
\end{aligned}$$

However, only for Tweety can an undercutter be constructed:

$$\begin{aligned}
C_1&: \quad \texttt{Penguin}(\mathit{Tweety})\\
C_2&: \quad C_1 \Rightarrow \neg d(\mathit{Tweety})
\end{aligned}$$

The point is that $d(x)$ is not a rule name but a rule name scheme, and only for its instance $d_1(Tweety)$ can an undercutter be constructed. If, by contrast, the birds-fly rule had been named with d, then applying the undercutter for Tweety would also block the default for Polly, which is clearly undesirable.

6.4.2 Satisfying rationality postulates

We are now in a position to state under what conditions *ASPIC*$^+$ satisfies Caminada and Amgoud (2007)'s four rationality postulates. These are listed below (it is helpful to refer to concepts defined in Definition 6.3.3 when reading these postulates), adapted to the *ASPIC*$^+$ framework.[10]

Definition 6.4.4 **[Rationality postulates for *ASPIC*$^+$]** Let *(c-)SAF* $= (\mathcal{A}, \mathcal{C}, \preceq)$ be an *ASPIC*$^+$ (c-)structured argumentation framework defined by an *ASPIC*$^+$ AT with $AS = (\mathcal{L}, \mathcal{R}, n)$ and $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_p$. Let AF be the abstract argumentation framework corresponding to *(c-)SAF* and let $T \in \{$complete, preferred, grounded, stable$\}$. Then:

- *(c-)SAF* satisfies the *closure under subarguments postulate* iff for all T-extensions E of AF it holds that if an argument A is in E then all subarguments of A are in E;

- *(c-)SAF* satisfies the *direct consistency postulate* iff for all T-extensions E of AF it holds that $\texttt{Conc}(E)$ is directly consistent;

- *(c-)SAF* satisfies the *indirect consistency postulate* iff for all T-extensions E of AF it holds that $\texttt{Conc}(E)$ is ndirectly consistent;

- *(c-)SAF* satisfies the *strict closure postulate* iff for all T-extensions E of AF it holds that $\texttt{Conc}(E) = Cl_{\mathcal{R}_s}(\texttt{Conc}(E))$.

The first postulate, closure under subarguments, holds unconditionally for the present framework.

Proposition 6.4.5 Let $\langle args, defeat\rangle$ be an abstract argumentation framework as defined in Definition 6.3.16 and E any of its grounded, preferred or stable extensions. Then

- for all $A \in E$: if $A' \in \texttt{Sub}(A)$ then $A' \in E$;

- $\texttt{Conc}(E) = Cl_{\mathcal{R}s}(\texttt{Conc}(E))$.

The two consistency postulates do not hold in general.

Example 6.4.6 A simple counterexample to consistency is with two defeasible rules d_1: $\Rightarrow p$ and d_2: $\Rightarrow q$ and a strict rule $p \rightarrow \neg q$, where $d_1 < d_2$. Then with the weakest- or last-link ordering the argument for $\neg q$ does not defeat the argument for q so in all semantics we have a single extension with both arguments.

[10] Caminada and Amgoud (2007) also propose postulates for the intersection of extensions and their conclusion sets, but since their satisfaction directly follows from satisfaction of the postulates for individual extensions, these postulates will below be ignored.

We now discuss under which conditions the consistency postulates are satisfied.

Depending on the choices outlined in Section 6.4.1, the first requirement for satisfying the consistency postulates is that your argumentation theory is closed under transposition or contraposition. This is because if neither property is satisfied, then since strict rule applications cannot be attacked, direct consistency may then be violated. This can be illustrated with the first version of Example 6.4.1. Suppose we only have one strict rule, namely, s_1. we cannot construct B_3, since B_3 applies the now missing rule s_2. We still have that A_3 rebuts B_2. Suppose now that $d_1 < d_2$ and we apply the last-link argument ordering. Then A_3 does not defeat B_2. In fact, no argument in the example is defeated, so we end up with a single extension in all semantics, which contains arguments for both $Bachelor$ and $\neg Bachelor$ and so violates direct and indirect consistency.

However, with transposition this bad outcome is avoided: if we also have s_2, then argument B_3 can be constructed, which rebuts A_3 on A_2. Again applying the preference $d_1 < d_2$ with the last-link ordering, we have that B_3 strictly defeats A_2. Again we have a unique extension in all semantics, containing all arguments except A_2 and A_3. This extension does not violate consistency.

Example 6.4.7 Consider Example 6.3.6. As discussed in Example 6.3.18, if the argument ordering is such that C_3 does not defeat B_1, then both arguments will be in the same extension, which thus violates consistency since the conclusions of these arguments contradict each other. However, if the transposition $s \rightarrow \neg v$ of $v \rightarrow \neg s$ is added to $\mathcal{R}_s$, then B_1 can be continued to an argument for $\neg v$, which successfully rebuts C_3 on C_2, excluding the consistency-violating extensions.

Some say that the above violation of consistency, before inclusion of the transposed rule, arises because *ASPIC*$^+$ forbids attacks on strictly derived conclusions. Consistency would not be violated if B_2 was allowed to attack $A3$ in the first version of Example 6.4.1. However, apart from the reasons discussed in Section 6.2, there is another reason for prohibiting attacks on strictly derived conclusions: if they are allowed, then extensions may not be strictly closed or indirectly consistent, even if the strict rules are closed under transposition. To see why, suppose we changed *ASPIC*$^+$'s definitions to allow attacks on strict conclusions, so that B_2 attacks A_3, A_2 attacks B_3, and A_3 and B_3 attack each other in Example 6.4.1. Suppose also that all knowledge-base items and all defeasible rules in the example are of equal preference, and suppose we apply the weakest- or last-link argument ordering. Then all rebutting attacks in the example succeed. But then the set $\{A_1, A_2, B_1, B_2\}$ is admissible and is in fact both a stable and preferred extension. But this violates the rationality postulates of strict closure and indirect consistency. The extension contains an argument for $Bachelor$ but not for $\neg Married$, which strictly follows from it by rule s_2. Likewise, the extension contains an argument for $Married$ but not for $\neg Bachelor$, which strictly follows from it by rule s_1. So the extension is not closed under strict rule application. Moreover, the extension is indirectly inconsistent, since its strict closure contains both $Married$ and $\neg Married$, and both $Bachelor$ and $\neg Bachelor$.

Other requirements for satisfying the consistency postulates are that the axioms $\mathcal{K}_n$ are indirectly consistent (axiom consistency) and the preference ordering is *reasonable*. The rationale for requiring the former is self-evident. A reasonable argument ordering essentially amounts to requiring that: 1) arguments that are both strict and firm are strictly preferred over all other arguments; 2) the strength (and implied relative preference) of an argument is determined exclusively by the defeasible rules and/or ordinary premises; 3) the preference ordering is acyclic, and if $B \prec A$ then it must be that $B' \prec A$ where B' is some maximal fallible (i.e., defeasible or plausible) sub-argument of B (for example in our running example C_2 but not C_1 is a maximal fallible argument of C_3). We refer the reader to Modgil and Prakken (2013) for the technical definition of a reasonable ordering; suffice to say that it has been shown that the weakest- and last-link argument orderings of Section 6.3.4 are reasonable.

We are now in a position to state an important result proved in Modgil and Prakken (2013) that if your *(c-)SAF* is *well-defined*, in that its arguemntation theory satisfies axiom consistency, and transposition or contraposition, and your argument preference ordering is reasonable, then the consistency postulates are satisfied by the *ASPIC*$^+$ framework as defined in Section 6.3.

Theorem 6.4.8 Let $\langle args, defeat \rangle$ be an abstract argumentation framework corresponding to a well-defined *(c-)SAF* and let E be any of its grounded, preferred or stable extensions. Then

- $\texttt{Conc}(E)$ is consistent;
- $Cl_{\mathcal{R}s}(\texttt{Conc}(E))$ is consistent.

Finally, note that if you do not include any strict rules or axiom premises in your argumentation theory, then the requirement that your *(c-)SAF* be well defined obviously does not apply, but it is also worth noting that the preference ordering need *not* be reasonable in order that all four rationality postulates be satisfied (indeed no assumptions as to the properties of the preference ordering are required in this case).

6.4.3 Using *ASPIC*$^+$ to model argument schemes

We concluded Section 6.4.1 by remarking on the use of defeasible inference rules as principles of cognition in John Pollock's work and as argument schemes in informal argumentation theory. We now illustrate how both approaches can be formalised in *ASPIC*$^+$ and how strict inference rules can also be accommodated when doing so.

Let us first look in more detail at John Pollock's work. He formalised defeasible rules for reasoning patterns involving perception, memory, induction, temporal persistence and the statistical syllogism, as well as undercutters for these reasons.

In *ASPIC*$^+$ his principles of perception and memory can be written as follows:

$$\begin{array}{ll} d_p(x,\varphi): & \texttt{Sees}(x,\varphi) \Rightarrow \varphi \\ d_m(x,\varphi): & \texttt{Recalls}(x,\varphi) \Rightarrow \varphi \end{array}$$

In fact, these defeasible inference rules are schemes for all their ground instances (that is, for any instance where x and φ are replaced by ground terms denoting a specific perceiving agent and a specific perceived state of affairs). Therefore, their names $d_p(x,\varphi)$ and $d_m(x,\varphi)$ as assigned by the n function are in fact also schemes for names. A proper name is obtained by instantiating these variables by the same ground terms as used to instantiate these variables in the scheme. Thus it becomes possible to formulate undercutters for one instance of the scheme (say for Jan who saw John in Amsterdam) while leaving another instance unattacked (say for Bob who saw John in Holland Park).

Note, finally, that these schemes assume a naming convention for formulas in a first-order language, since φ is a term in the antecedent while it is a well-formed formula in the consequent. In the remainder we will leave this naming convention implicit.

Now undercutters for d_p state circumstances in which perceptions are unreliable, while undercutters of d_m state conditions under which memories may be flawed. For example, a well-known cause of false memories of events is that the memory is distorted by, for instance, seeing pictures in the newspaper or watching a TV programme about the remembered event. A general undercutter for distorted memories could be

$$u_m(x,\varphi): \quad \texttt{DistortedMemory}(x,\varphi) \Rightarrow \neg d_m(x,\varphi)$$

combined with information such as

$$\forall x,\varphi(\texttt{SeesPicturesAbout}(x,\varphi) \supset \texttt{DistortedMemory}(x,\varphi))$$

Pollock's epistemic inference schemes are in fact a subspecies of argument schemes. The notion of an argument scheme was developed in philosophy and is currently an important topic in the computational study of argumentation. Argument schemes are stereotypical non-deductive patterns of reasoning, consisting of a set of premises and a conclusion that is presumed to follow from them. Uses of argument schemes are evaluated in terms of critical questions specific to the scheme. An example of an epistemic argument scheme is the scheme from the position to know (Walton, 1996, pp. 61–63):

$$\frac{\begin{array}{l} A \text{ is in the position to know whether } P \text{ is true} \\ A \text{ asserts that } P \text{ is true} \end{array}}{P \text{ is true}}$$

Walton gives this scheme three critical questions:

1. Is A in the position to know whether P is true?
2. Did A assert that P is true?
3. Is A an honest (trustworty, reliable) source?

A natural way to formalise reasoning with argument schemes is to regard them as defeasible inference rules and to regard critical questions as pointers to counterarguments. For example, in the scheme from the position to know questions (1) and (2) point to underminers (of, respectively, the first and second premise) while questions (3) points to undercutters (the exception that the person is for some reason not credible).

Accordingly, we formalise the position to know scheme and its undercutter as follows:

$$\begin{array}{ll} d_w(x,\varphi): & \texttt{PositionToKnow}(x,\varphi), \texttt{Says}(x,\varphi) \Rightarrow \varphi \\ u_w(x,\varphi): & \neg\texttt{Credible}(x) \Rightarrow \neg d_w(x,\varphi) \end{array}$$

We will now illustrate the modelling of both Pollock's defeasible reasons and Walton's argument schemes with our example from Section 6.2, focusing on a specific class of persons who are in the position to know, namely, witnesses. In fact, witnesses always report about what they observed in the past, so they will say something like "I remember that I saw that John was in Holland Park". Thus an appeal to a witness testimony involves the use of three schemes: first the position to know scheme is used to infer that the witness indeed remembers that he saw that John was in Holland Park, then the memory scheme is used to infer that he indeed saw that John was in Holland Park, and finally, the perception scheme is used to infer that John was indeed in Holland Park. Now recall that John was a suspect in a robbery in Holland Park and that Jan testifed that he saw John in Amsterdam on the same morning, while Jan is a friend of John.

Suppose now we also receive information that Bob read newspaper reports about the robbery in which a picture of John was shown. One way to model this in *ASPIC*$^+$ is as follows.

The knowledge base consists of the following facts (since we don't want to dispute them, we put them in $\mathcal{K}_n$):

f_1: $\mathtt{PositionToKnow}(Bob, \mathtt{Recalls}(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John))))$
f_2: $\mathtt{Says}(Bob, \mathtt{Recalls}(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John))))$
f_3: $\mathtt{SeesPicturesAbout}(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John)))$
f_4: $\forall x, \varphi.(\mathtt{SeesPicturesAbout}(x, \varphi) \supset \mathtt{DistortedMemory}(x, \varphi))$
f_5: $\forall x.\mathtt{InHollandPark}(x) \supset \mathtt{InLondon}(x)$
f_6: $\mathtt{PositionToKnow}(Jan, \mathtt{Recalls}(Jan, \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John))))$
f_7: $\mathtt{Says}(Jan, \mathtt{Recalls}(Jan, \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John))))$
f_8: $\mathtt{Friends}(Jan, John)$
f_9: $\mathtt{SuspectedRobber}(John)$
f_{10}: $\forall x, y, \varphi.\mathtt{Friends}(x, y) \wedge \mathtt{SuspectedRobber}(y) \wedge \mathtt{InvolvedIn}(y, \varphi) \supset \neg\mathtt{Credible}(x)$
f_{11}: $\mathtt{InvolvedIn}(John, \mathtt{Recalls}(Jan, \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John))))$
f_{12}: $\forall x \neg(\mathtt{InAmsterdam}(x) \wedge \mathtt{InLondon}(x))$

Combining this with the schemes from perception, memory and position to know, we obtain the following arguments (for reasons of space we don't list separate lines for arguments that just take an item from $\mathcal{K}$).

A_3: $f_1, f_2 \Rightarrow_{dw} \mathtt{Recalls}(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John)))$
A_4: $A_3 \Rightarrow_{dm} \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John))$
A_5: $A_4 \Rightarrow_{dp} \mathtt{InHollandPark}(John)$
A_7: $A_5, f_5 \rightarrow \mathtt{InLondon}(John)$

This argument is undercut (on A_4) by the following argument applying the undercutter for the memory scheme:

B_3: $f_3, f_4 \rightarrow \mathtt{DistortedMemory}(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John)))$
B_4: $B_3 \Rightarrow_{um} \neg d_m(Bob, \mathtt{Sees}(Bob, \mathtt{InHollandPark}(John)))$

Moreover, A_7 is rebutted (on A_5) by the following argument:

C_3: $f_6, f_7 \Rightarrow_{dw} \mathtt{Recalls}(Jan, \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John)))$
C_4: $C_3 \Rightarrow_{dm} \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John))$
C_5: $C_4 \Rightarrow_{dp} \mathtt{InAmsterdam}(John)$
C_8: $C_5, f_5, f_{12} \rightarrow \neg\mathtt{InLondon}(John)$

This argument is also undercut, namely, on C_3 based on the undercutter of the position to know scheme:

D_5: $f_8, f_9, f_{10}, f_{11} \rightarrow \neg\mathtt{Credible}(Jan)$
D_6: $D_5 \Rightarrow_{uw} \neg d_w(Jan, \mathtt{Recalls}(Jan, \mathtt{Sees}(Jan, \mathtt{InAmsterdam}(John))))$

Finally, C_8 is rebutted on C_5 by the following continuation of argument A_7:

A_8: $A_5, f_5, f_{12} \Rightarrow \neg\mathtt{InAmsterdam}(John)$

A_8 is in turn undercut by B_4 (on A_4) and rebutted by C_8 (on A_5).

The example is displayed in Figure 6.4.

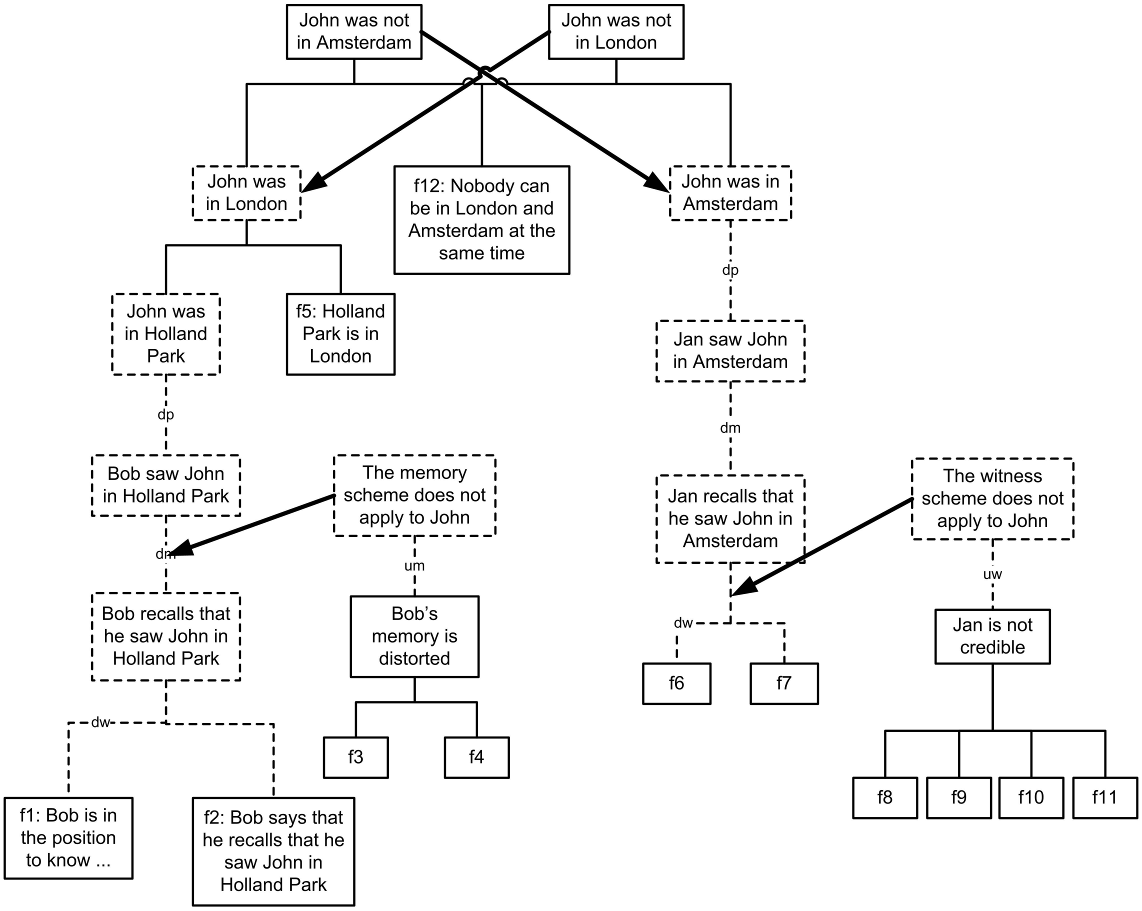

John was not in Amsterdam
John was not in London
John was in London
f12: Nobody can be in London and Amsterdam at the same time
John was in Amsterdam
dp
John was in Holland Park
f5: Holland Park is in London
Jan saw John in Amsterdam
dp
dm
Bob saw John in Holland Park
The memory scheme does not apply to John
Jan recalls that he saw John in Amsterdam
The witness scheme does not apply to John
dm
um
uw
Bob recalls that he saw John in Holland Park
Bob's memory is distorted
dw
Jan is not credible
f6
f7
dw
f3
f4
f8
f9
f10
f11
f1: Bob is in the position to know ...
f2: Bob says that he recalls that he saw John in Holland Park

Figure 6.4: A formalised example

Because of the two undercutting arguments, neither of the testimony arguments are credulously or sceptically justified in any semantics. Let us see what happens if we do not have the two undercutters. Then we must apply preferences to the rebutting attack of C_8 on A_5 and to the rebutting attack of A_8 on C_5. As it turns out, the same preferences have to be applied in both cases, namely, those between the three defeasible-rule applications in the respective arguments. And this is what we intuitively want.

Finally, we note that counterarguments based on critical questions of argument schemes may themselves apply argument schemes. For example, we may believe that Jan and John are friends because another witness told our so. Or we may believe that Holland Park is in London because a London taxi driver told us so (an application of the so-called expert testimony scheme).

6.4.4 Instantiations with no defeasible rules

All that has been said so far about ways to choose the strict rules applies irrespective of whether you also want to include defeasible rules in your argumentation system. In fact, *ASPIC*$^+$ allows you to only use strict inference rules. Principled ways to do so are to base the strict rules on classical logic or indeed on any Tarskian consequence relation. In this way, *ASPIC*$^+$ extends the classical-logic approach of Besnard and Hunter (2009) and the abstract-logic approach of Amgoud and Besnard (2009), by providing guidelines for using preferences to resolve inconsistencies in classical logic or any other underlying Tarksian logic. The use of preferences is of particular importance in such contexts, since in these contexts the stable and preferred extensions of Dung frameworks simply correspond to the maximal consistent subsets of the instantiating theories (Amgoud and Besnard, 2013). One thus needs some 'extra-logical' means, such as preferences, to resolve inconsistencies.

The idea is as follows. Given a set S of wff in some language $\mathcal{L}$ and a Tarksian consequence relation Cn over $\mathcal{L}$ (note that classical consequence is such a Tarskian consequence relation), we let the axioms and defeasible inference rules be empty, and the strict rules defined as indicated in Section 6.4.1, namely, as $S \rightarrow p \in \mathcal{R}_s$ iff $p \in Cn(S)$, for any finite $S \subseteq \mathcal{L}$. Furthermore, in keeping with the above mentioned classical, and more general Tarskian Logic approaches, we assume all arguments to be consistent and, moreover, their premise sets subset-minimal in applying their conclusion.

For this special case all *ASPIC*$^+$ arguments are strict, so all attacks are undermining attacks. In Modgil and Prakken (2013) it was shown that these *ASPIC*$^+$ reconstructions of Tarskian and classical approaches are equivalent to the originals if these originals use a form of undermining attack. Moreover, the result stated in Section 6.4.1 – that any *ASPIC*$^+$ AT with the strict rules derived from a Tarskian logic satisfies closure under contraposition — then implies that without preferences these reconstructions are well-defined and thus satisfy the rationality postulates. Moreover, if these reconstructions are extended with a reasonable argument ordering, then this result also holds for the case with preferences. Thus the *ASPIC*$^+$ framework has in fact been used to extend both the classical-logical approach of Besnard and Hunter (2009) and the more general Tarskian approach of Amgoud and Besnard (2009) with preferences in a way that satisfies all rationality postulates of Caminada and Amgoud (2007). A final result of Modgil and Prakken (2013) is that if a thus defined classical-logic instantiation of *ASPIC*$^+$is combined with a total priority ordering $\leq'$, then one obtains a correspondence with Brewka (1989)'s Preferred Subtheories.

6.4.5 Illustrating uses of *ASPIC*$^+$ with and without defeasible rules

In this section we compare respective uses of *ASPIC*$^+$ with and without defeasible rules in more detail. We first say more about the arguments of some that classical-logic simulations of defeasible rules may yield counterintuitive results. Let us assume a classical-logic instantiation of *ASPIC*$^+$ as defined in Section 6.4.4 and formalise natural-language generalisations 'If P then normally Q' as material implications $P \supset Q$ put in $\mathcal{K}_p$. The idea is that since $P \supset Q$ is an ordinary premise, its use as a premise can be undermined in exceptional cases. Observe that by classical reasoning we then have a strict argument for $\neg Q \supset \neg P$. Some say that this is problematic. Consider the following example: 'Anyone who is a man usually has no beard', so (strictly) 'Anyone who has a beard usually is not a man'. This strikes some as counterintuitive, since we know that virtually everyone who has a beard is a man, so the contraposition of 'If P then normally Q' cannot be deductively valid[11].

[11] One way to argue why classical simulations may give counter-intuitive results is to recall that a number of researchers provide statistical semantics for defeasible inference rules. These semantics regard a defeasible rule of the form $P \Rightarrow Q$ as a qualitative approximation of the statement that the conditional probability of Q, given P, is high. The laws of probability theory then tell us that this does not entail that the conditional probability of $\neg P$, given $\neg Q$, is high. The problem with the classical-logic approach is

A more refined classical approach is to give the material implication an extra normality condition N, which informally reads as 'everything is normal as regards P implying Q', and which is also put in $\mathcal{K}_p$. The idea then is that exceptional cases give rise to underminers of N. However, $(P \wedge N) \supset Q$ also deductively contraposes, namely, as $(\neg Q \wedge N) \supset \neg P$, so it seems that we still have the controversial deductive validity of contraposition for generalisations (in the beard and men example the contraposition of the rule with the added normality condition would read: 'Anyone who has a beard and all is normal regarding men and having beards, usually is not a man' !).

So far we only discussed reasons for belief but argumentation is often about what to do, prefer or value (what philosophers often call *practical reasoning*). Here too it has been argued on philosophical grounds that reasons for doing, preferring or valuing cannot be expressed in classical logic since they do not contrapose. This view can, of course, not be based on a statistical semantics for such reasons, since statistics only applies to reasoning about what is the case (what philosophers often call *epistemic reasoning*). Space limitations prevent us from giving more details about these philosophical arguments.

We next illustrate two different ways to use *ASPIC*$^+$ with a detailed example. Both ways use classical logic in their strict part and use explicit preferences, but only the second way uses defeasible inference rules. The first way instead expresses defeasible generalisations as material implications with normality assumptions. The example will shed further light on the issue whether empirical generalisations can be represented in classical logic, and it will also motivate the use of axiom premises. Our example is a well-known one from the literature on nonmonotonic logic. Suppose a defeasible reasoner accepts all following natural-language statements are true. For the generalisations (1) and (2) this means that the reasoner accepts that they hold in general but that they may have exceptions.

(1) Birds normally fly
(2) Penguins normally don't fly
(3) All penguins are birds
(4) Penguins are abnormal birds with respect to flying
(5) Tweety is a penguin

A defeasible reasoner then wants to know what can be concluded from this information about whether Tweety can fly. It seems uncontroversial to say that any defeasible reasoner will conclude that Tweety can fly.

We now formalise these statements with the just-explained method to represent empirical generalisations as material implications with explicit normality assumptions. We use a classical-logic instantiation of *ASPIC*$^+$ with preferences as defined above in Section 6.4.4.

(1) $\mathit{bird} \wedge \neg ab_1 \supset \mathit{canfly}$
(2) $\mathit{penguin} \wedge \neg ab_2 \supset \neg \mathit{canfly}$
(3) $\mathit{penguin} \supset \mathit{bird}$
(4) $\mathit{penguin} \supset ab_1$
(5) $\mathit{penguin}$

Let us first add these formulas to $\mathcal{K}_p$. The idea now is that the normality assumptions of a defeasible reasoner are expressed as additional statements $\neg ab_1$ and $\neg ab_2$, also added

then that it conflates this distinction by turning the conditional probability of Q given P into the unconditional probability of $P \supset Q$, which then has to be equal to the unconditional probability of $\neg Q \supset \neg P$.

to $\mathcal{K}_p$. We then define the preference ordering on $\mathcal{K}_p$ such that all of (1-5) are strictly preferred over any of these two assumptions and that $\neg ab_1 <' \neg ab_2$.

We can then construct many arguments on the issue whether Tweety can fly. Note that $\{1, 2, 3, 4, 5\} \cup \{\neg ab_1, \neg ab_2\}$ is minimally inconsistent, so if we take any single element out, the rest can be used to build an argument against it. This means that we can formally build arguments not just against the two normality assumptions but also against any of (1-5) (note the similarity with the fact that, as noted above, in classical-logic argumentation without preferences the stable and preferred extensions corespond to maximal consistent subsets of the knowledge base). With the weakest- or last-link ordering we do obtain the intuitive conclusion $\neg \mathit{canfly}$, but the fact that arguments against any of (1-5) can be built may be regarded as somewhat odd, since we just noted that a defeasible reasoner accepts (1-5) as given and is only interested in what follows from them.

Let us therefore move (1-5) to the axioms $\mathcal{K}_n$, so that they cannot be attacked. Then we have just a few arguments on the issue whether Tweety can fly: we have an argument $\{1, 2, 3, 4, 5\} \cup \{\neg ab_2\} \rightarrow \neg \mathit{canfly}$, which has one attacker, namely, $\{1, 2, 3, 5\} \cup \{\neg ab_1\} \rightarrow ab_2$. However, with the weakest- or last link principle this attacker does not defeat it target, since we have $\neg ab_1 <' \neg ab_2$. Hence $\neg \mathit{canfly}$ is justified in any semantics. So at first sight it would seem that the classical-logic approach enriched with axiom premises adequately models reasoning with empirical generalisations.

However, this approach still has some things to explain, as can be illustrated by changing our example a little: above it was given as a matter of fact that Tweety is a penguin but in reality the particular 'facts' of a problem are often not simply given but derived from information sources (sensors, testimonies, databases, the internet, and so on). Now in reality none of these sources is fully reliable, so inferring facts from them can only be done under the assumption that things are normal. So let us change the example by saying that Tweety was observed to be a penguin and that animals that are observed to be penguins *normally* are penguins. We change 5 to $5'$ and we add 6 to $\mathcal{K}_n$:

(5') $\mathit{observed\_as\_penguin}$
(6) $\mathit{observed\_as\_penguin} \wedge \neg ab_3 \supset \mathit{penguin}$

Moreover, we add $\neg ab_3$ to $\mathcal{K}_p$. We can still build an argument that Tweety cannot fly, namely, $\{1, 2, 3, 4, 5'\} \cup \{\neg ab_2, \neg ab_3\} \rightarrow \neg \mathit{canfly}$. However, we can also build an attacker of this argument, namely $\{1, 2, 3, 4, 5', 6\} \cup \{\neg ab_1, \neg ab_2\} \rightarrow ab_3$. We can still obtain the intuitive outcome by preferring the assumption $\neg ab_3$ over the assumption $\neg ab_1$. However, some have argued that this is an ad-hoc solution, since there would be no general principle on which such a preference can be based. The heart of the problem, they say, is the fact that the material implication satisfies contraposition, a property which, as we just mentioned, can be argued to be too strong for defeasible generalisations. In reality a defeasible reasoner would not even construct an argument against $\mathit{penguin}$. As can be easily checked, the same issues arise if we put (1-4,5',6) in $\mathcal{K}_p$ while we then have our old issue back that arguments can be constructed against any element of $\mathcal{K}_p$.

Concluding so far, those who want to model 'default reasoning' in classical argumentation have to explain why arguments as the one for ab_3 can be constructed and why it does not defeat the argument for $\neg \mathit{canfly}$ (or alternatively, why the latter conclusion is not justified). Moreover, if they apply the first version of this approach, by putting all of $\{1, 2, 3, 4, 5', 6\}$ in $\mathcal{K}_p$, then they also have to explain why arguments against any of these premises can be constructed and whether these arguments succeed as defeats.

Let us next formalise the example with domain-specific defeasible rules and with the strict rules still corresponding to classical logic.

d_1: $bird \Rightarrow canfly$
d_2: $penguin \Rightarrow \neg canfly$
d_3: $observed\_as\_penguin \Rightarrow \neg penguin$
f_1: $penguin \supset bird$
f_2: $penguin \supset \neg r_1$
f_3: $observed\_as\_penguin$

It now does not matter whether we put the facts in $\mathcal{K}_n$ or $\mathcal{K}_p$, nor does it matter which priorities we define on $\mathcal{K}_p$ or $\mathcal{R}_d$. We have the following arguments:

A_1:	$observed\_as\_penguin$	B_1:	$A_2 \Rightarrow \neg canfly$
A_2:	$A_1 \Rightarrow penguin$		
A_3:	$penguin \supset bird$		
A_4:	$A_2, A_3 \Rightarrow canfly$	C_1:	$A_2 \Rightarrow \neg r_1$

Note also that no argument can be built against the conclusion $penguin$. We have that A_4 and B_1 rebut each other while C_1 undercuts A_4. Whatever the argument ordering between A_4 and B_1, we thus obtain that the conclusion $\neg canfly$ is justified in any semantics.

Concluding, the classical modelling of this example is simpler in that it only uses classical inference and does not have to rely on the notion of a defeasible inference rule. On the other hand, to obtain the intuitive outcome it needs more preferences than the modelling with defeasible rules, while the issue arises on which grounds these preferences can be stated. Moreover, if the classical approach regards all knowledge as fallible in principle, then it generates many more arguments than perhaps intuitively expected, at least many more than in the modelling with defeasible rules.

6.4.6 Representing facts

ASPIC$^+$ allows you to represent facts in various ways, each with their pros and cons. *Disputable facts* φ can either be put as such in $\mathcal{K}_p$ or as defeasible rules $\Rightarrow \varphi$ with empty antecedents. An advantage of including disputable facts in $\mathcal{K}_p$ is that thus *ASPIC*$^+$ captures classical and abstract-logic argumentation with preferences as special cases. On the other hand, if disputable facts φ are represented as defeasible rules $\Rightarrow \varphi$, then the definition of the weakest- and last-link argument orderings becomes simpler, since then only sets of defeasible rules need to be compared. In addition, this choice removes the need for undermining attack, which simplifies the definitions of attack and defeat.

Undisputable facts φ can either be put as such in $\mathcal{K}_n$ or as strict rules $\rightarrow \varphi$ with empty antecedents. This choice does not make a difference for the weakest- or last-link argument ordering, since these orderings disregard axiom premises and strict rules. However, a disadvantage of representing undisputable fact φ as strict rules $\rightarrow \varphi$ is that then the strict rules do not express a logic any more, so the above-mentioned theorems on definitions of $\mathcal{R}_s$ in terms of Tarskian abstract logics do not apply any more.

6.4.7 Summary

We have seen that *ASPIC*$^+$ allows you to make any choice of axioms, strict and defeasible rules you like. You can choose domain-specific strict and/or defeasible inference rules, and you can choose logical strict and/or defeasible inference rules, for any deductive and/or nonmonotonic logic of your choice, good or bad. You can add logical axioms to $\mathcal{K}_n$ but you can also add any other information to $\mathcal{K}_n$ that you don't want to put up for discussion. You can also base your defeasible rules on informal accounts of argument schemes. All that *ASPIC*$^+$ tells you is how arguments can be built with your rules of choice, how they can be attacked, and how these attacks can be resolved, given an argument ordering of your choice. Moreover, we have some theorems about *ASPIC*$^+$ that inform you about some properties of your choices.

6.5 Generalising negation in *ASPIC*$^+$

The notion of an argumentation system in Section 6.3.1, assumed a language $\mathcal{L}$ closed under negation ($\neg$), where the standard classical interpretation of $\neg$ licenses a symmetric notion of conflict based attack, so that an argument consisting of an ordinary premise ϕ or with a defeasible top rule concluding ϕ, *symmetrically* attacks an argument consisting of an ordinary premise $\neg\phi$ or with a defeasible top rule concluding $\neg\phi$. However, the *ASPIC*$^+$framework as presented in Prakken (2010); Modgil and Prakken (2013), accommodates a more general notion of conflict, by defining an argumentation system to additionally include a function $^-$ that, for any wff $\psi \in \mathcal{L}$, specifies the set of wff's that are in conflict with ψ. With this idea, which is taken from assumption-based argumentation (Bondarenko *et al.*, 1997; Dung *et al.*, 2009), one can define both an asymmetric and symmetric notion of conflict-based attack. More formally:

Definition 6.5.1 $^-$ is a function from $\mathcal{L}$ to $2^{\mathcal{L}}$, such that:

- φ is a *contrary* of ψ if $\varphi \in \overline{\psi}$, $\psi \notin \overline{\varphi}$;
- φ is a *contradictory* of ψ (denoted by '$\varphi = -\psi$'), if $\varphi \in \overline{\psi}$, $\psi \in \overline{\varphi}$;
- each $\varphi \in \mathcal{L}$ has at least one contradictory.

Note that classical negation is now a special case of the symmetric contradictory relation: $\alpha \in \overline{\beta}$ iff α is of the form $\neg\beta$ or β is of the form $\neg\alpha$ (i.e., for any wff α, α and $\neg\alpha$ are contradictories). Modgil and Prakken (2013) then redefine Definition 6.3.3's notion of direct consistency so that a set S is *directly consistent* iff $\nexists\ \psi, \varphi \in S$ such that $\psi \in \overline{\varphi}$. Also, $\texttt{Conc}(A) \in \overline{\varphi}$ ($\texttt{Conc}(A) \in \overline{n(r)}$) replaces $\texttt{Conc}(A) = -\varphi$ ($\texttt{Conc}(A) = -n(r)$) in Definition 6.3.10's definition of attacks.

With this, one can reconstruct assumption-based argumentation (ABA) in *ASPIC*$^+$, since as noted above, ABA also generalises the notion of conflict through the use of a $^-$ function. Indeed, this reconstruction is formally shown in Prakken (2010), in which assumption premises were distinguished from ordinary premises, and used to model ABA assumptions. However, one can do without such specialised premises, and model assumptions as ordinary premises. So, to summarise, an *ASPIC*$^+$reconstruction of ABA will have empty sets of defeasible rules and axiom premises, and consist of ordinary premises and strict rules (respectively corresponding to the assumptions and rules in an ABA theory). Then, for every ordinary premise α, one specifies that:

1. there is a β in $\mathcal{L}$ such that β is a contrary or contradictory of α

2. α is not the conclusion of a strict inference rule (corresponding to so called 'flat' ABA theories)

Then, without the use of preference relation, a correspondence can be shown between ABA and *ASPIC*$^+$. Note that by reconstructing ABA in *ASPIC*$^+$, one can then identify conditions under which ABA satisfies rationality postulates (by requiring, for instance, that the strict rules are closed under transposition). For example, consider the *ASPIC*$^+$reconstruction of an ABA theory consisting of strict rules $a \rightarrow p$ and $b \rightarrow \neg p$, and ordinary premises (assumptions) $\{a, b\}$ such that a and $\neg a$ are contradictories, and b and $\neg b$ are contradictories. Consistency is violated since one can construct a single preferred (and grounded) extension containing arguments $A = [a; a \rightarrow p]$ and $B = [b; b \rightarrow \neg p]$, neither of which attack each other. However with the additional transpositions $p \rightarrow \neg b$ and $\neg p \rightarrow \neg a$, then extending A and B yields $A' = [a; a \rightarrow p; p \rightarrow \neg b]$ and $B' = [b; b \rightarrow \neg p; \neg p \rightarrow \neg a]$. A' and B' respectively attack B and A. So the set of arguments $\{A, B\}$ is no longer admissible (neither A or B can defend against these attacks).

The rationale for these more general notions of conflict and attack is two-fold. Firstly, one can for pragmatic reasons state that two formulae are in conflict, rather than requiring that one implies the negation of another; for example, assuming a predicate language with the binary '$<$' relation, one can state that any two formulae of the form $\alpha < \beta$ and $\beta < \alpha$ are contradictories. Secondly, the $^-$ function allows for an asymmetric notion of negation. This in turn is required for modelling negation as failure (as in logic programming). Using the negation as failure symbol $\sim$ (also called 'weak' negation, in contrast to the 'strong' negation symbol $\neg$), then $\sim \alpha$ denotes the negation of α under the assumption that α is not provable (i.e., the negation of α is assumed in the absence of evidence to the contrary). It is not then meaningful to assert that such an assumption brings into question (and so initiates an attack on) the evidence whose very absence is required to make the assumption in the first place. In other words, if A is an argument consisting of the premise $\sim \alpha$, and B concludes α (the contrary of $\sim \alpha$), then B attacks A, but not vice versa. Furthermore, since the very construction of A is invalidated by evidence to the contrary, i.e., B, then such attacks succeed as defeats *independently* of preferences.

To accommodate the notion of contrary, and attacks on contraries succeeding as defeats independently of preferences, we further modify Definition 6.3.10 to distinguish the special cases where $\texttt{Conc}(A)$ is a contrary of φ, in which case we say that A *contrary rebuts* B and A *contrary undermines* B, and then modify Definition 6.3.12 so that:

- A successfully rebuts B if A contrary rebuts B, or A rebuts B on B' and $A \not\prec B'$.
- A successfully undermines B if A contrary undermines B, or A undermines B on ϕ and $A \not\prec \phi$.

Following on from the discussion in Section 6.4.2, one can then show (Modgil and Prakken, 2013) that with the additional notion of contrary, satisfaction of the four rationality postulates not only requires that the argument theory satisfy axiom consistency, and transposition or contraposition, but also that it is *well formed* in the following sense:

Definition 6.5.2 An argumentation theory is *well-formed* if the following holds: if ϕ is a contrary of ψ then $\psi \notin \mathcal{K}_n$ and ψ is not the consequent of a strict rule.

To illustrate the use of negation as failure, suppose you want your arguments to be built from a propositional language that includes both $\neg$ and $\sim$. One could then define

$\mathcal{L}$ as a language of propositional literals, composed from a set of propositional atoms $\{a, b, c, \ldots\}$ and the symbols $\neg$ and $\sim$. Then:

- α is a *strong literal* if α is a propositional atom or of the form $\neg\beta$ where β is a propositional atom (strong negation cannot be nested).

- α is a wff of $\mathcal{L}$, if α is a strong literal or of the form $\sim \beta$ where β is a strong literal (weak negation cannot be nested).

Then $\alpha \in \overline{\beta}$ iff (1) α is of the form $\neg\beta$ or β is of the form $\neg\alpha$; or (2) β is of the form $\sim \alpha$ (i.e., for any wff α, α and $\neg\alpha$ are contradictories and α is a contrary of $\sim \alpha$). Finally, for any $\sim \alpha$ that is in the antecedent of a strict or defeasible inference rule, one is required to include $\sim \alpha$ in the ordinary premises.

Consider now Example 6.3.6, where we now have that $u \in \overline{\sim u}$, and we replace the rule $d_4 : u \Rightarrow v$ with d_4': $\sim u \Rightarrow v$, and add $\sim u$ to the ordinary premises: $\mathcal{K}_p = \{\sim u, s, u, x\}$. Then, the arguments C_3 and D_4 are now replaced by arguments C_3' and D_4' each of which contain the sub-argument $E : \sim u$ (instead of $C_1 : u$). Then $C_1 : u$ contrary undermines, and so defeats, C_3' and D_4' on $\sim u$.

We finally note that according to Toni (2014) the philosophy behind ABA is to translate preferences and defeasible rules into ABA rules plus ABA assumptions, so that rebutting and undercutting attack and the application of preferences all reduce to premise attack. The idea of this is to keep the formal theory simpler and to make the technical machinery of ABA available for other approaches. We agree that this approach has its merits but note that it is an open question whether *ASPIC*$^+$ can in its full generality be translated into ABA. Also, as we noted above, we claim that there is also some merit in having a theory with explicit notions of rebutting and undercutting attack and preference application, namely, if the aim is to formalise modes of reasoning in a way that corresponds with human modes of reasoning and debate.

6.6 Self-defeat

In Chapter 4, Section 4.2 we said that a proper analysis of self-defeating arguments must make the structure of arguments explicit. Now that we have done so, we can explain why this is needed. In the present framework two types of self-defeating arguments are possible: *serial self-defeat* occurs when an argument defeats one of its earlier steps, while *parallel self-defeat* occurs when the contradictory conclusions of two or more arguments are taken as the premises for $\perp$. It turns out that parallel self-defeating can cause problems if argumentation systems are not carefully defined, particularly if they include standard propositional logic.

The following example explains why serial self-defeat does not cause problems.

Example 6.6.1 Consider the following version of the argument scheme from witness testimony plus an undercutter in case the witness is incredible:

$d_w(x, \varphi)$: $\texttt{Says}(x, \varphi) \Rightarrow \varphi$
$u_w(x, \varphi)$: $\texttt{Incredible}(x) \rightarrow \neg d_w(x, \varphi)$

Now suppose that $\mathcal{K}_p$ contains $\texttt{Says}(\mathit{John}, \text{“}\texttt{Incredible}(\mathit{John})\text{”})$. Then we have

A_1: $\texttt{Says}(\mathit{John}, \text{“}\texttt{Incredible}(\mathit{John})\text{”})$
A_2: $A_1 \Rightarrow \texttt{Incredible}(\mathit{John})$
A_3: $A_2 \rightarrow \neg d_w(\mathit{John}, \text{“}\texttt{Incredible}(\mathit{John})\text{”})$

Argument A_3 is self-defeating since it undercuts itself on A_2. In both preferred and grounded semantics there is a unique extension $E = \{A_1\}$. Arguably this is the desired outcome, since suppose witness John also says something completely unrelated, say, 'the suspect stabbed the victim with a knife' if the self-defeating argument A_3 were overruled, the argument that can be constructed for 'the suspect stabbed the victim with a knife' would be justified since all its defeaters are overruled, while yet it is based on a statement of a witness who says of himself that he is incredible.

The following abstract example illustrates the problems that can be caused by parallel self-defeat.

Example 6.6.2 Let $\mathcal{R}_d = \{p \Rightarrow q;\ r \Rightarrow \neg q;\ t \Rightarrow s\}$ and $\mathcal{K} = \{p, r, t\}$ while $\mathcal{R}_s$ consists of all propositionally valid inferences. Then:

$$\begin{array}{ll}
A_1\colon p & A_2\colon A_1 \Rightarrow q \\
B_1\colon r & B_2\colon B_1 \Rightarrow \neg q \\
C_1\colon A_2, B_2 \rightarrow \bot & C_2\colon C_1 \rightarrow \neg s \\
D_1\colon t & D_2\colon D_1 \Rightarrow s
\end{array}$$

Here a problem arises since s can be any formula, so any defeasible argument unrelated to A_2 or B_2, such as D_2, can, depending on the argument ordering, be rebutted by C_2. Clearly, this is extremely harmful, since the existence of just a single case of mutual rebutting defeat, which is very common, could trivialise the system. In fact, of the semantics defined by Dung (1995) this is only a problem for grounded semantics. Since all preferred/stable extensions contain either A_2 or B_2, argument C_2 is not in any of these extensions so D_2 is in these extensions. However, if neither of A_2 and B_2 strictly defeats the other, then neither of them is in the grounded extension so that extension does not defend D_2 against C_2 and therefore does not contain D_2.

(Actually, if examples of parallel 'self-defeat' are translated into a Dung-style abstract argumentation framework, there are no abstract self-defeating arguments. Nevertheless, intuitively, this is a case of self-defeat, which is why it is discussed in this section.)

Current research on tackling these issues has made some progress. Wu (2012) proves for the special case with $\mathcal{L}$ a propositional or first-order language with a classical interpretation and with a simple argument ordering that if the set of all conclusions of an argument is required to be indirectly consistent, the above problems do not arise while all results on the rationality postulates still hold. Note that with this requirement, the argument C_1 in Example 6.6.2 cannot be constructed. Moreover, Grooters and Prakken (2016) prove for the more general case with any reasonable argument ordering that the problems can avoided by imposing two additional constraints on the construction of arguments: (1) strict rules can only be applied to classically consistent sets of formulas, and (2) strict rules cannot be chained. This also rules out C_1 in Example 6.6.2 and, moreover, rules out other problematic examples. Note that Grooters and Prakken (2016) do not adopt Wu (2012)'s constraint that the set of all conclusions of an argument should be consistent.

In conclusion, there are good reasons to believe that the two types of self-defeating arguments should be treated differently: while arguments based on parallel self-defeat should always be overruled, arguments with serial self-defeat should retain their force to prevent other arguments from being justified or defensible.

6.7 Conclusion

In this chapter we presented *ASPIC*$^+$, a framework for structured argumentation based on two ideas: that conflicts between arguments are sometimes resolved with explicit preferences, and that arguments are built with two kinds of inference rules: strict, or deductive rules, which logically entail their conclusion, and defeasible rules, which only create a presumption in favour of their conclusion. The second idea implies that *ASPIC*$^+$ does not primarily see argumentation as inconsistency handling in a given 'base' logic: conflicts between arguments may not only arise from the inconsistency of a knowledge base but also from the defeasibility of the reasoning steps in an argument.

ASPIC$^+$ is not a system but a framework for specifying systems. A main objective is to identify conditions under which instantiations of ASPIC+ satisfy logical consistency and closure properties. We first discussed *ASPIC*$^+$'s philosophical underpinnings. We then illustrated the main definitions with examples and we presented some more and less principled ways to instantiate the framework. We also briefly discussed how *ASPIC*$^+$ captures several other approaches as special cases. As we saw above, the *ASPIC*$^+$ framework can be instantiated in many different ways. We have already discussed some of these ways and their properties. We hope that in due course more 'best practices' in using *ASPIC*$^+$ will emerge.

Finally, three implementations are available online of instantiations of *ASPIC*$^+$ with domain-specific inference rules and with rule priorities:

- Mark Snaith's TOAST (http://www.arg.dundee.ac.uk/toast/);
- Wietske Visser's EPR (http://www.wietskevisser.nl/research/epr/);
- Matthew South's implementation based on a prototype by Gerard Vreeswijk (http://aspic.cossac.org/).

6.8 Exercises

In the following exercises an argument ordering is called *simple* if it holds that $A \prec B$ iff A is plausible or defeasible while B is strict and firm, and $A \approx B$ otherwise.

EXERCISE 6.8.1 Consider the following argumentation theory with:

$\mathcal{R}_s = \{p, q \rightarrow r,\ t \rightarrow \neg d_1\}$,
$R_d = \{$
  d_1: $p \Rightarrow q$,
  d_2: $s \Rightarrow t$,
  d_3: $u \Rightarrow v$,
  d_4: $v \Rightarrow \neg t\}$
$\mathcal{K}_p = \{p, s, u\}$

With orderings $\leq$ on R_d and $\leq'$ on $\mathcal{K}_p$ such that $d_2 < d_4$, $d_3 < d_2$ and $u <' s$.

1. Verify the status of r according to preferred semantics, assuming the weakest-link ordering on arguments.
2. Answer the same question assuming the last-link ordering on arguments.

EXERCISE 6.8.2 Consider an argumentation system in which $\mathcal{R}_s$ consists of all valid propositional and first-order inferences from finite sets, and with as knowledge base

$$\mathcal{K}_n = \{\forall x(Px \supset Qx)\}$$
$$\mathcal{K}_p = \{Pa, \forall x(Qx \supset Rx)\}$$

1. Construct a consistent argument A for Ra.
2. Identify $\texttt{Prem}(A)$, $\texttt{Conc}(A)$, $\texttt{Sub}(A)$, $\texttt{DefRules}(A)$ and $\texttt{TopRule}(A)$.
3. What is in terms of Definition 6.3.7 the type of this argument?

EXERCISE 6.8.3 Consider the following argumentation theory with a simple argument ordering and:

$\mathcal{R}_s$ consists of all valid inferences of propositional logic from finite sets;
$R_d = \{$
$p, q \Rightarrow r,$
$r \vee s \Rightarrow t,$
$u \Rightarrow v,$
$w \Rightarrow \neg u\}$
$\mathcal{K}_n = \{\neg(q \wedge v)\}$
$\mathcal{K}_p = \{p, q, u, w\}$ Evaluate the following questions relative to the *c-SAF* induced by this example.

1. Verify the status of t according to grounded semantics, assuming the weakest-link ordering on arguments.
2. Assume now the following preference orderings $\leq$ on $\mathcal{R}_d$ and $\leq'$ on $\mathcal{K}_p$:

   $w \Rightarrow \neg u < u \Rightarrow v$

   $q <' u$

   $w <' u$

   Verify how the answer to question (1) changes for the elitist last-link ordering.
3. Answer the same question for the elitist weakest-link ordering.

EXERCISE 6.8.4 Consider the following argumentation theory with:

$\mathcal{R}_s$ consists of all valid propositional inferences from finite sets,
$R_d = \{$
d_1: $p \Rightarrow q,$
d_2: $p, q \Rightarrow r,$
d_3: $s \Rightarrow t\}$
$\mathcal{K}_p = \{p, s, (q \wedge r) \supset \neg t\}$

With an ordering $\leq$ on R_d such that $d_3 < d_1$ and $d_2 < d_3$. Evaluate the following questions relative to the *c-SAF* induced by this example.

1. Verify the status of t and $\neg t$ according to preferred semantics, assuming the last-link ordering on arguments.

2. Specify the following for all arguments X that you constructed in your answer: $\texttt{Prem}(X)$, $\texttt{Conc}(X)$, $\texttt{Sub}(X)$, $\texttt{DefRules}(X)$, $\texttt{LastDefRules}(X)$ and $\texttt{TopRule}(X)$.

EXERCISE 6.8.5 Consider Example 6.6.1.

1. Explain why $E = \{A_1\}$ is the only grounded and preferred extension.

2. Extend the example with the argument based on John's testimony about the suspect and verify its status in grounded and preferred semantics.

EXERCISE 6.8.6 Consider the following example of a civil legal case. Assume that in a medical malpractice case, a doctor is liable for compensation if the patient was injured because of the doctor's negligence, and that if a patient is injured in a non-risky operation, this is negligence. We also have that an appendicitis operation generally is a non-risky operation but that operations on patients with bad blood circulation are generally risky. Assume finally, that a given patient was injured in an appendicitis operation and that two medical tests gave contradicting results on whether the patient had bad blood circulation. One way to represent this is with the following facts and domain-specific defeasible rules: $\mathcal{R}_s = \mathcal{K}_p = \varnothing$, $\mathcal{R}_d = \{r_1\text{-}r_6\}$ while $\mathcal{K}_n = \{f_1\text{-}f_4\}$.

r_1: *injury*, *negligence* $\Rightarrow$ *compensation*
r_2: *injury*, $\neg$ *risky operation* $\Rightarrow$ *negligence*
r_3: *appendicitis* $\Rightarrow \neg$ *riskyOperation*
r_4: *badCirculation* $\Rightarrow$ *riskyOperation*
r_5: *medicalTest1* $\Rightarrow$ *badCirculation*
r_6: *medicalTest2* $\Rightarrow \neg$ *badCirculation*

f_1: *injury*
f_2: *appendicitis*
f_3: *medicalTest1*
f_4: *medicalTest2*

1. Construct all arguments on the basis of this argumentation theory and their attack relations.

2. Specify the following for all arguments X: $\texttt{Prem}(X)$, $\texttt{Conc}(X)$, $\texttt{Sub}(X)$, $\texttt{DefRules}(X)$ and $\texttt{TopRule}(X)$.

3. Suppose that $r_3 < r_4$ and $r_5 < r_6$. Determine the defeat relations with the elitist last-link ordering.

4. Determine the grounded extension of the SAF defined by the above argumentation theory and the argument ordering induced by the preference relation of (b).

5. Determine the preferred extension(s).

6. Move f_3 and f_4 from $\mathcal{K}_n$ to $\mathcal{K}_p$ and assume also that $f_4 <' f_3$. Answer again questions (b-d) but now for the elitist weakest-link ordering.

EXERCISE 6.8.7 Give the abstract argumentation framework corresponding to Figure 6.4.

EXERCISE 6.8.8 Consider the following, equally strong defaults

1. Persons born in The Netherlands are typically Dutch.
2. Persons with a Norwegian name are typically Norwegian.
3. Persons who are Dutch or Norwegian typically like ice skating.

and the following facts:

4. Brigt Rykkje was born in the Netherlands
5. Brigt Rykkje has a Norwegian name.
6. Nobody is both Dutch and Norwegian.

Evaluate the following questions relative to the *c-SAF* induced by this example.

1. Translate this information into an argumentation theory of which $\mathcal{R}_s$ consists of all valid propositional and first-order inferences from finite sets and $\mathcal{R}_d$ consists of the defeasible inference scheme for $\leadsto$ from Section 6.4.1.
2. Assume that the argument ordering is determined by the last-link principle. We want to know whether Brigt Rykkje likes ice skating. Construct all arguments that are relevant for this proposition and determine whether the conclusion that Brigt Rykkje likes ice skating is justified in grounded semantics.
3. Answer the same question for preferred semantics.
4. Answer the same question for f-justification in preferred semantics.

EXERCISE 6.8.9 Formalise the example of Exercise 4.8.12 as an argumentation theory with domain-specific defeasible rules in a way that satisfies your intuitions about this example.

EXERCISE 6.8.10 Let S be a set of strict rules and let $Cl_{tp}(S)$ be defined as the smallest set such that:

- $S \subseteq Cl_{tp}(S)$, and
- If $s \in Cl_{tp}(S)$ and t is a transposition of s then $t \in Cl_{tp}(S)$.

Let $\mathcal{R}_s = \{p \rightarrow q; p \rightarrow r; p, r \rightarrow s\}$.

1. Determine $Cl_{tp}(R_s)$.
2. Determine whether with $Cl_{tp}(R_s)$ it holds that $\{p\} \vdash s$.
3. Determine whether with $Cl_{tp}(R_s)$ it holds that $\{-s\} \vdash -p$.

EXERCISE 6.8.11 Let $\mathcal{R}_s = \{p \rightarrow q;\ \neg q \rightarrow r;\ r \rightarrow \neg p;\ \neg r \rightarrow q;\ p \rightarrow \neg r\}$ and let $^-$ correspond to classical negation.

1. Is an argumentation theory with $\mathcal{R}_s$ closed under transposition?
2. Is an argumentation theory with $\mathcal{R}_s$ closed under contraposition?

EXERCISE 6.8.12 [12] Let $(\mathcal{L}, ^-, \mathcal{R}, n)$ be an argumentation system where:

- $\mathcal{L}$ is a language of propositional literals, composed from a set of propositional atoms $\{a, b, c, \ldots\}$ and the symbols $\neg$ and $\sim$ respectively denoting strong and weak negation (i.e., negation as failure). α is a strong literal if α is a propositional atom or of the form $\neg\beta$ where β is a propositional atom (strong negation cannot be nested). α is a wff of $\mathcal{L}$, if α is a strong literal or of the form $\sim \beta$ where β is a strong literal (weak negation cannot be nested).

[12] Adapted from S. Modgil & H. Prakken, A general account of argumentation with preferences. *Artificial Intelligence* 195 (2013): 361–397.

- $\alpha \in \overline{\beta}$ iff (1) α is of the form $\neg\beta$ or β is of the form $\neg\alpha$; or (2) β is of the form $\sim \alpha$ (i.e., for any wff α, α and $\neg\alpha$ are contradictories and α is a contrary of $\sim \alpha$).
- $\mathcal{R}_s = \{t, q \rightarrow \neg p\}, \mathcal{R}_d = \{\sim s \Rightarrow t; r \Rightarrow q; a \Rightarrow p\}$
- $n(\sim s \Rightarrow t) = d_1, n(r \Rightarrow q) = d_2, n(a \Rightarrow p) = d_3$

Furthermore, $\mathcal{K}$ is the knowledge base such that $\mathcal{K}_n = \varnothing$ and $\mathcal{K}_p = \{a, r, \neg r, \sim s\}$.

1. Construct all arguments on the basis of this argumentation theory.
2. Determine the attack relations.
3. Assume that the argument ordering $\preceq$ is defined in terms of preorderings $\leq$ on defeasible rules and $\leq'$ on ordinary premises. Assume that $r \Rightarrow q < a \Rightarrow p$ (i.e., $d_1 < d_2$) and $\neg r <' r; \neg a \approx' r; \sim s <' \neg r$. Determine the defeat relations with the elitist last link ordering.
4. Add the transpositions of $t, q \rightarrow \neg p$ to $\mathcal{R}_s$. Which new arguments, attacks and defeats are now generated?

EXERCISE 6.8.13 Consider the same language $\mathcal{L}$ as in Exercise 6.8.12 but let now $\mathcal{R}_s = \{\sim a \rightarrow b\}$, $\mathcal{R}_d = \{b \Rightarrow_{d_1} \neg c;\ \Rightarrow_{d_2} c;\ c \Rightarrow_{d_3} a\}$ (here the names of the defaults are attached to $\Rightarrow$), $\mathcal{K}_n = \varnothing$ and $\mathcal{K}_p = \{\sim a\}$. Finally, assume a partial preorder $<$ on $\mathcal{R}_d$ such that that $d_2 < d_1$ and $d_1 < d_3$.

1. Determine the arguments and their attack relations.
2. Determine which attacks succeed as defeats with the elitist last-link ordering.
3. Determine the grounded extension of the resulting abstract argumentation theory.
4. Determine the preferred extension(s) of this abstract argumentation theory.

EXERCISE 6.8.14 Consider the argumentation theory of Example 6.6.2.

1. Verify the status of argument D_2 for s in grounded semantics.
2. Verify the status of argument D_2 for s in preferred semantics.

EXERCISE 6.8.15 Consider an argumentation theory in which $\mathcal{R}_s$ consists of all valid propositional inferences from finite sets, $\mathcal{R}_d = \mathcal{K}_n = \varnothing$ and $\mathcal{K}_p =$

$$\begin{array}{l}\{\neg ab \supset \neg guilty,\\ murder \supset guilty,\\ murder,\\ \neg ab\}.\end{array}$$

Consider a variant of *ASPIC*$^+$ in which all arguments are consistent and in which strict rules cannot be chained.

1. Verify whether $guilty$ is justified according to grounded semantics, assuming a simple argument ordering.

2. Then specify a partial preorder on $\mathcal{K}_n$ such that with the elitist weakest-link argument ordering $guilty$ is justified according to grounded semantics.

3. Alternatively to (b), move one or more formulas from $\mathcal{K}_p$ to $\mathcal{K}_n$ such that $guilty$ becomes justified as a result of the change.

Chapter 7

Dynamics of argumentation

7.1 Introduction

In this chapter aspects of the dynamics of argumentation are discussed while abstracting from the procedural context in which argumentation takes place. For example, when discussing methods for extending or revising argumentation frameworks, we disregard the question whether such a change is allowed according to the rules of debate (for example, whether certain types of evidence are admissible or whether claims made earlier can be retracted). The procedural aspects of argumentation are discussed in Chapter 8.

The study of information dynamics in argumentation concerns the nature and effects of change operations on a given argumentation state. This work is motivated by several application scenario's, such as:

- Adjudication dialogues like in legal procedure, where two adversaries aim to persuade an adjudicator of the dispute (judge or jury).
- Debates in parliament or similar bodies that have to vote on proposals, where members try to persuade each other to vote for or against the various proposals.
- Any individual or group of individuals interested in a debate and wanting to evaluate it from his/her/their point of view.

In dynamic contexts, adding new arguments clearly makes sense but adding attacks only seems to make sense when these attacks involve at least one new argument. Deleting attacks makes sense when interpreted as applying preferences to decide that a given attack relation does not succeed as defeat. Finally, deleting arguments makes sense in contexts where elements of arguments can be retracted by a participant or can be rejected by an adjudicator without stating a counterargument. An example of rejection by an adjudicator is in legal dialogues, where a judge can, for example, reject a factual premise since it has not been sufficiently backed by evidence and must therefore be ignored by the rules of legal procedure.

Most current work on argumentation dynamics concerns abstract argumentation. In particular the following operations on abstract argumentation frameworks have been studied:[1] addition or deletion of (sets of) arguments (e.g. Baumann (2012); Baumann

[1] To be consistent with the literature, we will in this chapter rename the defeat relations of the reader of this course to attack relations.

and Brewka (2010); Cayrol *et al.* (2010)) and addition or deletion of (sets of) attack relations (e.g. Modgil (2006); Baroni *et al.* (2011); Bisquert *et al.* (2013)). This work then studies preservation and enforcement properties. Preservation is about the extent to which the current status of arguments is preserved under change, while enforcement concerns the extent to which desirable outcomes can or will be obtained by changing a framework.

This current work about abstract argumentation disregards the structure of arguments and the nature of their conflicts, which is a serious limitation. For example, abstract models of argumentation dynamics do not recognise that some arguments are not attackable (such as deductive arguments with certain premises) or that some attacks cannot be deleted (for example between arguments that were determined to be equally strong), or that the deletion of one argument implies the deletion of other arguments (when the deleted argument is a subargument of another), or that the deletion or addition of one attack implies the deletion or addition of other attacks (for example attacking an argument implies that all continuations of that argument with further inferences are also attacked). All this means that formal results on preservation and enforceability of outcomes are only relevant for very specific cases and do not cover many realistic situations in argumentation.

Accordingly, the purpose of this chapter is twofold:

1. to introduce the current research on the dynamics of argumentation;

2. to warn against naive work at the abstract level.

7.2 Work on enforcement properties

We first briefly discuss work on enforcement. The study of enforcement concerns contexts where new arguments and possibly new attacks involving new arguments can be added. This is motivated by applications in which one agent wants to persuade another agent. All current work on enforcement is in abstract argumentation. Baumann and Brewka (2010) define expansions of argumentation frameworks as follows.

Definition 7.2.1 [**Expansions**] An abstract argumentation framework AF' = is a *expansion* of an abstract argumentation framework $AF = (\mathcal{A}, \mathcal{C})$ iff $AF' = (\mathcal{A} \cup \mathcal{A}', \mathcal{C} \cup \mathcal{C}')$ for some nonempty $\mathcal{A}'$ disjoint from $\mathcal{A}$, such that for all A, B: if $(A, B) \in \mathcal{C}'$ then $A \in \mathcal{A}'$ or $B \in \mathcal{A}'$.

Given the definition of expansions, it can for any argument $A \in \mathcal{A}$ for a given semantics be studied whether there exist expansions in which this argument is in some or all extensions. For grounded or preferred semantics the answer is (for non-selfdefeating arguments) trivially 'yes', since one can always add unattacked attackers of any attacker of A. Note that this implicitly assumes that any argument can be attacked. This assumption is not satisfied by *ASPIC*$^+$, in which strict-and-firm arguments cannot be attacked. This again shows that the structure of arguments and the nature of attack is important when study the dynamics of argumentation.

A more interesting issue here is the degree of controversiality of a change. This could, for instance, be defined in terms of a minimality ordering on changes, comparing the number of changes needed or subset relations between changes needed to enforce different arguments. Even more interesting would be to define the degree to which a change agrees or disagrees with the arguments that the agent to be persuaded has uttered in the debate. But this requires a full modelling of the dialogical context.

The enforcement question is somewhat more interesting for enforcement of *sets* of arguments. For example, if $S \subseteq \mathcal{A}$ is not conflict-free, then clearly no extension of any expansion will contain S. The same holds for sets in which an argument A indirectly attacks an argument B in that there is an attack path from A to B of odd length.

Baumann and Brewka (2010) remark that including also deletions of arguments and attacks in the model would trivialize the enforcement problem, since one could then just delete everything and add the wanted arguments without any attacks. However, both the structure of arguments and the dialogical context are relevant here. There are dialogue systems for argumentation in which a claim or an argument's premise can be challenged and in which such a challenge can be answered with a further argument. For example:

claim p
why p
p *since* q

One way to look at this dialogue is that initially a premise argument p is stated and that after the challenge of p this premise argument is replaced with an argument $q \Rightarrow p$ where its subargument q is a new premise argument. In a dialogical context and with structured argumentation this is a meaningful and non-arbitrary constructive operation, but a the abstract level it turns into an arbitrary destructive one, deleting one argument and replacing it with two other arguments.

7.3 Work on preservation properties

The first work on preservation properties concerned so-called resolution semantics. Here the focus is on deleting attack relations as a way to express a preference of one argument over another: that an attack from A on B is deleted means that A is regarded as inferior to B so that A's attack on B does not succeed as defeat. This idea was introduced for abstract argumentation by Modgil (2006) and further developed by Baroni *et al.* (2011).

7.3.1 Abstract argumentation

Given an abstract argumentation framework $AF = (\mathcal{A}, \mathcal{C})$ (where $\mathcal{A}$ is a set of *arguments* and $\mathcal{C}$ a binary *attack relation* on $\mathcal{A}$), a resolution $AF' = (\mathcal{A}, \mathcal{C}')$ is such that $\mathcal{C}'$ replaces one or more symmetric attacks in $\mathcal{C}$ by an asymmetric relation in $\mathcal{C}'$. More precisely:

Definition 7.3.1 [**Resolutions**] An argumentation framework $AF' = (\mathcal{A}, \mathcal{C}')$ is a *resolution* of an argumentation framework $AF = (\mathcal{A}, \mathcal{C})$ iff for all arguments A and B:

1. If $(A, B) \in \mathcal{C}$ and $(B, A) \notin \mathcal{C}$ or $A = B$, then $(A, B) \in \mathcal{C}'$;
2. If $(A, B) \in \mathcal{C}$ and $(B, A) \in \mathcal{C}$ and $A \neq B$ then $(A, B) \in \mathcal{C}'$ or $(B, A) \in \mathcal{C}'$;
3. If $(A, B) \in \mathcal{C}'$ then $(A, B) \in \mathcal{C}$.

A resolution $AF' = (\mathcal{A}, \mathcal{C}')$ is *partial* if there exist $A, B \in \mathcal{A}$ such that $A \neq B$ and $(A, B) \in \mathcal{C}'$ and $(B, A) \in \mathcal{C}'$; otherwise a resolution is *full*.

Then properties can be studied concerning the relations between the original status of an argument and its status in some or all resolutions. We will discuss some of these properties for grounded and preferred semantics.

Property 7.3.2 [**Left to Right Sceptical**] If X is a justified argument of $AF = (\mathcal{A}, \mathcal{C}, \preceq)$, then X is a justified argument of every full resolution $AF' = (\mathcal{A}, \mathcal{C}, \preceq')$ of AF.

This property holds for grounded semantics but not for preferred semantics. For a counterexample for preferred semantics let $\mathcal{A} = \{A, B\}$ such that A attacks A and A and B attack each other. Then the unique preferred extension is $\{B\}$ but there exists a resolution with an empty preferred extension, namely, when the attack of B on A is deleted.

Property 7.3.3 [**Right to Left Sceptical**] If X is a justified argument of every full resolution $AF' = (\mathcal{A}, \mathcal{C}, \preceq')$ of $AF = (\mathcal{A}, \mathcal{C}, \preceq)$, then X is a justified argument of AF.

This property holds for preferred semantics but not for grounded semantics. For a counterexample for grounded semantics let $\mathcal{A} = \{A, B, C, D\}$ such that A and B attack each other, both A and B attack C and C attacks D. Then there are two full resolutions: one in which the attack of A on B is deleted and one in which the attack of B on A is deleted. The first resolution yields the grounded extension $\{B, D\}$ while the second resolution yields the grounded extension $\{A, D\}$. So D is justified in all full resolutions. However, the initial grounded extension is empty.

Other preservation properties can be formulated by replacing one or both occurrences of 'justified' with 'defensible' and/or replacing occurrences of 'all' with 'some'. For example:

Property 7.3.4 [**Left to Right Credulous to Justified**] If X is a defensible argument of $AF = (\mathcal{A}, \mathcal{C}, \preceq)$, then X is a justified argument of some full resolution $AF' = (\mathcal{A}, \mathcal{C}, \preceq')$ of AF.

This property does not hold for grounded semantics. The counterexample to *Right to Left Sceptical* also holds here.

7.3.2 Structured argumentation

When resolutions are intended to model the outcome of preference arguments, then the above-defined abstract study of resolutions has limited applicability (cf. Modgil and Prakken (2012)). Firstly, one must also account for the resolution of *asymmetric* attacks, since many argumentation formalisms, including *ASPIC*$^+$, apply preferences to deny the success of asymmetric attacks as defeats. Furthermore, some formalisms apply preferences so that *both* attacks in a symmetric attack fail to succeed as defeats. Third, sometimes resolutions of symmetric attacks are impossible; for example when two symmetrically attacking arguments are assigned equal strength.

Resolutions can also be impossible for another reason: preference relations have properties, so the addition of preferences to resolve one attack may imply further preferences and thereby make resolutions based on conflicting preferences impossible. Finally, resolutions are impossible if some attacks succeed irrespective of preferences (e.g., undercutters or contrary-underminers in *ASPIC*$^+$).

Such subtleties can only be fully appreciated in a setting where the structure of arguments and the nature of attack and the use of preference to define defeats is made explicit. To this end Modgil and Prakken (2012) study resolutions in the *ASPIC*$^+$ framework. They are interested in the case where given a $(c-)SAF$ $\Delta = (\mathcal{A}, \mathcal{C}, \preceq)$ and its defined defeat relation, what is the relationship, under different semantics, between the justified arguments of Δ and the justified arguments of $\Delta' = (\mathcal{A}, \mathcal{C}, \preceq')$, where Δ' is a resolution of Δ obtained by extending' $\preceq$ to the preference relation $\preceq'$. They assume that the preference relation on arguments is a partial preorder, that is, transitive and reflexive.

Definition 7.3.5 Let $\preceq$ be a partial preorder over a set Γ. Then $\preceq'$ *extends* $\preceq$ iff $\preceq \subseteq \preceq'$ and $\forall X, Y \in \Gamma$, $X \prec Y$ implies $X \prec' Y$.
Let $\Delta = (\mathcal{A}, \mathcal{C}, \preceq)$ be a *SAF*. Then $\Delta' = (\mathcal{A}, \mathcal{C}, \preceq')$ *preference-extends* Δ iff $\preceq'$ *extends* $\preceq$.

To motivate the definition of *extends*, recall that $\preceq$ is a partial preorder. Thus it does not suffice to define *extends* in terms of the condition $X \prec Y$ implies $X \prec' Y$ alone. To see why, suppose $X \preceq Y$ and $Y \preceq X$, which implies $X \approx Y$; that is they are effectively assigned the same strength. Hence, it might be that $\preceq'$ preserves the strict preferences in $\preceq$, but $X \not\preceq Y$ and $Y \not\preceq X$. But we certainly want to preserve the assignment of equal strength to X and Y. On the other hand, it does not suffice to define *extends* in terms of the condition $\preceq \subseteq \preceq'$ alone. This is because given only $X \preceq Y$ and so $X \prec Y$, we want that this strict preference be preserved in the extended argument ordering. However, if $X \preceq' Y$ and $Y \preceq' X$, then this strict preference would not be preserved.

It is straightforward to then show that if $(\mathcal{A}, \mathcal{C}, \preceq')$ *preference-extends* $(\mathcal{A}, \mathcal{C}, \preceq)$, and $\mathcal{D}'$ and $\mathcal{D}$ are the defeat relations respectively defined by $\preceq'$ and $\preceq$, then $\mathcal{D}' \subseteq \mathcal{D}$.

Now the notion of a preference-based resolution can be defined:

Definition 7.3.6 Let $\Delta' = (\mathcal{A}, \mathcal{C}, \preceq')$ be a SAF that *preference-extends* $\Delta = (\mathcal{A}, \mathcal{C}, \preceq)$, and let $\mathcal{D}'$ and $\mathcal{D}$ be defeat relations respectively defined by $\preceq'$ and $\preceq$. Then

- Δ' is a *preference-based resolution* of Δ iff $\mathcal{D}' \subset \mathcal{D}$.
- Δ' is a *full preference-based resolution* of Δ iff Δ' is a preference-based resolution of Δ and there exists no preference-based resolution $\Delta'' = (\mathcal{A}, \mathcal{C}, \preceq'')$ with induced defeat relations $\mathcal{D}''$ such that $\mathcal{D}'' \subset \mathcal{D}'$.

Below we will assume that the argument ordering $\preceq$ is an elitist weakest- or last-link ordering induced by partial preorders $\leq$ on $\mathcal{R}_d$ and $\leq'$ on $\mathcal{K}_p$. Moreover, we will only consider preference-based resolutions that extend $\leq$ and $\leq'$ in the sense of Definition 7.3.5.

Next the preservation properties for preference-based resolutions are restated as follows:

Property 7.3.7 **[Left to Right Sceptical]** If X is a justified argument of $\Delta = (\mathcal{A}, \mathcal{C}, \preceq)$, then X is a justified argument of every full preference-based resolution $\Delta' = (\mathcal{A}, \mathcal{C}, \preceq')$ of Δ.

Property 7.3.8 **[Right to Left Sceptical]** If X is a justified argument of every full preference-based resolution $\Delta' = (\mathcal{A}, \mathcal{C}, \preceq')$ of $\Delta = (\mathcal{A}, \mathcal{C}, \preceq)$, then X is a justified argument of Δ.

The grounded extension fails *Right to Left Sceptical*. Consider the following counterexample:

Example 7.3.9 Let $(\mathcal{L}, ^{-}, \mathcal{R}, n)$ be an argumentation system where:

- $\mathcal{L}$ is a language of propositional literals, composed from a set of propositional atoms $\{p, q, r, s, \ldots\}$ and the symbols $\neg$ and $\sim$ respectively denoting strong and weak negation (i.e., negation as failure). α is a strong literal if α is a propositional atom or of the form $\neg\beta$ where β is a propositional atom. α is a wff of $\mathcal{L}$, if α is a strong literal or of the form $\sim \beta$ where β is a strong literal.

- For any wff α, α and $\neg\alpha$ are contradictories and α is a contrary of $\sim \alpha$.

- $\mathcal{R}_s = \varnothing$, $\mathcal{R}_d = \{\neg q \Rightarrow p; \neg p \Rightarrow q; \sim p, \sim q \Rightarrow r; \sim r \Rightarrow s\}$, and $\neg q \Rightarrow p \leq \neg p \Rightarrow q$ and $\neg p \Rightarrow q \leq \neg q \Rightarrow p$ (i.e., $\neg q \Rightarrow p \approx \neg p \Rightarrow q$) and $\preceq$ = $\approx$.

- $\mathcal{K}$ is the knowledge base $\mathcal{K}_n = \varnothing$, $\mathcal{K}_p = \{\neg p, \neg q, \sim p, \sim q\}$, and $\leq'$ = $\approx$.

Figure 7.1-a) shows the induced arguments and defeats. Note the attacks on R and S are contrary attacks and so are preference independent, and since α is a contrary of $\sim \alpha$, the arguments $[\sim p]$ and $[\sim q]$ do not attack and so defeat P and Q respectively. Figures 7.1-b) and 7.1-c) show the two possible preference-based resolutions, obtained respectively by extending $\leq'$ to include $\neg q <' \neg p$ (and so $P \prec P'$, $P \prec Q$, $Q' \prec Q$) and $\neg p <' \neg q$ (and so $P' \prec P$, $Q \prec P$, $Q \prec Q'$). Argument S is in the grounded extension of both resolutions, but not in the grounded extension of Figure 7.1-a).

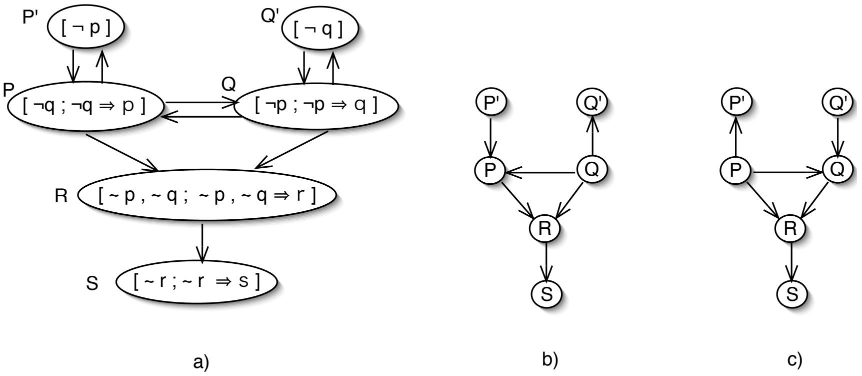


Figure 7.1: b) and c) are the two preference-based resolutions of a)

For finitary frameworks *Left to Right Sceptical* is by Modgil and Prakken (2012) proven to hold. Preferred semantics still fails *Left to Right Sceptical* but now it also fails *Right to Left Sceptical*: a counterexample is given in Modgil and Prakken (2012). We will not give it here but only remark that it is due to the fact that some preferences entail other preferences by the properties of partial preorders, so that not all resolutions that are possible in abstract resolution semantics as defined in Modgil (2006); Baroni *et al.* (2011) are possible in preference-based resolution semantics. This shows that the nature of attack and defeat is relevant when defining resolution semantics.

7.4 Exercises

EXERCISE 7.4.1 Consider an AF such that A and B defeat each other, B defeats C and C defeats A.

1. Is B justified in preferred semantics?
2. Is B justified in all full resolutions in preferred semantics?

EXERCISE 7.4.2 Consider again Exercise 4.8.11(a,b,e) from the reader *Commonsense Reasoning and Argumentation*.

1. Is D is justified in some and/or in all full resolutions in grounded semantics?
2. Is D is justified in some and/or in all full resolutions in preferred semantics?

EXERCISE 7.4.3 Give examples in *ASPIC*$^+$ of the three types of situations mentioned in the first paragraph of Section 7.3.2.

EXERCISE 7.4.4 Consider again Example 6.8.1 from the reader *Commonsense Reasoning and Argumentation*. Does the status of r change in some resolutions? Answer this question both for the elitist weakest- and for the elitist last-link ordering.

EXERCISE 7.4.5 let $\mathcal{R}_d = \mathcal{K}_n = \varnothing$, let $\mathcal{R}_s$ consist of all valid propositional inferences from finite sets and let $\mathcal{K}_p = \{p, q, \neg(p \wedge q\}$. Assume that $p <' \neg(p \wedge q)$ and $p <' q$ and apply the elitist weakest link ordering.

1. Is the argument $A = \neg(p \wedge q)$ justified in grounded semantics?
2. Is the argument $A = \neg(p \wedge q)$ justified in all full preference-based resolutions in grounded semantics?

Chapter 8

Dialogue systems for agent interaction with argumentation

This chapter is about formal dialogue systems for agent interaction with argumentation. The main focus is on so-called persuasion dialogues, in which two or more participants try to resolve a difference of opinion by arguing about the tenability of one or more claims or arguments, each trying to persuade the other participants to adopt their point of view. Dialogue systems for persuasion regulate what utterances the participants can make and under which conditions they can make them, what the effects of their utterances are on their propositional commitments, when a dialogue terminates and what the outcome of a dialogue is. Good dialogue systems regulate all this in such a way that conflicts of view can be resolved in a way that is both fair and effective.

The term ‘persuasion dialogue’ was introduced into argumentation theory by Douglas Walton (Walton, 1984) as part of his influential classification of dialogues into six types according to their goal (see also e.g. Walton and Krabbe (1995)). While *persuasion* aims to resolve a difference of opinion, *negotiation* tries to resolve a conflict of interest by reaching a deal, *information seeking* aims at transferring information, *deliberation* wants to reach a decision on a course of action, *inquiry* is aimed at “growth of knowledge and agreement” and *quarrel* is the verbal substitute of a fight. This classification is not meant to be exhaustive and leaves room for dialogues of mixed type, such as a negotiation that can shift to an embedded persuasion if the negotiating agents disagree about a relevant matter of fact.

The modern study of formal dialogue systems for persuasion probably started with two publications by Charles Hamblin (Hamblin, 1970, 1971). Initially, the topic was studied only within philosophical logic and argumentation theory. From the early nineteen nineties the study of persuasion dialogues was taken up in several fields of computer science. In Artificial Intelligence logical models of commonsense reasoning have been extended with formal models of persuasion dialogue as a way to deal with resource-bounded reasoning. In artificial intelligence & law interest in dialogue systems arose when researchers realised that legal reasoning is bound not only by the rules of logic and rational inference but also by those of fair and effective procedure. Persuasion was here seen as an appropriate model of legal procedures. Finally, in the field of multi-agent systems dialogue systems have been incorporated into models of rational agent interaction. To fulfill their own or joint goals, intelligent agents often need to interact with other agents. When they pursue joint goals, the typical modes of interaction are information seeking and deliberation and when they self-interestedly pursue their own goals, they often interact by way of negotiation. In all these cases the dialogue can shift to persuasion. For example, in information-seeking a conflict of opinion could arise on the credibility of a source of information, in deliberation the participants may disagree about likely effects of plans or actions and in negotiation they may disagree about the reasons why a proposal is in one's interest; also, in all three cases the participants may disagree about relevant factual matters.

To delineate the precise scope of this chapter, it is useful to discuss what is the subject matter of dialogue systems. According to Carlson (Carlson, 1983) dialogue systems define the principles of coherent dialogue. In his words, whereas logic defines the conditions under which a proposition is true, dialogue systems define the conditions under which an utterance is appropriate. And the leading principle here is that an utterance is appropriate if it furthers the goal of the dialogue in which it is made. So, for instance, an utterance in a persuasion should contribute to the resolution of the conflict of opinion that triggered the persuasion, and an utterance in a negotiation should contribute to reaching agreement on a reallocation of scarce resources. Thus according to Carlson the principles governing the meaning and use of utterances should not be defined at the level of individual speech acts but at the level of the dialogue in which the utterance is made. Carlson therefore proposes a game-theoretic approach to dialogues, in which speech acts are viewed as moves in a game and rules for their appropriateness are formulated as rules of the game. Virtually all work on formal dialogue systems for persuasion follows this approach and therefore the discussion in this chapter will assume a game format of dialogue systems. It should be noted that the term *dialogue system* as used in this chapter only covers the rules of the game, i.e., which moves are allowed; it does not cover principles for playing the game well, i.e., strategies and heuristics for the individual players. Of course, the latter are also important in the study of dialogue, but they will be treated as being external to dialogue systems and instead of aspects of models of dialogue participants.

This chapter is organised as follows. First in Section 8.1 an example persuasion dialogue will be presented, to give a feel for what persuasion dialogues are and to provide material for illustration and comparison in the subsequent discussions. Then in Section 8.2 a formal framework for specifying dialogue game systems is proposed, which in Section 8.3 is instantiated for persuasion dialogues (paying attention to several alternative ways to instantiate the general framework). Then in Section 8.4 two particular dialogue systems for persuasion are discussed. Exercises can be found at the end of the chapter.

8.1 An example persuasion dialogue

The following example persuasion dialogue exhibits some typical features of persuasion and will be used in this chapter to illustrate different degrees of expressiveness and strictness of the various persuasion systems.

`Paul:` My car is safe. (*making a claim*)
`Olga:` Why is your car safe? (*asking grounds for a claim*)
`Paul:` Since it has an airbag, (*offering grounds for a claim*)
`Olga:` That is true, (*conceding a claim*) but this does not make your car safe. (*stating a counterclaim*)
`Paul:` Why does that not make my care safe? (*asking grounds for a claim*)

`Olga:` Since the newspapers recently reported on airbags expanding without cause. (*stating a counterargument by providing grounds for the counterclaim*)
`Paul:` Yes, that is what the newspapers say (*conceding a claim*) but that does not prove anything, since newspaper reports are very unreliable sources of technological information. (*undercutting a counterargument*)
`Olga:` Still your car is still not safe, since its maximum speed is very high. (*alternative counterargument*)
`Paul:` OK, I was wrong that my car is safe.

This dialogue illustrates several features of persuasion dialogues.

- Participants in a persuasion dialogue not only exchange arguments and counterarguments but also express various propositional attitudes, such as claiming, challenging, conceding or retracting a proposition.
- As for arguments and counterarguments it illustrates the following features.
  - An argument is sometimes attacked by constructing an argument for the opposite conclusion (as in Olga's two counterarguments) but sometimes by saying that in the given circumstances the premises of the argument do not support its conclusion (as in Paul's counterargument). This is the distinction between rebutting and undercutting counterarguments.
  - Counterarguments are sometimes stated at once (as in Paul's undercutter and Olga's last move) and are sometimes introduced by making a counterclaim (as in Olga's second and third move).
  - Natural-language arguments sometimes leave elements implicit. For example, Paul's second move arguably leaves a commonsense generalisation 'Cars with airbags usually are safe' implicit.
- As for the structure of dialogues, the example illustrates the following features.
  - The participants may return to earlier choices and move alternative replies: in her last move Olga states an alternative counterargument after she sees that Paul had a strong counterattack on her first counterargument. Note that she could also have moved the alternative counterargument immediately after her first, to leave Paul with two attacks to counter.
  - The participants may postpone their replies, sometimes even indefinitely: by providing her second argument why Paul's car is not safe, Olga postpones her reply to Paul's counterattack on her first argument for this claim; if Paul fails to successfully attack her second argument, such a reply might become superfluous.

8.2 Elements of dialogue systems

In this section a formal specification is proposed of the common elements of dialogue systems. To summarise, dialogue systems have a *dialogue goal* and at least two *participants*, who can have various *roles*. Dialogue systems have two languages, a *topic language* and a *communication language*. Sometimes, dialogues take place in a *context* of fixed and undisputable knowledge. Typical examples of contexts are the relevant laws in a legal dispute or a system description in a dialogue about a diagnostic problem. The heart of a dialogue system is formed by a *protocol*, specifying the allowed moves at each point in a dialogue, the *effect rules*, specifying the effects of utterances on the participants' commitments, and the *outcome rules*, defining the outcome of a dialogue. Two kinds of protocol rules are sometimes separately defined, viz. *turntaking* and *termination* rules.

Let us now specify these elements more formally. In the rest of this chapter this specification will be used when describing systems from the literature; in consequence, their appearance in this text may differ from their original presentation. As for notation, the complement $\overline{\varphi}$ of a formula φ is $\neg\varphi$ if φ is a positive formula and ψ if φ is a negative formula $\neg\psi$. (Note that the argument games of Chapter 5 are a special case of the following definitions).

Definition 8.2.1 (Dialogue systems) A *dialogue system* is a tuple of the following elements.

- A *topic language* $\mathcal{L}_t$, closed under classical negation.
- A *communication language* $\mathcal{L}_c$, consisting of a set of *speech acts* with a *content*.

  The set of *dialogues*, denoted by $M^{\leq\infty}$, is the set of all sequences from $\mathcal{L}_c$, and the set of *finite dialogues*, denoted by $M^{<\infty}$, is the set of all finite sequences from $\mathcal{L}_c$. For any dialogue $d = m_1, \ldots, m_n, \ldots$, the subsequence $m_1, \ldots, m_i$ is denoted with d_i.
- A *dialogue purpose*.
- A set $\mathcal{A}$ of *participants* (or 'players') and a set $\mathcal{R}$ of *roles*, defined as disjoint subsets of $\mathcal{A}$. A participant a may or may not have a, possibly inconsistent, *belief base* $\Sigma_a \subseteq \mathcal{L}_t$, which may or may not change during a dialogue. Furthermore, each participant has a, possibly empty set of *commitments* $C_a \subseteq \mathcal{L}_t$, which usually changes during a dialogue.
- A *context* $K \subseteq \mathcal{L}_t$, containing the knowledge that is presupposed and must be respected during a dialogue. The context is assumed consistent and remains the same throughout a dialogue.
- A *logic* L for $\mathcal{L}_t$, which may or may not be monotonic and which may or may not be argument-based.
- A set of *effect rules* C for $\mathcal{L}_c$, specifying for each utterance $\varphi \in \mathcal{L}_c$ its effects on the commitments of the participants. These rules are specified as functions
  - $C_a : M^{<\infty} \longrightarrow Pow(\mathcal{L}_t)$
- A *protocol* Pr for $\mathcal{L}_c$, specifying the allowed (or 'legal') moves at each stage of a dialogue. Formally, A *protocol* on $\mathcal{L}_c$ is a function Pr with domain the context plus a nonempty subset D of $M^{<\infty}$ taking subsets of $\mathcal{L}_c$ as values. That is:
  - $Pr : Pow(\mathcal{L}_t) \times D \longrightarrow Pow(\mathcal{L}_c)$

such that $D \subseteq M^{<\infty}$. The elements of D are called the *legal finite dialogues*. The elements of $Pr(K, d)$ are called the moves allowed after d given K. If d is a legal dialogue and $Pr(K, d) = \varnothing$, then d is said to be a *terminated* dialogue. Pr must satisfy the following condition: for all finite dialogues d and moves m, $d \in D$ and $m \in Pr(K, d)$ iff $d, m \in D$.

It is useful (although not strictly necessary) to explicitly distinguish elements of a protocol that regulate turntaking and termination:

  - A *turntaking* function is a function $T : D \times Pow(\mathcal{L}_t) \longrightarrow Pow(\mathcal{A})$. A *turn* of a dialogue is defined as a maximal sequence of moves in the dialogue in which the same player is to move. Note that T can designate more than one player as to-move next.
  - *Termination* is above defined as the case where no move is legal. Accordingly, an explicit definition of termination should specify the conditions under which Pr returns the empty set.

- *Outcome rules* O^K, defining the outcome of a dialogue given a context. For instance, in negotiation the outcome is an allocation of resources, in deliberation it is a decision on a course of action, and in persuasion dialogue it is a winner and a loser of the persuasion dialogue. The outcome must be defined for terminated dialogues and may be defined for nonterminated ones; in the latter case the outcome rules capture an ‘anytime’ outcome notion.

Note that no relations are assumed between a participant’s commitments and belief base. Commitments are an agent’s publicly declared points of view about a proposition, which may or may not agree or coincide with the agent’s internal beliefs. For instance, an accused in a criminal trial may very well publicly defend his innocence while he knows he is guilty.

Definition 8.2.2 (Some protocol types)

- A protocol has a *public semantics* iff the set of legal moves is always independent from the agents’ belief bases.

- A protocol is *context-independent* if the set of legal moves and the outcome is always independent of the context, so if $Pr(K, d) = Pr(\varnothing, d)$ and $O^K(d) = O^{\varnothing}(d)$ for all K and d. For context-independent protocols the context will be omitted as an argument of Pr.

- A protocol Pr is *fully deterministic* if Pr always returns a singleton or the empty set. It is *deterministic in* $\mathcal{L}_c$ if the set of moves returned by Pr at most differ in their content but not in their speech act type.

- A protocol is *unique-move* if the turn shifts after each move; it is *multiple-move* otherwise.

Paul and Olga (ct’d): The protocol in our running example clearly is multiple-move.

As explained in the introduction, participants in a dialogue can have strategies and heuristics for playing the dialogue, given their individual dialogue goal. The notion of a *strategy* for a participant a can be defined in the game-theoretical sense, as a function from the set of all finite legal dialogues in which a is to move into $\mathcal{L}_c$. A strategy for a is a *winning strategy* if in every dialogue played in accord with the strategy a realises his dialogue goal (for instance, winning in persuasion). *Heuristics* generalise strategies in two ways: they may leave the choice for some dialogues undefined and they may specify more than one move as a choice option. More formally:

Definition 8.2.3 Let D_a, a subset of D, be the set of all dialogues where a is to move, and let D'_a be a subset of D_a. Then a strategy and a heuristic for a are defined as functions s_a and h_a as follows.

- $s_a : D_a \longrightarrow \mathcal{L}_c$
- $h_a : D'_a \longrightarrow Pow(\mathcal{L}_c)$

8.3 Persuasion

Let us now become more precise about persuasion. In Walton and Krabbe (1995) persuasion dialogues are defined as dialogues where the goal of the dialogue is to resolve a conflict of points of view between at least two participants by verbal means. A *point of view* with respect to a proposition can be positive (for), negative (against) or doubtful. The participant's individual aim is to persuade the other participant(s) to take over its point of view. According to Walton & Krabbe a conflict of points of view is resolved if all parties share the same point of view on the proposition that is the topic of the conflict.

Walton & Krabbe distinguish *disputes* as a subtype of persuasion dialogues where two parties disagree about a single proposition φ, such that at the start of the dialogue one party has a positive (φ) and the other party a negative ($\neg\varphi$) point of view towards the proposition. Walton & Krabbe then extend this notion to *conflicts of contrary opinions*, where the participants have a positive point of view on, respectively, φ and ψ such that $\models \neg(\varphi \wedge \psi)$.

8.3.1 Defining persuasion

Dialogue systems for persuasion can be formally defined as a particular class of instantiations of the general framework.

Definition 8.3.1 (dialogue systems for persuasion) A *dialogue system for persuasion* is defined as any dialogue system with at least the following instantiations of Definition 8.2.1.

- The *dialogue purpose* is resolution of a conflict of opinion about one or more propositions, called the *topics* $T \subseteq \mathcal{L}_t$. This dialogue purpose gives rise to the following participant roles and outcome rules.
- The participants can have the following *roles*. To start with, $prop(t) \subseteq \mathcal{A}$, the *proponents* of topic t, is the (nonempty) set of all participants with a positive point of view towards t. Likewise, $opp(t) \subseteq \mathcal{A}$, the *opponents* of t, is the (nonempty) set of all participants with a doubtful point of view toward a topic t. Together, the proponents and opponents of t are called the *adversaries* with respect to t. For any t, the sets $prop(t)$ and $opp(t)$ are disjoint but do not necessarily jointly

exhaust $\mathcal{A}$. The remaining participants, if any, are the *third parties* with respect to t, assumed to be neutral towards t.

Note that this allows that a participant is a proponent of both t and $\neg t$ or has a positive attitude towards t and a doubtful attitude towards a topic t' that is logically equivalent to t. Since protocols can deal with such situations in various ways, this should not be excluded by definition.

- The *Outcome rules* of systems for persuasion dialogues define for a dialogue d, context K and topic t the *winners* and *losers* of d with respect to topic t. More precisely, O consists of two partial functions w and l:

  - $w : D \times Pow(\mathcal{L}_t) \times \mathcal{L}_t \longrightarrow Pow(\mathcal{A})$
  - $l : D \times Pow(\mathcal{L}_t) \times \mathcal{L}_t \longrightarrow Pow(\mathcal{A})$

  such that they are defined at least for all terminated dialogues but only for those t that are a topic of d. These functions will be written as $w_t^K(d)$ and $l_t^K(d)$ or, if there is no danger for confusion, as $w_t(d)$ and $l_t(d)$. They further satisfy the following conditions for arbitrary but fixed context K:

  - $w_t(d) \cap l_t(d) = \varnothing$
  - $w_t(d) = \varnothing$ iff $l_t(d) = \varnothing$
  - if $\mid \mathcal{A} \mid = 2$, then $w_t(d)$ and $l_t(d)$ are at most singletons

- Next, to make sense of the notions of proponent and opponent, their commitments at the start of a dialogue should not conflict with their points of view.

  - If $a \in prop(t)$ then $\overline{t} \notin C_a(\varnothing)$
  - If $a \in opp(t)$ then $t \notin C_a(\varnothing)$

- Finally, in persuasion at most one side in a dialogue gives up, i.e.,

  - $w_t(d) \subseteq prop(t)$ or $w_t(d) \subseteq opp(t)$; and
  - If $a \in w_t(d)$ then
    - if $a \in prop(t)$ then $t \in C_a(d)$
    - if $a \in opp(t)$ then $t \notin C_a(d)$

  These conditions ensure that a winner did not change its point of view. Note that the only-ifs of the two latter winning conditions do not hold in general. This will be explained further below when the distinction between pure persuasion and conflict resolution is made. Note also that these conditions make that two-person persuasion dialogues are zero-sum games. Perhaps this is the main feature that sets persuasion apart from information seeking, deliberation and inquiry.

Some further distinctions can be made. With respect to outcomes, a distinction can be made between so-called *pure persuasion* and *conflict resolution*. The outcome of pure persuasion dialogues is fully determined by the participants' points of view and commitments:

Definition 8.3.2 (types of persuasion systems)

- A dialogue system is for *pure persuasion* iff for any terminated dialogue d it holds that $a \in w_t(d)$ iff
    - either $a \in prop(t)$ and $t \in C_{a'}(d)$ for all $a' \in prop(d) \cup opp(d)$
    - or $a \in opp(t)$ and $t \notin C_{a'}(d)$ for all $a' \in prop(d) \cup opp(d)$
- Otherwise, it is for *conflict resolution*.

In addition, pure persuasion dialogues are assumed to terminate as soon as the right-hand-side conjuncts of one of these two winning conditions hold.

Paul and Olga (ct'd): In our running example, if the dialogue is regulated by a protocol for pure persuasion, it terminates after Paul's retraction.

In conflict resolution dialogues the outcome is not fully determined by the participant's points of view and commitments. In other words, in such dialogues it is possible that, for instance, a proponent of φ loses the dialogue about φ even if at termination he is still committed to φ. A typical example is legal procedure, where a third party can determine the outcome of the case. For instance, a crime suspect can be convicted even if he maintains his innocence throughout the case.

If the win and loss functions are defined on all legal dialogues instead of on terminated dialogues only, then another distinction can be made: a protocol is *immediate-response* if the turn shifts just in case the speaker is the 'current' winner and if it then shifts to a 'current' loser.

8.3.2 Common elements of most persuasion systems

As for the communication language and commitment rules, some common elements can be found throughout the literature. We list the most common speech acts, with their informal meaning and the various ways they are named in the literature.[1]

- *claim* φ (assert, statement, ...). The speaker asserts that φ is the case.
- *why* φ (challenge, deny, question, ...) The speaker challenges that φ is the case and asks for reasons why it would be the case.
- *concede* φ (accept, admit, ...). The speaker admits that φ is the case.
- *retract* φ (withdraw, no commitment, ..) The speaker declares that he is not committed (any more) to φ. Retractions are 'really' retractions if the speaker is committed to the retracted proposition, otherwise it is a mere declaration of non-commitment (for example, in reply to a question).
- φ *since* S (argue, argument, ...) The speaker provides reasons why φ is the case. Some protocols do not have this move but require instead that reasons be provided by a *claim* φ or *claim* S move in reply to a *why* ψ move (where S is a set of propositions). Also, in some systems the reasons provided for φ can have structure, for example, of a proof three or a deduction.

[1] To make this chapter more uniform, the present terminology will be used even if the original publication of a system uses different terms.

- *question* φ (...) The speaker asks another participant's opinion on whether φ is the case.

Paul and Olga (ct'd): In this communication language our example from Section 8.1 can be more formally displayed as follows:

P_1: *claim* safe
O_2: *why* safe
P_3: safe *since* airbag
O_4: *concede* airbag
O_5: *claim* $\neg$ safe
P_6: *why* $\neg$ safe
O_7: $\neg$ safe *since* newspaper: "explode"
P_8: *concede* newspaper: "explode"
P_9: so what *since* $\neg$ newspapers reliable
O_{10}: $\neg$ safe *since* high max. speed
P_{11}: *retract* safe

As for the commitment rules, the following ones seem to be uncontroversial and can be found throughout the literature. (Below pl denotes the speaker of the move and s denotes the speech act performed in the move; effects on the other parties' commitments are only specified when a change is effected.)

- If $s(m) = claim(\varphi)$ then $C_{pl}(d, m) = C_{pl}(d) \cup \{\varphi\}$
- If $s(m) = why(\varphi)$ then $C_{pl}(d, m) = C_{pl}(d)$
- If $s(m) = concede(\varphi)$ then $C_{pl}(d, m) = C_{pl}(d) \cup \{\varphi\}$
- If $s(m) = retract(\varphi)$ then $C_{pl}(d, m) = C_{pl}(d) \ - \{\varphi\}$
- If $s(m) = \varphi$ *since* S then $C_{pl}(d, m) \supseteq C_{pl}(d) \cup S$

The rule for *since* uses $\supseteq$ since such a move may commit to more than just the premises of the moved argument. For instance, in Prakken (2005) the move also commits to φ, since arguments can also be moved as counterarguments instead of as replies to challenges of a claim. And in some systems that allow incomplete arguments, such as Walton and Krabbe (1995), the move also commits the speaker to the material implication $S \rightarrow \varphi$.

Paul and Olga (ct'd): According to these rules, the commitment sets of Paul and Olga at the end of the example dialogue are

- $C_P(d_{11}) \supseteq \{$airbag, newspaper: "explode", $\neg$ newspapers reliable$\}$
- $C_O(d_{11}) \supseteq \{\neg$ safe, airbag, newspaper: "explode", high max. speed$\}$

8.3.3 Further instantiations: some features, design choices and issues

In this section some further general features of persuasion systems are discussed, as well as some design choices and related issues.

Table 8.1: Locutions and typical replies

Locutions	Replies
claim φ	*why* φ, *claim* $\overline{\varphi}$, *concede* φ
why φ	φ *since* S (alternatively: *claim* S), *retract* φ
concede φ	
retract φ	
φ *since* S	*why* ψ $(\psi \in S)$, *concede* ψ $(\psi \in S)$, φ' *since* S
question φ	*claim* φ, *claim* $\overline{\varphi}$, *retract* φ

Dialectical obligations

Sometimes the expectancies created by commitments are called "(dialectical) obligations". For instance, in some sense committing oneself to a proposition requires the speaker to support the proposition with an argument when it is challenged or else retract it. However, this can be called an obligation only in a loose sense. Some protocols allow that under certain conditions a challenge can be ignored, such as when an answer would be irrelevant, or when an answer can be postponed since it may become irrelevant because of some other way of continuing the dialogue. Strictly speaking the only dialectical obligation that a participant has is making an allowed move when it is one's turn.

On the other hand, it still seems useful to systematise the loose sense of dialectical obligation. One way in which this can be done is by listing the typical replies to speech acts. Table 8.1 lists the typical replies of the common speech acts listed above. This table more or less sums up the 'dialectical obligations' imposed by persuasion systems in the literature (but individual systems may deviate).

Paul and Olga (ct'd): In terms of this table our running example can now be displayed as in Figure 8.1, where the boxes stand for moves and the links for reply relations.

A table like the above one induces another distinction between dialogue protocols.

Definition 8.3.3 A dialogue protocol is *unique-reply* if at most one reply to a move is allowed throughout a dialogue; otherwise it is *multiple-reply*.

Of course, this distinction can be made fully precise only for systems that formally incorporate the notion of replies.

Paul and Olga (ct'd): The protocol governing our running example is multiple-reply, as illustrated by the various branches in Figure 8.1.

Types of protocol rules

According to their subject matter, several types of protocol rules can be distinguished. Some rules regulate a participant's *consistency*. This can be about *dialogical* consistency, such as a rule that each move must leave the speaker's commitments consistent or a rule that upon demand a speaker must resolve such an inconsistency. Or it can be about a participant's *internal* consistency, such as the use of so-called assertion and acceptance attitudes (see Sections 8.3.3 and 8.4.1 below). For instance, a protocol rule could say that a participant may claim or accept a proposition only if his belief base contains a justified argument for the claim.

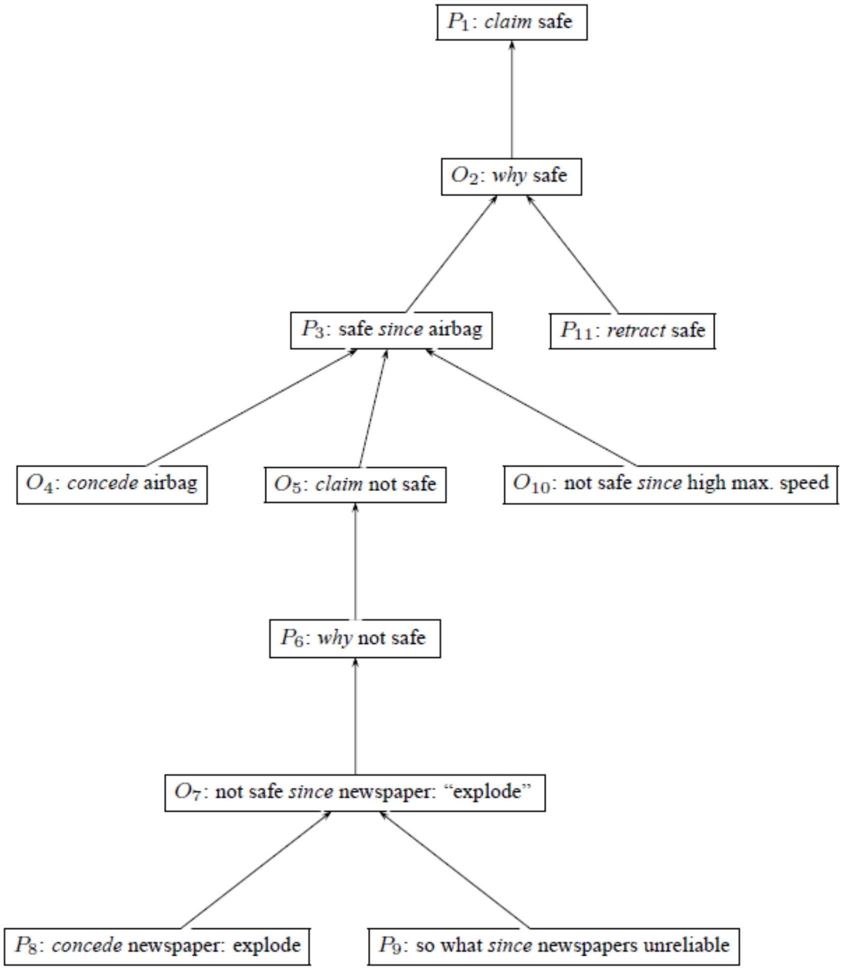

P_1: *claim* safe
O_2: *why* safe
P_3: safe *since* airbag
P_{11}: *retract* safe
O_4: *concede* airbag
O_5: *claim* not safe
O_{10}: not safe *since* high max. speed
P_6: *why* not safe
O_7: not safe *since* newspaper: "explode"
P_8: *concede* newspaper: explode
P_9: so what *since* newspapers unreliable

Figure 8.1: Reply structure of the example dialogue.

Other rules are about *dialogical coherence*, such as the rules that require a non-initial move to be an appropriate reply to some earlier move (see e.g. the table above).

Yet other rules are about the *dialogical structure*, such as the termination rules and the rules that make the protocol a unique- or multiple move protocol, a unique- or -multiple reply protocol, or an immediate- or non-immediate-response protocol.

Assertion and acceptance attitudes

Sometimes so-called 'assertion and acceptance attitudes' are incorporated into persuasion protocols, which specify how an agent must choose between various otherwise legal moves given the information that the agent has available. We discuss the attitudes defined in Parsons *et al.* (2003), generalising them to any argument-based logic. In particular, we define them relative to an implicitly assumed argumentation theory AT as defined in Chapter 6, assuming that each argument has a conclusion, and also assuming a preference ordering on arguments. The idea is that AT contains all arguments that can be constructed on the basis of the information with which an agent reasons internally.

Definition 8.3.4 (Assertion and acceptance attitudes) An agent can have one of the following three *assertion attitudes*.

- A *confident* agent can assert any proposition for which he can construct an argument.
- A *careful* agent can assert any proposition p for which he can construct an argument and cannot construct a stronger argument for $-p$[2].
- A *thoughtful* agent can assert any proposition for which he can construct a justified argument.

An agent can have one of the following three *acceptance attitudes*.

- A *credulous* agent can accept any proposition for which he can construct an argument.
- A *cautious* agent can accept any proposition p for which he can construct an argument and cannot construct a stronger argument for $-p$.
- A *skeptical* agent can accept any proposition for which he can construct a justified argument.

It can be debated whether such attitudes must be part of a protocol or of a participant's heuristics. According to one approach, a dialogue protocol should only enforce coherence of dialogues; according to another approach, it should also enforce rationality of the agents engaged in a dialogue. The second approach allows protocol rules to refer to an agent's internal belief base and therefore such protocols do not have a public semantics (in the sense defined above in Section 8.2). The first approach does not allow such protocol rules and instead studies assertion and acceptance attitudes as an aspect of dialogical behaviour of agents.

Roles of commitments

Commitments can serve several purposes in dialogue systems (though particular systems may not use all of them). One role is in enforcing a participant's dialogical consistency, for instance, by requiring him to keep his commitments consistent at all times or to make them consistent upon demand. Another role is to enlarge the hearer's means to construct arguments. For instance, in Parsons *et al.* (2003)'s use of assertion and acceptance attitudes, they are applied relative to the agents' internal beliefs plus the other participant's commitments (see further Section 8.4.1 below). A third role of commitments is to determine termination and outcome of a dialogue, such as in the above definition of pure persuasion. For example, in two-party pure persuasion the proponent wins as soon as the opponent concedes his main claim while the opponent wins as soon as the proponent has retracted his main claim. Finally, commitments can determine certain 'dialectical obligations', as in a protocol rule that a participant's commitments must be consistent, or in a protocol rule that a commitment must be supported with an argument when it is challenged.

[2] Here $-p$ is a contradictory of p in the sense of Definition 6.3.1

The role of the logic

The logic of most philosophical persuasion-dialogue systems is monotonic (usually standard propositional logic), while of most AI & Law and MAS-systems it is nonmonotonic. The logic of a persuasion-dialogue system can serve several purposes (though again particular systems may not use all of them). Firstly, it can be used in determining consistency of a participant's commitments. For this purpose a monotonic logic must be used. Secondly, it can be used to determine whether the reasons given by a participant for a challenged proposition indeed imply the proposition. When the logic is monotonic, the sense of 'imply' is obvious; when the logic is nonmonotonic, 'imply' means 'being an argument' in argument-based logics and (roughly) 'being a nonmonotonic consequence from the premises alone' in other nonmonotonic logics. Not all protocols require the reasons to be 'valid' in these senses. For instance, Walton and Krabbe (1995) allow the moving of incomplete arguments (but this still commits the speaker to the material implication *premises* $\rightarrow$ *conclusion*).

Note that this second use of a nonmonotonic logic does not yet exploit the nonmonotonic aspects of the logic. In argument-based terms, it only focuses on how arguments can be constructed, not on how they can be attacked by counterarguments. This is different in a third use of the logic, viz. to determine whether a participant respects his assertion or acceptance attitude: as we have just seen, most of these attitudes are defined in terms of counterarguments and/or defeasible consequence.

However, even if the full power of a nonmonotonic logic is used, it is still possible to distinguish between *internal* and *external* use of the logic. In Parsons *et al.* (2003) the nonmonotonic aspects of their (argument-based) logic are only used in verifying compliance with the assertion and acceptance attitudes; as we will see in Section 8.4.1, no other protocol rule refers to the notion of a counterargument. In particular, there is no rule allowing the attack of a moved argument by a counterargument. Also, the logic is not used in defining the outcome of a dialogue. Consequently, (if the attitudes are regarded as heuristics and therefore external to a dialogue system), in these systems defeasible argumentation takes place only *within* an agent and not *between* agents. By contrast, in the system of Prakken (2005)(See Section 8.4.2) the moving of counterarguments in dialogues is allowed.

One external use of argumentation logics is to formulate dialogical notions of soundness and completeness. For example:

- A protocol is *sound* if whenever at termination p is accepted, p is justified by the participants' joint knowledge bases.

- A protocol is *weakly* complete if whenever p is justified by the participants' joint knowledge bases, there is a legal dialogue at which at termination p is accepted.

- A protocol is *strongly* complete if whenever p is justified by the participants' joint knowledge bases, all legal dialogues terminate with acceptance of p.

Similar notions can be defined relative to the joint theory constructed during a dialogue, while the notions can also be made conditional on particular agent strategies and heuristics.

8.4 Two systems

To illustrate the general discussion and some of the main design options, now two persuasion protocols will be discussed and applied to our running example.

8.4.1 Parsons, Wooldridge & Amgoud (2003)

In a series of papers Parsons, Wooldridge & Amgoud have developed an approach to specify dialogue systems for various types of dialogues. We base our discussion on Parsons *et al.* (2003), focusing on their system for persuasion dialogue.

The system is for dialogues between two players called White (W) and Black (B) on a single topic. The player who starts a dialogue is its proponent and the other player must, depending on her acceptance attitude, declare at her first move whether she is negative or doubtful towards the topic or wants to concede it. Dialogues have no context but the participants have their own, possibly inconsistent belief base Σ. The players are assumed to adopt an assertion and an acceptance attitude, which they must respect throughout the dialogue. The attitudes are defined relative to their internal belief base (which remains constant throughout a dialogue) plus the commitment set of the other player (which may vary during a dialogue). The communication language $\mathcal{L}_c$ consists of claims, challenges, and concessions; it has no explicit reply structure but the protocol largely conforms to Table 8.1. Claims can concern both individual propositions and sets of propositions.

The logic of $\mathcal{L}_t$ is in fact a special case of the ASPIC framework of Chapter 6 (as shown by Modgil and Prakken (2013)). The language $\mathcal{L}_t$ is that of propositional logic. Arguments are classical proofs from consistent premises, which in ASPIC can be modelled by having only strict inference rules, namely, all valid propositional inferences. Arguments can be attacked by undermining them. Defeat relations between counterarguments are defined in terms of a priority relation on the premises of both the attacking and the attacked argument, applying the weakest-link principle. Defeasible inference is then defined with grounded semantics. PWA formally define the logic as follows:

Definition 8.4.1 (PWA argumentation logic) Let Σ be a finite set of propositional formulas ordered by a total preference ordering $\preceq$. This ordering induces a partitioning of Σ into sets $\Sigma_1, \ldots, \Sigma_n$ such that for all $p_i \in \Sigma_i$ and $p_j \in \Sigma_j$ such that $i < j$ we have that $p_j \prec p_i$, that is, p_i is *preferred over* p_j. The *preference level* of a nonempty subset H of Σ, written as $Level(H)$, is the number of the highest numbered Σ_n which has a member in H.

An *argument* is a pair $A = (H, h)$ where $H \subseteq \Sigma$ and $h \in \mathcal{L}_t$ such that:

1. H is consistent; and
2. $H \vdash h$; and
3. no proper subset of H satisfies (1) and (2).

H is called the *support* of A, written $H = Support(A)$ and h is the *conclusion* of A, written as $h = Conc(A)$.

An argument A_1 *defeats* an argument A_2 if $Conc(A_1) \vdash -h$ for some $h \in Support(A_2)$ and $Level(Support(A_2)) \not< Level(Support(A_1))$.

The *dialectical status* of arguments is defined with grounded semantics.

In dialogues, arguments cannot be moved as such but only implicitly as *claim S* replies to challenges of another claim φ, such that S is consistent and $S \vdash \varphi$. The logic is used to verify this condition and whether the players comply with their assertion and acceptance attitudes. The logic is not used externally. Finally, the commitment rules are standard and commitments are only used to enlarge the player's belief base with the other player's commitments; they are not used to constrain move legality or to define the dialogue's outcome.

The use of preferences involves some subtleties when applied to verify an assertion or acceptance attitude. As noted above, at any stage in a dialogue an agent a must reason with his own belief base Σ_a plus the commitments $C_{\overline{a}}(d)$ that the other party has in d. So W must define a total ordering on $\Sigma_W \cup C_B(d)$ while B must define a total ordering on $\Sigma_B \cup C_W(d)$. In practice these orderings may well be different but Parsons *et al.* (2003) still assume that the players agree on the ordering. This may be justified by regarding the ordering on which the players agree as composed from their individual orderings. Several ways exist to define an overall preference ordering in terms of individual orderings (for example, p_i is overall preferred to p_j just in case both players prefer p_i to p_j, otherwise p_i and p_j are equal) but below we will abstract from such ways and simply assume that there is a unique ordering on which the agents agree.

We now present the formal definition of the persuasion protocol, which in fact defines a state transition diagram.

Definition 8.4.2 (PWA persuasion protocol) A move is legal iff it does not repeat a move of the same player, and satisfies the following procedure:

1. W claims φ (assuming W's assertion attitude allows it).
2. B concedes φ if its acceptance attitude allows, if not B claims $-\varphi$ if its assertion attitude allows it, or otherwise challenges φ.
3. If B claims $-\varphi$, then goto 2 with the roles of the players reversed and $-\varphi$ in place of φ.
4. If B has challenged, then:
   (a) W claims S, an argument for φ;
   (b) Goto 2 for each $s \in S$ in turn.
5. B concedes φ if its acceptance attitude allows, or the dialogue terminates.

Dialogues *terminate* as specified in condition 5, or when the move required by the procedure cannot be made, or when the player-to-move has conceded all claims made by the hearer.

No explicit win and loss functions are defined, but the possible outcomes are defined in terms of the propositions claimed by one player and conceded by the other.

To comment on this protocol, note first that in (4b) it is ambiguous in the case where S contains more than one premise, since it is unclear whether the turn shifts as soon as the first premise has been replied to or not. In the latter case, the protocol is multi-move, since a player may reply to each premise in turn. However, for simplicity we will below assume that the turn shifts after the first reply to a *claim S* move; in this interpretation the protocol is unique move, except that after one premise is conceded, the next premise may immediately be replied to. Also, in both interpretations the protocol is unique-reply except that each element of a *claim* S move can be separately challenged or conceded. The protocol is deterministic in $\mathcal{L}_c$ but not fully deterministic, since if a player can construct more than one argument for a challenged claim, he has a choice which argument to play. Finally, the semantics of the protocol is not public, since agents have to comply with their assertion and acceptance attitudes, and these are partly defined in terms of their internal beliefs.

Let us first consider some simple dialogues that fit this protocol.

Example 8.4.3 First, let $\Sigma_W = \{p\}$ and $\Sigma_B = \varnothing$. Then the only legal dialogue is:

- W_1: *claim* p, B_1: *concede* p.

B_1 is B's only legal move, whatever its acceptance attitude, since after W_1, B must reason from $\Sigma_B \cup C_W(d_1) = \{p\}$ so that B can construct the trivial argument $(\{p\}, p)$. Here the dialogue terminates.

This example illustrates that the fact that the players must reason with the commitments of the other player makes that they can learn from each other. However, the following example illustrates that the same mechanism sometimes makes them learn too easily.

Example 8.4.4 Assume $\Sigma_W = \{q, q \supset p\}$ and $\Sigma_B = \{\neg q\}$, where all formulas are of the same preference level.

- W_1: *claim* p.

Now whatever her acceptance attitude, B has to concede p since she can construct the trivial argument $(\{p\}, p)$ for p while she can construct no argument for $\neg p$. Yet B has a defeater for W's only argument for p, namely, $(\{\neg q\}, \neg q)$, which defeats $(\{q, q \supset p\}, p)$. So even though p is not justified on the basis of the agents' joint knowledge, W_1 can win a dialogue about p.

This example thus illustrates that if the players have to reason with the other player's commitments, one player can sometimes 'force' an opinion onto the other player by simply making a claim. A possible solution to this problem is to restrict the information with which agent reason to their internal belief bases plus their own commitments. The following example illustrates another reason why this may be better.

Example 8.4.5 Consider next $\Sigma_W = \{q, q \supset p\}$ and $\Sigma_B = \{\neg p\}$, where q and $q \supset p$ are preferred over $\neg p$. Let W be thoughtful and skeptical and B careful. Then:

- W_1: *claim* p.

Since B must now reason with p, the continuation depends on the preference level of p. In fact, the protocol turns out to be problematic here. Since the players agree on the preference ordering, it seems reasonable to give p the same level as the level of the support of the strongest argument that can be constructed for p. However, the problem is that at this point in the dialogue B does not know which arguments W can construct for p. Let us sidestep this problem for the moment and let us first assume that p is preferred over $\neg p$. Then B must concede p whatever her acceptance attitude is. If, by contrast $\neg p$ is preferred over p, then a credulous agent must still concede p but a cautious and skeptical agent must instead proceed by claiming $\neg p$:

- B_1: *claim* $\neg p$.

Now W must apply clause (2) of the protocol, with $\varphi = \neg p$. Note that W must now reason with $\Sigma_W \cup \{\neg p\}$. He finds that he cannot accept $\neg p$ since his counterargument $(\{q, q \supset p\}, p)$ is acceptable since it is preferred over its only attacker $(\{\neg p, q \supset p\}, \neg q)$. Therefore, clause (2) requires him to assert p. However, the non-repetition rule makes this impossible, so that the dialogue terminates without agreement.

This example also illustrates that even if a proposition is defeasibly implied by $\Sigma_W \cup \Sigma_B$, it may not be agreed upon by the players (note that p is justified on the basis of this information). In fact, it also illustrates that sometimes there are no legal dialogues that agree upon such an implied proposition.

Paul and Olga (ct'd): Finally, our running example can be modelled in this approach as follows. Let us give Paul and Olga the following beliefs:

$\Sigma_W = \{$airbag, airbag $\supset$ safe, $\neg$(newspaper $\supset \neg$ safe)$\}$
$\Sigma_B = \{$newspaper, high-speed, newspaper $\supset \neg$ safe, high-speed $\supset \neg$ safe$\}$

(Note that Paul's undercutter must now be formalised as the negation of Olga's material implication.) Assume that all these propositions are equally preferred. We must also make some assumptions on the players' assertion and acceptance attitudes. Let us first assume that Paul is thoughtful and skeptical while Olga is careful and cautious, and that they only reason with their own beliefs and commitments.

P_1: *claim* safe O_2: *claim* $\neg$ safe

Olga could not challenge Paul's main claim as in the example's orginal version, since she can construct an argument for the opposite claim '$\neg$ safe', while she cannot construct an argument for 'safe'. So she had to make a counterclaim. Now since players may not repeat moves, Paul cannot make the move required by the protocol and his assertion attitude, namely, claiming 'safe', so the dialogue terminates without agreement.

Let us now assume that the players must also reason with each others commitments. Then the dialogue evolves as follows:

P_1: *claim* safe O_2: *concede* safe

Olga has to concede, since she can use Paul's commitment to construct the trivial argument ($\{$safe$\}$, safe), while her own argument for '$\neg$ safe' is not stronger. So here the dialogue terminates with agreement on 'safe', even though this proposition is not acceptable on the basis of the players' joint beliefs.

So far, neither of the players could develop their arguments. To change this, assume now that Olga is also thoughtful and skeptical, and that the players reason with each others commitments. Then:

P_1: *claim* safe O_2: *why* safe

Olga could not concede, nor could she state her argument for $\neg$ safe since it is not preferred over its attacker ($\{$safe$\}$,safe). So she had to challenge.

P_3: *claim* $\{$airbag, airbag $\supset$ safe$\}$

Now Olga can create a (trivial) argument for 'airbag' by using Paul's commitments, but she can also create an argument for its negation by using her own beliefs. Neither of these arguments is acceptable, so she must challenge again. Likewise for the second premise, so:

O_4: *why* airbag
P_5: *claim* $\{$airbag$\}$ O_6: *why* airbag $\supset$ safe
P_7: *claim* $\{$airbag $\supset$ safe$\}$

Here the nonrepetition rule makes the dialogue terminate without agreement. Note that only Paul could develop his arguments. To give Olga a chance to develop her arguments, let us make her careful and skeptical while the players still reason with each others commitments. Then:

P_1: *claim* safe O_2: *claim* $\neg$ safe

In the new dialogue state Paul's argument for 'safe' is not acceptable any more, since it is not preferred over its attacker ($\{\neg$ safe$\}$, $\neg$ safe). So he must challenge.

P_3: *why* $\neg$ safe O_4: *claim* $\{$newspaper, newspaper $\supset \neg$ safe $\}$

Although Paul can construct an argument for Olga's first premise, namely, ($\{\neg$(newspaper $\supset \neg$ safe'$\}$, safe), it is not acceptable since it is not preferred over its attacker based on Olga's second premise. So he must challenge.

P_5: *why* newspaper O_6: *claim* $\{$newspaper$\}$

Olga had to reply with a (trivial) argument for her first premise, after which Paul cannot repeat his challenge, so here the nonrepetition rule again makes the dialogue terminate without agreement. In this dialogue only Olga could develop her arguments (although she could not state her second counterargument).

In conclusion, the PWA persuasion protocol leaves little room for choice and exploring alternatives. Also, it induces one-sided dialogues in that at most one side can develop their arguments for a certain issue. The above examples also suggest that if a claim is accepted, it is accepted in the first 'round' of moves (but this should be formally verified). On the other hand, the strictness of the protocol induces short dialogues which are guaranteed to terminate, which is good for efficiency reasons. Also, without the requirement to respect the assertion and acceptance attitudes the protocol would be much more liberal while still enforcing some coherence.

8.4.2 Prakken (2005)

In Prakken (2005) a framework for specifying two-party persuasion dialogues about a single dialogue topic is presented, which is then instantiated with some example protocols. The participants have proponent and opponent role, and their beliefs are irrelevant to the protocols. Dialogues have no context. The framework largely abstracts from the communication language, except for an explicit reply structure. It also largely abstracts from the logical language and the logic, except that the logic is assumed to conform to the format of the framework of this reader's Chapter 6 with Dung (1995)'s grounded semantics. The logic is used to verify whether a moved argument is logically constructible, to allow for explicit counterarguments, and to verify whether these arguments defeat their targets.

A main motivation of the framework is to ensure focus of dialogues while yet allowing for freedom to move alternative replies and to postpone replies. This is achieved with two main features of the framework. Firstly, an explicit reply structure on $\mathcal{L}_c$ is assumed, where each move either *attacks* or *surrenders to* its target. An example $\mathcal{L}_c$ of this format is displayed in Table 8.2. This enables the second feature of the frame-

Table 8.2: An example L_c in Prakken's framework

Acts	Attacks	Surrenders
claim φ	*why* φ	*concede* φ
argue A	*why* φ ($\varphi \in \texttt{Prem}(A)$) *argue* B (B defeats A)	*concede* φ ($\varphi \in \texttt{Prem}(A)$) *concede* φ ($\varphi = \texttt{Conc}(A)$)
why φ	*argue* A ($\varphi = \texttt{Conc}(A)$)	*retract* φ
concede φ		
retract φ		

work, namely, an 'any-time' notion of winning that is defined in terms of a notion of *dialogical status* of moves.

Accordingly, particular communication languages must satisfy the following format.

Definition 8.4.6 (Dialogues) The set $\mathcal{L}_c$ of *moves* is defined as $\mathbb{N} \times \{P, O\} \times L_c \times \mathbb{N}$, where the four elements of a move m are denoted by, respectively:

- $id(m)$, the *identifier* of the move,
- $pl(m)$, the *player* of the move,
- $s(m)$, the *speech act* performed in the move,
- $t(m)$, the *target* of the move.

When $t(m) = id(m')$ we say that m replies to m' in d and that m' is the target of m in d. Abusing notation we sometimes let $t(m)$ denote a move instead of just its identifier. When $s(m)$ is an attacking (surrendering) reply to $s(m')$ we also say that m is an attacking (surrendering) reply to m'.

All protocols are further assumed to satisfy the following basic conditions for all moves m_i and all legal finite dialogues d. Note that these protocol rules only state necessary conditions for legality of moves; they can be completed in many ways with further conditions.

If $m \in Pr(d)$, then:

R_1: $t(m) = 0$ iff $m = m_1$.

R_2: If $t(m) \neq 0$ then $t(m) = i$ for some m_i preceding m in d.

R_3: $pl(m) \in T(d)$.[3]

R_4: If $t(m) \neq 0$ then $s(m)$ is a reply to $s(t(m))$ according to L_c.

R_5: If m replies to m', then $pl(m) \neq pl(m')$.

R_6: If there is an m' in d such that $t(m) = t(m')$ then $s(m) \neq s(m')$.

R_7: For any $m' \in d$ that surrenders to $t(m)$, m is not an attacking counterpart of m'.

[3] Recall that $T(d)$ denotes the player(s) whose turn it is to move in d.

R_8: If $d = d_0$ then $s(m)$ is of the form *claim* φ or *argue* A.

R_1 gives the first move a 'dummy' target; together with R_2 it says that all moves except the first reply to some earlier move in the dialogue. Rule R_3 says that the player of a move must be to move according to the turntaking function. R_4 says that a replying move must pick the reply to its target from Table 8.2. R_5 says that a player can only reply to the other player's moves. R_6 makes sure that a new reply to the same target has a different content. Rule R_7 says that once a move is surrendered, it may not be attacked any more (an attacking counterpart of a surrendering move is any attacking move that replies to the same target as the surrendering move). Finally, R_8 says that each dialogue begins with a claim or argument. The claim or conclusion of the argument is the dialogue's topic.

To define the dialogical status of a move first the notion of a surrendered move must be defined. A complication here is that surrendering to a premise of an argument does not yet mean that the argument is surrendered, since if the argument is defeasible; it can still be attacked with a counterargument even if all of its premises are conceded. Therefore, the notion of a surrendered move is defined as follows.

Definition 8.4.7 A move m in a dialogue d is *surrendered* in d iff

- if m is an *argue* A move then it has a *concede* φ reply in d, where $\varphi = \texttt{Conc}(A)$;
- else m has a surrendering reply in d.

The *dialogical status* of a move is now recursively defined as follows, exploiting the reply structure of dialogues.

Definition 8.4.8 [Dialogical status of moves] All attacking moves in a finite dialogue d are either *in* or *out* in d. Such a move m is *in* iff

1. m is surrendered in d; or else
2. all attacking replies to m are *out*

Otherwise m is *out*.

We can now define an 'anytime' outcome function for dialogues (whether or not they are terminated).

Definition 8.4.9 [The current winner of a dialogue]

- The status of the initial move m_1 of a dialogue d is *in favour of* $P(O)$ and *against* $O(P)$ iff m_1 is *in* (*out*) in d. We also say that m_1 favours, or is against p.
- $w_t(d) = p$ (i.e., player p *currently wins* dialogue d on topic t) iff m_1 of d favours p. Furthermore, $l_t(d) = p$ iff $w_t(d) = \overline{p}$.

The framework defined thus far allows for a structural notion of relevance that ensures focus while yet leaving the desired degree of freedom: a move is *relevant* just in case making its target *out* would make the speaker the current winner.

Definition 8.4.10 [Relevance] An attacking move in a dialogue d is *relevant* iff it changes the dialogical status of d's initial move. A surrendering move is relevant iff its attacking counterparts are relevant.

Note that, if not surrendered, an irrelevant target can become relevant again later in a dialogue, viz. if a player returns to a dialogue branch from which s/he has earlier retreated.

To illustrate these definitions, consider Figure 8.2 (where + means *in* and - means *out*). The dialogue tree on the left is the situation after P_7. The tree in the middle shows the dialogical status of the moves when O has continued after P_7 with O_8, replying to P_5: this move does not affect the status of P_1, so O_8 is irrelevant. Finally, the tree on the right shows the situation where O has instead continued after P_7 with O_8', replying to P_7: then the status of P_1 has changed, so O_8' is relevant.

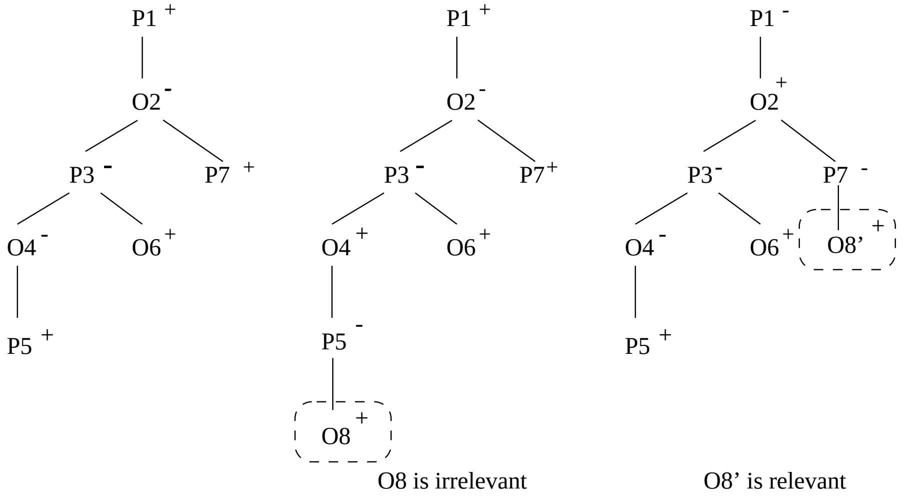


Figure 8.2: Dialogical status and relevance.

As for dialogue structure, the framework allows for all kinds of protocols. The instantiations presented in Prakken (2005) are all multi-move and multi-reply. One of them has the communication language of Table 8.2 and has one additional protocol rule, viz. that each move be relevant, while the turn shifts as soon as the player-to-move has succeeded in becoming the current winner. Protocols with this protocol and turntaking rule are called *protocols for relevant dialogue*. Together, these rules imply that each turn consists of zero or more surrenders followed by one attacker. Within these limits postponement of replies is allowed, sometimes even indefinitely.

We next discuss some examples in terms of a logic within the framework of Chapter 6 combined with grounded semantics. The connective $\rightsquigarrow$ is governed by defeasible modus ponens as in Section 6.4.1 above. We assume that the logic supports arguments about preferences, so that the definition of an overall preference ordering on the basis of the players' individual preferences is in fact the result of the dialogue. The example below should speak for itself so no formal definitions about the logic will be given. Consider two agents with the following belief bases (rule connectives are tagged with a rule name, which is needed to express rule priorities in the object language)

$$\Sigma_P = \{q, q \rightsquigarrow_{r_1} p,\ q \wedge s \rightsquigarrow_{r_3} r_1 > r_2\}$$
$$\Sigma_O = \{r, r \rightsquigarrow_{r_2} \neg p\}.$$

Then the following is a legal dialogue:[4]

[4] From now on we will, when the internal structure of the reasoning within an argument does not matter,

- P_1: *claim* p, O_1: *why* p, P_2: p *since* $q, q \rightsquigarrow p$, O_2: *concede* $q \rightsquigarrow p$, O_3: *why* q.

At this point P has four allowed moves, viz. retracting p, retracting q, giving an argument for q or giving a second argument for p. Note that the set of allowed moves is not constrained by P's belief base. If the dialogue terminates here since P withdraws from it then O has won since P_1 is *out*.

The dialogue may also evolve as follows. The first three moves are as above and then:

- O_2: $\neg p$ *since* $r, r \rightsquigarrow \neg p$
  P_3: $r_1 > r_2$ *since* $q, s, q \wedge s \rightsquigarrow r_1 > r_2$

P_3 is a priority argument which in the underlying logic makes P_2 strictly defeat O_2 (note that the fact that s is not in P's own knowledge base does not make the move illegal). At this point, P_1 is *in*; the opponent has various allowed moves, viz. challenging or conceding any premise of P_2 or P_3, moving a counterargument to P_3 or a second counterargument to P_2, conceding one of these two arguments, and conceding P's initial claim.

This example shows that the participants have much more freedom in this system than in the one of Parsons *et al.* (2003). The downside of this is that dialogues can be much longer, and that the participants can prevent losing by simply continuing to challenge premises of arguments of the other participant. One way to tackle such 'filibustering' is to introduce a context; another way is to introduce a third party who may reverse the burden of proof after a challenge: the challenger of φ then has to provide an argument for $\overline{\varphi}$.

Another drawback of Prakken's approach is that not all dialogues that can be found in natural language conform to an explicit reply structure. For instance, in legal cross-examination dialogues the purpose of the cross-examiner is to reveal an inconsistency in the testimony of a witness. Typically, questions by cross-examiners do not indicate from the start what they are aiming at, as in

> *Witness*: Suspect was at home with me that day.
> *Prosecutor*: Are you a student?
> *Witness*: Yes.
> *Prosecutor*: Was that day during summer holiday?
> *Witness*: Yes.
> *Prosecutor*: Aren't all students away during summer holiday?

Paul and Olga (ct'd): Let us finally model our running example in this protocol. Figure 8.3 displays the dialogue tree, where moves within solid boxes are *in* and moves within dotted boxes are *out*. As can be easily checked, this formalisation captures all aspects of our original version, except that arguments have to be complete and that counterarguments cannot be introduced by a counterclaim. (But other instantiations of the framework may be possible without these limitations.)

8.5 Conclusion

In this chapter we have discussed two systems for persuasion dialogue in terms of a formal specification of the main elements of such systems. In the literature a number of

write *argue* A moves as φ *since* S, where φ is A's conclusion and S are A's premises.

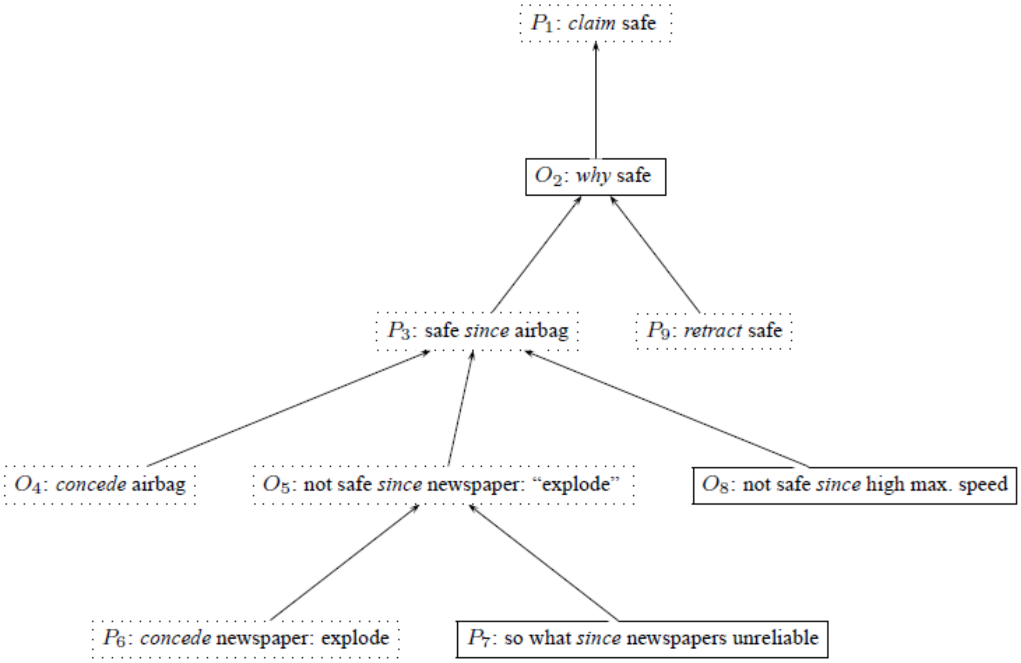

P_1: *claim* safe
O_2: *why* safe
P_3: safe *since* airbag
P_9: *retract* safe
O_4: *concede* airbag
O_5: not safe *since* newspaper: "explode"
O_8: not safe *since* high max. speed
P_6: *concede* newspaper: explode
P_7: so what *since* newspapers unreliable

Figure 8.3: The example dialogue in Prakken's approach.

interesting dialogue-game protocols for persuasion have been proposed, some of which have been applied in insightful case studies or applications. However, a consensus on many issues is stil lacking. As a consequence, there is still little work on formally relating the various systems or on a general framework for designing persuasion protocols, and a formal metatheory of systems is still in its early stages. These are some of the main issues that should be tackled in future research. Some other issues are the study of strategies and heuristics for individual participants and how these interact with the protocols to yield certain properties of dialogues, a similar study of varying degrees of cooperativeness of participants, and the integration of persuasion systems with systems for other types of dialogues. Perhaps the main challenge in tackling all these issues is how to reconcile the need for flexibility and expressiveness with the aim to enforce coherent dialogues. The answer to this challenge may well vary with the nature of the context and application domain, and a precise description of the grounds for such variations would provide important insights in how dialogue systems for persuasion can be applied.

8.6 Exercises

EXERCISE 8.6.1 Show in detail how the argument games of Chapter 5 instantiate Definition 8.2.1.

EXERCISE 8.6.2 Define disputes as a subtype of persuasion dialogues in terms of Definition 8.3.1.

8.6.1 On Parsons, Wooldridge & Amgoud (2003)

EXERCISE 8.6.3 Let $\Sigma_W = \{q, q \supset p\}$ and $\Sigma_B = \{\neg p, q \supset p\}$. Let the preference ordering $\preceq$ on formulas be:

$\Sigma_1 = \{q\}$
$\Sigma_2 = \{p, \neg p, \neg(q \supset p)\}$
$\Sigma_3 = \{q \supset p\}$

Finally, assume that both players are thoughtful and skeptical and that these attitudes are verified relative to the speaker's beliefs and the hearer's commitments.

1. What is the dialectical status of p, $\neg p$ and $q \supset p$ on the basis of $\Sigma_W \cup \Sigma_B$ and $\preceq$?
2. Produce all legal dialogues on topic p. Determine the commitment sets of the players at termination. Are these sets consistent? And what is for each player the dialectical status of p and $\neg p$ on the basis of their internal beliefs plus their own commitments?
3. Assume now that the assertion and acceptance attitudes are verified relative to the speaker's beliefs and his own commitments, and answer again the previous question.

EXERCISE 8.6.4 Let $\Sigma_W = \{q, q \supset p\}$ and $\Sigma_B = \{q \supset p\}$ and let all formulas be of the same preference level. Assume that W is thoughtful and cautious while B is careful and skeptical and that both players reason with their own beliefs only.

1. Produce all legal dialogues on topic p.
2. Think of an acceptance attitude that allows a player to learn from the other agent but that avoids the problems as illustrated by Example 8.4.4.

EXERCISE 8.6.5 Let $\Sigma_W = \{p, p \supset q, q \supset r\}$ and $\Sigma_B = \{s, s \supset \neg q\}$. Let all formulas be of the same preference level. Assume that W is thoughtful and cautious while B is careful and skeptical and that both players reason with their own beliefs and commitments. Assume finally that the players also apply the attitude that you defined in your answer to Exercise 8.6.4(2). Produce all legal dialogues on topic r if clause (4b) of the PWA protocol is applied in a depth-first fashion, i.e., if after each response to an element from S the other player may first respond to that response before the first player responds to the next element from S.

EXERCISE 8.6.6 Assume both players are thoughtful and skeptical.

1. Assume that these attitudes are verified relative to the speaker's beliefs and the hearer's commitments. Prove or refute:

   > If W and B agree on preference ordering $\preceq$ and at termination of dialogue d on topic t both $C_W(d) \vdash t$ and $C_B(d) \vdash t$, then t is justified on the basis of $\Sigma_W \cup \Sigma_B$ and $\preceq$.

2. Assume now that the assertion and acceptance attitudes are verified relative to the speaker's beliefs and commitments, and that the players also apply the attitude that you defined in your answer to Exercise 8.6.4(2). Answer the same question.

8.6.2 On Prakken (2005)

EXERCISE 8.6.7 Prove that for each finite dialogue d there is a unique dialogical status assignment. Give a counterexample for infinite dialogues. (Hint: use results stated in Chapter 4.)

EXERCISE 8.6.8 Explain that a reply to a surrendered move is never relevant.

EXERCISE 8.6.9 Answer the following questions about Figure 8.3.

1. What are the relevant targets for O after P_7?

2. What are the relevant targets for P after O_8?

3. Assume at P_9 that P does not retract *safe* but instead moves another argument for *safe* in reply to O_2. What are then the relevant targets for O after P_9?

EXERCISE 8.6.10 Assume an instance of the dialogue framework of Prakken (2005) with the same argumentation logic as the dialogue system of Parsons, Wooldridge & Amgoud, with the communication language of Table 8.2, and with a protocol for relevant dialogue. Give a terminated dialogue starting with *claim* q and won by O in which at least three different arguments constructible from the knowledge base $\Sigma = \{p, p \supset q, r, r \supset \neg p\}$ are moved, where all formulas are of equal preference.

Chapter 9

Answers to exercises from Chapters 3-7

9.1 Answer to exercise Chapter 3

EXERCISE 3.2.1

1. B and D are justified. B is reinstated by D.
2. A, C and E are justified. No argument is reinstated by D, since D is not justified. A and C are reinstated by E.

9.2 Answers to exercises Chapter 4

EXERCISE 4.8.1

(a): C is justified since it has no defeaters. B is not justified, since it is defeated by a justified argument, viz. by C. Therefefore, A is justified, since its only defeater, which is B, is not justified.
(b): The status of A and B cannot be determined: A is justified if and only if its only defeater, which is B, is not justifed. But B is not justified just in case A, which is its only defeater, is justified. Thus we enter a loop. And since the status of C depends on the status of its only defeater, which is B, the status of C cannot be determined either.

EXERCISE 4.8.2 Consider an arbitrary argument A. By assumption, there is an argument B such that B defeats A. So $A \in F(\varnothing)$ iff there is a $C \in \varnothing$ such that C defeats B. However, no such C exists, so $A \notin F(\varnothing)$. Since A was chosen arbitrarily, we can conclude that no argument is in $F(\varnothing)$. □.

EXERCISE 4.8.3

a:	b:	c:	d:
$F^0 = \varnothing$	$F^0 = \varnothing$	$F^0 = \varnothing$	$F^0 = \varnothing$
$F^1 = \{A\}$	$F^1 = F^0$	$F^1 = \{C\}$	$F^1 = \{A, E\}$
$F^2 = \{A, D\}$		$F^2 = \{C, B\}$	$F^2 = \{A, E, C\}$
$F^3 = F^2$		$F^3 = F^2$	$F^3 = F^2$

The grounded extensions are the fixed points of these sequences.

So the grounded extension is $\{A, D\}$.

EXERCISE 4.8.4

1. To show that $F(X) = G^2(X)$, for every set of arguments X, it turns out that it is easier to show that the complements of the two sets are equal. This has to do with quantifying over arguments. Thus, suppose $x \notin G^2(X)$. By definition of G this means that there exists a $y \in G(X)$ defeating x, i.e., $x \leftarrow y$. Since $y \in G(X)$, the argument y is not defeated by a member of X. Hence y shows that $x \notin F(X)$. Conversely, suppose that $x \notin F(X)$. Then x is defeated by a y that is not defeated by a $z \in X$. Thus x is defeated by a $y \in G(X)$, and hence $x \notin G^2(X)$.

2. The result that G is anti-monotonic follows from the fact that, if an argument is not defeated by a member of B, then it surely cannot be defeated by a member of any subset $A \subseteq B$.

3. Suppose $A \subseteq B$. Since G is anti-monotonic, it follows that $G(B) \subseteq G(A)$. Again by anti-monotonicity of G, we obtain $G^2(A) \subseteq G^2(B)$, which is equal to the expression $F(A) \subseteq F(B)$.

4. If $\{G_i\}_{i \geq 0}$ with $G_0 =_{Def} \varnothing$ and $G_i =_{Def} G(G_{i-1})$, then in particular

$$G_0 \subseteq G_1 \text{ and } G_0 \subseteq G_2. \tag{9.1}$$

   Now apply the anti-monotonicity of G to (9.1) repeatedly, to obtain the chain of inclusions desired.

EXERCISE 4.8.5

- (a): justified: A, D; overruled: B, C; defensible: none.
- (b): justified: none; overruled: none; defensible: all.
- (c): justified: B, C; overruled: A, D; defensible: none.
- (d): justified: A, C, E; overruled: B, D; defensible: none.

EXERCISE 4.8.6

⇒:

Consider any stable extension E, and consider first any argument A not defeated by E. Then $A \in E$. Consider next any argument B defeated by E. Then, since E is conflict-free, $B \notin E$. So $E = \{A \mid A \text{ is not defeated by } E\}$.□

⇐:

Let $E = \{A \mid A \text{ is not defeated by } E\}$. Clearly, E is conflict-free. Furthermore, for all A, if $A \notin E$, then E defeats A. So E is a stable extension.□

EXERCISE 4.8.7

- Example 4.1.3: There is just one status assignment, which is maximal:
  - $S_1 = (\{A, C\}, \{B\})$
- Example 4.1.4: There are three status assignments:

  - $S_1 = (\varnothing, \varnothing)$
  - $S_2 = (\{A\}, \{B\})$
  - $S_3 = (\{B\}, \{A\})$

  Only S_2 and S_3 are maximal.

- Example 4.3.8: There is just one status assignment, which is maximal:

  - $S_1 = (\varnothing, \varnothing)$

EXERCISE 4.8.8

1. Consider any $A \in \mathit{Out}$. Then there is a $B \in \mathit{In}$ defeating A. But also $B \in \mathit{In}'$, so that $A \in \mathit{Out}'$. So $\mathit{Out} \subseteq \mathit{Out}'$.

2. Consider any argument C such that $C \notin \mathit{In}$ but $C \in \mathit{In}'$.
(i) Since $C \notin \mathit{In}$, there exists a $B \notin \mathit{Out}$ such that B defeats C.
(ii) Consider next any such B that defeats C and is not in Out. Any such B must be in Out', otherwise C would not be in In'.
Hence (from i and ii) there exists an argument that is in Out' but not in Out. Together with (1) this gives us that Out is a proper subset of Out'.

EXERCISE 4.8.9

- A is defensible iff is in *in* some but not all preferred status assignments, and A is overruled if A is *out* in all preferred status assignments. *This leaves open that there are arguments that neither justified, nor defensible, nor overruled. Cf. Example 4.3.8.*
- A is defensible iff is in *in* some but not all preferred status assignments, and A is overruled if there is no status assignment in which A is *in*. *With this definition all arguments are either justified, Xor defensible, Xor overruled.*

EXERCISE 4.8.10: The empty set, which is maximally admissible.

EXERCISE 4.8.11

1. (a) Preferred: $\{A, D\}$, also stable.

   (b) Preferred: $\{B, D, E\}$, also stable; $\{A, E\}$, also stable.

   (c) Preferred: $\varnothing$, no stable extensions.

   (d) Preferred: $\{A, C, E\}$, also stable.

   (e) (with slightly detailed explanation)
   (1) Preferred extensions:

   - $E_1 = \{A, B, D\}$
   - $E_2 = \{C\}$

   (2) Stable extensions. Both E_1 and E_2 are also stable extensions, since both sets defeat all arguments outside them. Furthermore, by Proposition 4.4.1 there are no other stable extensions.

2. (a) for preferred and stable semantics: A, D justified, B, C overruled.

(b) for preferred and stable semantics: E justified, C overruled, A, B, D defensible.

(c) for preferred semantics: neither is justified, defensible or overruled. For stable semantics: all are both justified and overruled.

(d) For preferred and stable semantics: A, C, E justified, B, D overruled.

(e) For preferred and stable semantics: all defensible

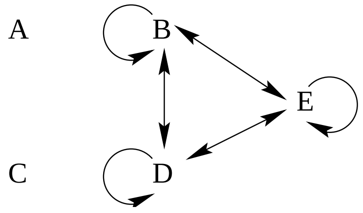
EXERCISE 4.8.12: The grounded extension is empty, while there are two preferred extensions, viz. $\{B, D\}$ and $\{A, C\}$. Note that one preferred extension concludes that Larry is rich, while the other concludes that Larry is not rich, so in bothsemantics no conclusion about Larry's richness is justified. Yet it may be argued that the conclusion that Larry is not rich is the intuitively justified conclusion, since all arguments for the opposite conclusion have a strict defeater. Anyone who adopts this analysis, will have to conclude that this example presents a problem for both grounded and preferred semantics. However, see Exercise 6.8.9 for a solution when the structure of arguments is made explicit.

EXERCISE 4.8.13

1. $AF(\Delta_3)$ contains five arguments:

   - $A = \varnothing$
   - $B = \frac{:p}{\neg p}$
   - $C = \frac{:q}{q}$
   - $D = \frac{:p}{\neg p}, \frac{:q}{q}$
   - $E = \frac{:q}{q}, \frac{:p}{\neg p}$

   The defeat graph is as follows:

   There is no stable extension, while there is one preferred extension, viz. $\{A, C\}$.

2. Since it recognizes that A and C should come out as justified, since they have no defeaters.

EXERCISE 4.8.14

1.

   (a) $A = \frac{:b}{a}, \frac{a:c \wedge d}{c}, \frac{c:b}{b}$

   (b)

   - $B = \frac{:e}{e}, \frac{e:\neg a}{\neg d}$
   - $C = \frac{:\neg a}{\neg a}, \frac{:b}{a}$

(c) Yes, for instance of $\{A\}$. Note that A defeats both B and C. Another admissible set is $\{A, A'\}$, where

- $A' = \frac{:b}{a}$

Note that A' also defeats both B and C.

(d) Yes, by (c) and the fact that every admissible set is contained in a preferred extension (see the proof of Proposition 4.3.13).

(e) No: the grounded extension is empty, since there is no undefeated argument. In particular, A' is defeated by C.

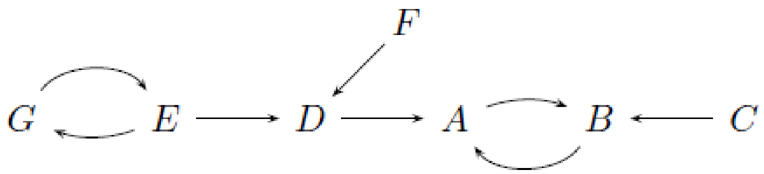
EXERCISE 4.8.15

Suppose A is finite and failed. Then $In(A) \cup Out(A) \neq \varnothing$, so $\varphi \in In(A)$ for some $\varphi \in Out(A)$. But then A defeats A.

9.3 Exercises Chapter 5

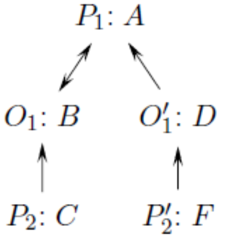
EXERCISE 5.5.1

1. The defeat graph is:

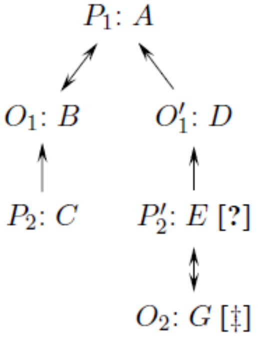


2. We are asked to list all strategies of P an O. There are two strategies for P ("?" indicates an unfortunate move, "‡" indicates the move that leads to a loss for the other party):

Strategy 1 for P (responding to D with F and winning)



Strategy 2 for P (responding to D with E and losing)



There are two strategies for O:

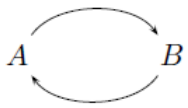
Strategy 1 for O (responding to A with B and losing)

$$P_1\text{: } A \longleftrightarrow O_1\text{: } B \longleftarrow P_2\text{: } C\ [\ddagger]$$

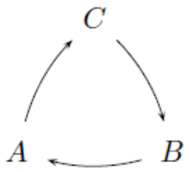
Strategy 2 for O (responding to A with D and losing)

$$\begin{tikzcd}
 & & P_2\text{: } E\ [?] \arrow[dl] \arrow[r, leftrightarrow] & O_2\text{: } G \\
P_1\text{: } A & O_1\text{: } D \arrow[l] & & \\
 & & P_2\text{: } F\ [\ddagger] \arrow[ul] &
\end{tikzcd}$$

EXERCISE 5.5.2



1.



2.

$$A_1 \longleftarrow A_2 \longleftarrow A_3 \longleftarrow A_4 \longleftarrow A_5 \longleftarrow \cdots$$

3.

EXERCISE 5.5.3

(1a) P has winning strategies for A and D, but not for B:
- A winning strategy for A consists of putting forward A, after which O cannot respond because A has no defeaters.
- A winning strategy for B does not exist, because O can reply to B with A, after which P cannot move.
- A winning strategy for D is simple: put forward D; the only responses to D are B and C, which can both be countered with A, after which O cannot move.

(3) We make the comparison for the proof of A in graph (a):

$F^0 = \varnothing$
$F^1 = \{A\}$
$F^2 = \{A, D\}$

Compared to a won dialogue on D, the order of stating A and D is reversed. With F, we start with the undefeated arguments and at each iteration add the arguments reinstated by the arguments added at the previous iteration. In a dialectical proof, P starts with an argument from F^i where i may be greater than 1, and at each next turn P moves an argument from F^{i-1} that can reinstate the argument of the previous move.

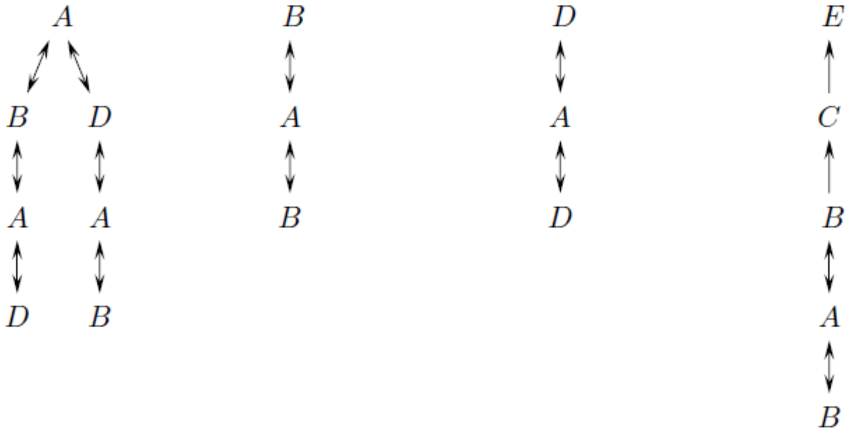


Figure 9.1: P's winning strategies (first attempt)

EXERCISE 5.5.4 The argument $A == \frac{:b}{a}, \frac{a:c \wedge d}{c}, \frac{c:b}{b}$ is not provably justified, since O can reply with $C = \frac{:\neg a}{\neg a}, \frac{:b}{a}$, after wich P has no strictly defeating reply.

EXERCISE 5.5.5 P successively moves $A_1, A_3, \ldots, A_{2i-1}, A_{2i+1}, \ldots$ and O successively moves $A_2, A_4, \ldots, A_{2i}, A_{2i+2}, \ldots$ so they wil never repeat their own argument. And P always uses 'odd' arguments while O always uses 'even' arguments, so they will never repeat each other's argument. Finally, since the defeat chain is infinite, they will always have a new move.

EXERCISE 5.5.6 The simplest example is with two arguments A and B such that A defeats itself and there are no other defeat relations. B is provable since O has no reply if P starts with B, but this argumentation framework has no stable extensions.

EXERCISE 5.5.7

(1a): All arguments except argument C are provable. Figure 9.1 contains a first attempt to display the winning strategies for P. However, these trees are not yet strategies, since they do not contain all possible backtracking replies of O as children of a P move. (Note that a strategy is not a tree of *dispute lines* but a tree of *disputes*, so that a next move in a branch of a strategy may well reply not to the previous move but to an earlier move in the branch.) So the correct winning strategy for A is a lot more complex.

Let us illustrate this with a simpler example, viz. the graph of Exercise 4.8.11(a). At first sight, a winning strategy for D would look as in Figure 9.2. However, the correct winning strategy is as displayed in Figure 9.3 (where the replied-to move is indicated between brackets).

(1b): We show that D is not provable. In general, to show that an argument is not provable, it suffices to show one strategy for O in which P cannot win. Here, such a strategy is the one in which O replies to D with A; then P's only legal reply is B, to which O replies with C and P has run out of moves, so O wins.

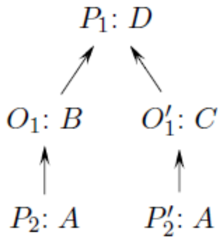


Figure 9.2: P's seeming winning strategy for D in 4.8.11(a)

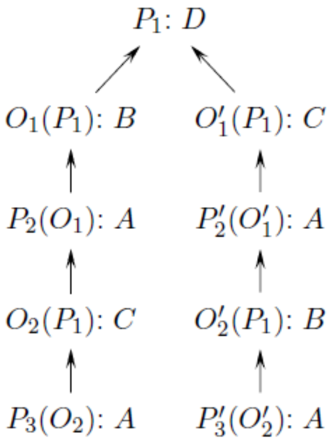


Figure 9.3: P's correct winning strategy for D in 4.8.11(a)

9.4 Exercises Chapter 6

EXERCISE 6.8.1.

1. The following argument for r can be constructed.

   A_1: p
   A_2: $A_1 \Rightarrow q$
   A_3: $A_1, A_2 \rightarrow r$

   We verify the status of r with the G-game. Argument A_3 has one defeater, namely the following undercutter of A_2.

   B_1: s
   B_2: $B_1 \Rightarrow t$
   B_3: $B_2 \rightarrow \neg d_1$

   Argument B_3 has one attacker, rebutting B_3 on B_2:

   C_1: u
   C_2: $C_1 \Rightarrow v$
   C_3: $C_2 \Rightarrow \neg t$

   We are in case (4) of Definition 6.3.24. First, since $u <' s$, we have that $\mathtt{Prem}_p(C_3) \triangleleft_{\mathtt{Eli}} \mathtt{Prem}_p(B_2)$. Next, B_2 uses one defeasible rule, namely, d_2, while C_3 uses two defeasible rules, namely, d_3 and d_4. Since $d_3 < d_2$ we have that

$\texttt{DefRules}(C_3) \lhd_{\texttt{Eli}} \texttt{DefRules}(B_2)$. So $C_3 \prec B_2$, so B_2 strictly defeats C_3 and C_3 does not defeat B_3. Hence the proponent has no winning strategy for A_3 in the G-game, so r is not justified. Moreover, B_3 is justified since it has no defeaters, so A_3 is overruled, which also makes r overruled, since there are no other arguments for r.

2. Now r is justified. First, since both arguments are defeasible, the premise ordering is now irrelevant. Next, since arguments B_2 and C_3 are now compared on d_2 and d_4 and since $d_2 < d_4$, we have $\texttt{LastDefRules}(B_2) \lhd_{\texttt{Eli}} \texttt{LastDefRules}(C_3)$. So $B_2 \prec C_3$ and C_3 strictly defeats both B_2 and B_3. Moreover, there are no defeaters of C_3, so the proponent now has a winning strategy for A_3 in the G-game.

EXERCISE 6.8.2

1. The following argument for Ra can be created.

A_1: $\forall x(Px \supset Qx)$
A_2: Pa
A_3: $A_1, A_2 \rightarrow Qa$
A_4: $\forall x(Qx \supset Rx)$
A_5: $A_3, A_4 \rightarrow Ra$

2. $\texttt{Prem}(A) = \{Pa, \forall x(Px \supset Qx), \forall x(Qx \supset Rx)\}$
$\texttt{Conc}(A) = Ra$
$\texttt{Sub}(A) = \{A_1, A_2, A_3, A_4, A_5\}$
$\texttt{DefRules}(A) = \varnothing$
$\texttt{TopRule}(A) = Qa, \forall x(Qx \supset Rx) \rightarrow Ra$

3. The argument is strict and plausible.

EXERCISE 6.8.3.

1. We again use the G-game. The following argument for t can be created.

A_1: p
A_2: q
A_3: $A_1, A_2 \Rightarrow r$
A_4: $A_3 \rightarrow r \vee s$
A_5: $A_4 \Rightarrow t$

A_5 has one attacker, namely, the following underminer on A_2:

B_1: u
B_2: $B_1 \Rightarrow v$
B_3: $\neg(q \wedge v)$
B_4: $B_2, B_3 \rightarrow \neg q$

Since the argument ordering is simple, we have that $A_2 \approx B_4$, so B_4 successfully undermines A_5 on A_2 and thus defeats A_5. Argument B_4 in turn has two attackers. Firstly, B_4 is undermined by the following argument for $\neg u$:

C_1: w
C_2: $C_1 \Rightarrow \neg u$

Again since the argument ordering is simple, C_2 successfully undermines B_4 on B_1, so C_2 strictly defeats B_4. (Note that this is strict defeat since B_4 does not even attack C_2.) However, we also have that B_1 rebuts C_2 and since the argument ordering is simple, we have that B_1 defeats C_2, so the opponent can reply to C_2 with B_1. Then the game ends with a win by the opponent, since we have $C_2 \approx B_1$ so there is no strict defeater of B_1.

The proponent can also strictly defeat B_4 with the following rebuttal of B_2:

D_1: q
D_2: $\neg(q \wedge v)$
D_3: $D_1, D_2 \rightarrow \neg v$

However, then the opponent can repeat B_4, defeating D_3 on D_1, and again the game ends with a win by the opponent. Since the proponent has no other options, he has no winning strategy for A_5, so A_5 and t are not justified.

To see whether A_5 is defensible or overruled, note that the only defeater of A_5 is B_4 but the proponent does not have a winning strategy for B_4: it is defeated by C_2, which has no strict defeater. Hence B_4 is not justified, so A_5 is defensible, which makes t defensible also.

2. t is now justified, since argument B_4 does not defeat argument A_2, so A_5 has no defeaters. To see this, observe that $\texttt{LastDefRules}(A_2) = \varnothing$ while $\texttt{LastDefRules}(B_4) = \{u \Rightarrow v\} \neq \varnothing$, so $\texttt{LastDefRules}(B_4) \triangleleft_{\texttt{Eli}} \texttt{LastDefRules}(A_2)$, so $B_4 \prec A_2$.

3. Now t is not justified. Note that $\texttt{Prem}_p(A_2) = \{q\}$ while $\texttt{Prem}_p(B_4) = \{u\}$ and since $q <' u$ we have that $\texttt{Prem}_p(A_2) \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(B_4)$. Then despite the fact that $\texttt{DefRules}(B_4) \triangleleft_{\texttt{Eli}} \texttt{DefRules}(A_2)$ we have that $B_4 \not\prec A_2$, so B_4 defeats A_2 and thus B_4 also defeats A_5.

   Next we have to verify whether any attack on B_4 succeeds as defeat. Consider first C_2. We have that $\texttt{Prem}_p(C_2) = \{w\}$ while $\texttt{Prem}_p(B_1) = \{u\}$ and $w <' u$, so we have that $\texttt{Prem}_p(C_2) \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(B_1)$. Moreover, we have that $\texttt{DefRules}(B_1) = \varnothing$ and $\texttt{DefRules}(C_2) = \{w \Rightarrow \neg u\}$ so $\texttt{DefRules}(C_2) \triangleleft_{\texttt{Eli}} \texttt{DefRules}(B_1)$. So $C_2 \prec B_1$ so C_2 does not defeat B_1.

   Consider next D_3. We have that $\texttt{DefRules}(B_4) = \{u \Rightarrow v\}$ and $\texttt{DefRules}(D_3) = \varnothing$, so $\texttt{DefRules}(B_4) \triangleleft_{\texttt{Eli}} \texttt{DefRules}(D_3)$. However, we also have that $\texttt{Prem}_p(D_3) = \{q\}$ while $\texttt{Prem}_p(B_4) = \{u\}$ and $q <' u$, so we also have that $\texttt{Prem}_p(D_3) \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(B_4)$. Since both B_4 and D_3 are neither strict nor firm, we have to apply clause (3) of Definition 6.3.24. But then $D_3 \not\prec B_4$ and $B_4 \not\prec D_3$, so D_3 and B_4 defeat each other. Then the proponent cannot move D_3 in reply to B_4 in the G-game.

   But then the proponent has no legal reply to B_4 in the G-game, so the proponent does not have a winning strategy for A_5.

EXERCISE 6.8.4.

1. The following argument for t can be created.

   A_1: s
   A_2: $A_1 \Rightarrow t$

A_2 is rebutted by the following argument for $\neg t$:

B_1: p
B_2: $B_1 \Rightarrow q$
B_3: $B_1, B_2 \Rightarrow r$
B_4: $B_2, B_3 \rightarrow q \wedge r$
B_5: $(q \wedge r) \supset \neg t$
B_6: $B_4, B_5 \rightarrow \neg t$

(Note that since B_6 is strict, A_2 does not in turn rebut B_6.) We have that $\texttt{LastDefRules}(A_2) = \{d_3\}$ while $\texttt{LastDefRules}(B_6) = \{d_1, d_2\}$. Since $d_2 < d_3$ we have that $\texttt{LastDefRules}(B_6) \lhd_{\texttt{Eli}} \texttt{LastDefRules}(A_2)$, so $B_6 \prec A_2$. Hence B_6 does not defeat A_2. Since A_2 has no other defeaters, we can concude at this point that A_2 will be *in* in all preferred status assignments, which makes it justified. Then t is a justified conclusion.

It is interesting to verify the status of argument B_6 for $\neg t$. Since the present argumentation theory is well defined, it is to be expected that this conclusion is not justified. This turns out to be indeed the case. First of all, A_2 can be extended to a rebuttal of B_3:

A_3: $(q \wedge r) \supset \neg t$
A_4: $A_2, A_3 \rightarrow \neg(q \wedge r)$
A_5: p
A_6: $A_5 \Rightarrow q$
A_7: $A_4, A_6 \rightarrow \neg r$

We have that $\texttt{LastDefRules}(A_7) = \{d_1, d_3\}$ while $\texttt{LastDefRules}(B_3) = \{d_1, d_2\}$. Since $<$ is transitive we have $d_2 < d_1$ so $\{d_1, d_2\} \lhd_{\texttt{Eli}} \{d_1, d_3\}$ and $B_3 \prec A_7$. Hence A_7 successfully rebuts and thus strictly defeats B_3. But then A_7 also defeats B_4, B_5 and B_6.

Yet another relevant argument can be constructed, which starts in the same way as A_7:

A_3: $(q \wedge r) \supset \neg t$
A_4: $A_2, A_3 \rightarrow \neg(q \wedge r)$
A_5: p
A_6: $A_5 \Rightarrow q$
A_8: $A_5, A_6 \Rightarrow r$
A_9: $A_4, A_8 \rightarrow \neg q$

A_9 rebuts B_2 (and not vice versa). We have $\texttt{LastDefRules}(A_9) = \{d_2, d_3\}$ while $\texttt{LastDefRules}(B_2) = \{d_1\}$. Since $d_2 < d_1$ so $A_9 < B_2$ we have that A_9 does not defeat B_2. Since $A_6 = B_2$ we also have that A_9 does not defeat A_6. Finally, A_7 rebuts A_8. Recall that $\texttt{LastDefRules}(A_7) = \{d_1, d_3\}$; moreover, $\texttt{LastDefRules}(A_8) = \{d_2\}$ and we have seen that $\{d_2\} \lhd_{\texttt{Eli}} \{d_1, d_3\}$ so $A_8 \prec A_7$, for which reason A_7 strictly defeats A_8.

Now to evaluate the status of the arguments, A_7 and all its subarguments can be made *in* since they have no defeaters. Since A_7 strictly defeats A_8 and thus also A_9, the latter two arguments can be made *out*. Moreover since A_7 strictly defeats

B_3 and thus also B_4, B_5 and B_6, the latter four arguments can also be made *out*. No alternative status assignments are possible, while moreover the present assignment is complete. So B_6 is out in all preferred status assigments, which makes $\neg t$ an overruled conclusion.

2.
- $\texttt{Prem}(A_1) = \{s\}$, $\texttt{Conc}(A_1) = s$, $\texttt{Sub}(A_1) = \{A_1\}$, $\texttt{DefRules}(A_1) = \varnothing$ and $\texttt{TopRule}(A_1) =$ undefined.
- $\texttt{Prem}(A_2) = \{s\}, \texttt{Conc}(A_2) = t, \texttt{Sub}(A_2) = \{A_1, A_2\}, \texttt{DefRules}(A_2) = \{d_3\}$, $\texttt{LastDefRules}(A_2) = \{d_3\}$ and $\texttt{TopRule}(A_2) = d_3$.
- $\texttt{Prem}(A_3) = \{(q \wedge r) \supset \neg t\}, \texttt{Conc}(A_3) = (q \wedge r) \supset \neg t, \texttt{Sub}(A_3) = \{A_3\}$, $\texttt{DefRules}(A_3) = \varnothing$ and $\texttt{TopRule}(A_3) =$ undefined.
- $\texttt{Prem}(A_4) = \{s, (q \wedge r) \supset \neg t\}$, $\texttt{Conc}(A_4) = \neg(q \wedge r)$, $\texttt{Sub}(A_4) = \{A_1, A_2, A_3, A_4\}$, $\texttt{DefRules}(A_4) = \{d_3\}$, $\texttt{LastDefRules}(A_4) = \{d_3\}$ and $\texttt{TopRule}(A_4) = (q \wedge r) \supset \neg t, t \rightarrow \neg(q \wedge r)$.
- $\texttt{Prem}(A_5) = \{p\}$, $\texttt{Conc}(A_5) = p$, $\texttt{Sub}(A_5) = \{A_5\}$, $\texttt{DefRules}(A_5) = \varnothing$ and $\texttt{TopRule}(A_5) =$ undefined.
- $\texttt{Prem}(A_6) = \{p\}, \texttt{Conc}(A_6) = q, \texttt{Sub}(A_6) = \{A_5, A_6\}, \texttt{DefRules}(A_6) = \{d_1\}$, $\texttt{LastDefRules}(A_6) = \{d_1\}$ and $\texttt{TopRule}(A_6) = d_1$.
- $\texttt{Prem}(A_7) = \{s, p, (q \wedge r) \supset \neg t\}, \texttt{Conc}(A_7) = \neg r, \texttt{Sub}(A_7) = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7\}, \texttt{DefRules}(A_7) = \{d_1, d_3\}, \texttt{LastDefRules}(A_7) = \{d_1, d_3\}$ and $\texttt{TopRule}(A_7) = \neg(q \wedge r), q \rightarrow \neg r$.
- $\texttt{Prem}(A_8) = \{p\}, \texttt{Conc}(A_8) = r, \texttt{Sub}(A_8) = \{A_1, A_2, A_3, A_4, A_5, A_6, A_8\}$, $\texttt{DefRules}(A_8) = \{d_1, d_2\}, \texttt{LastDefRules}(A_8) = \{d_2\}$ and $\texttt{TopRule}(A_8) = d_2$.
- $\texttt{Prem}(A_9) = \{s, p, (q \wedge r) \supset \neg t\}, \texttt{Conc}(A_9) = \neg q, \texttt{Sub}(A_9) = \{A_1, A_2, A_3, A_4, A_5, A_6, A_8, A_9\}, \texttt{DefRules}(A_9) = \{d_1, d_2, d_3\}, \texttt{LastDefRules}(A_9) = \{d_2, d_3\}$ and $\texttt{TopRule}(A_9) = \neg(q \wedge r), r \rightarrow \neg q$.
- B_1, B_2, B_3 equal A_5, A_6, A_8.
- $\texttt{Prem}(B_4) = \{p\}$, $\texttt{Conc}(B_4) = q \wedge r$, $\texttt{Sub}(B_4) = \{B_1, B_2, B_3, B_4\}$, $\texttt{DefRules}(B_4) = \{d_1, d_2\}, \texttt{LastDefRules}(B_4) = \{d_1, d_2\}$ and $\texttt{TopRule}(B_4) = q, r \supset q \wedge r$.
- B_5 equals A_3.
- $\texttt{Prem}(B_6) = \{p, (q \wedge r) \supset \neg t\}, \texttt{Conc}(B_6) = \neg t, \texttt{Sub}(B_6) = \{B_1, B_2, B_3, B_4, B_5, B_6\}, \texttt{DefRules}(B_6) = \{d_1, d_2\}, \texttt{LastDefRules}(B_6) = \{d_1, d_2\}$ and $\texttt{TopRule}(B_6) = q \wedge r, (q \wedge r) \supset \neg t \rightarrow \neg t$.

EXERCISE 6.8.5.

1. It can be verified that there is no status assignment that assigns a status to A_2 or A_3.

   Firstly, to make A_2 *in*, its defeater A_3 must be *out*. To make A_3 *out*, one of its defeaters must be *in*. However, the only defeater of A_3 is A_3 itself (by undercutting its subargument A_2) and A_3 cannot be both *in* and *out*. So A_2 cannot be made *in*.

   Next, to make A_2 *out*, it must have a defeater that is *in*. Its only defeater is A_3. To make A_3 *in*, all its defeaters must be *out*. However, A_3 defeats itself and A_3 cannot be both *in* and *out*. So A_2 cannot be made *out*.

So there is only one preferred status assignment, in which A_1 is *in*, since A_1 has no defeaters. Moreover, this set is also the grounded extension.

2. Add $\texttt{Says}(\mathit{John}, \text{“}\texttt{StabbedWithKnife}(\mathit{Suspect}, \mathit{Victim})\text{”})$ to $\mathcal{K}_p$. Then the following argument can be constructed:

   B_1: $\texttt{Says}(\mathit{John}, \text{“}\texttt{StabbedWithKnife}(\mathit{Suspect}, \mathit{Victim})\text{”})$
   B_2: $\texttt{StabbedWithKnife}(\mathit{Suspect}, \mathit{Victim})$

   This argument is undercut by A_3. Since, as we have seen, no status assignment assigns a status to A_3, argument B_2 cannot have a status either. Then $E = \{A_1, B_1\}$ is the only preferred and grounded extension of the extended argumentation framework. Then according to preferred semantics B_2 is neither justified, nor defensible, nor overruled while according to grounded semantics it is defensible.

EXERCISE 6.8.6.

1. We have the following arguments:

A_1:	*injury*	B_1:	*medicalTests1*
A_2:	*appendicitis*	B_2:	$B_1 \Rightarrow$ *badCirculation*
A_3:	$A_2 \Rightarrow \neg$ *riskyOperation*	B_3:	$B_2 \Rightarrow$ *riskyOperation*
A_4:	$A_1, A_3 \Rightarrow$ *negligence*		
A_5:	$A_1, A_4 \Rightarrow$ *compensation*	C_1:	*medicalTests2*
		C_2:	$C_1 \Rightarrow \neg$ *badCirculation*

   Their attack relations are shown in Figure 9.4.

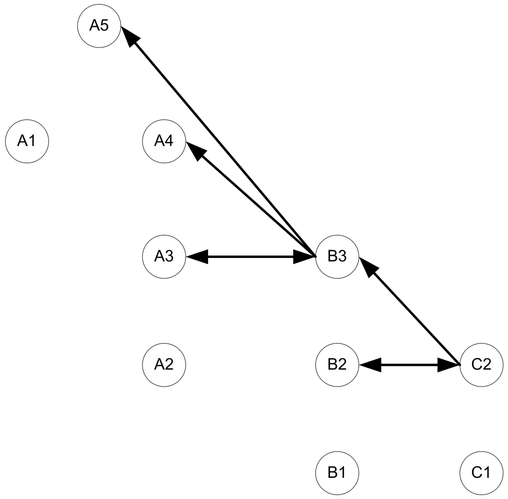


Figure 9.4: Abstract attack graph

2.
   - $\texttt{Prem}(A_1) = \{f_1\}, \texttt{Conc}(A_1) = \mathit{injury}, \texttt{Sub}(A_1) = \{A_1\}, \texttt{DefRules}(A_1) = \varnothing$ and $\texttt{TopRule}(A_1) =$ undefined.
   - $\texttt{Prem}(A_2) = \{f_2\}, \texttt{Conc}(A_2) = \mathit{appendicitis}, \texttt{Sub}(A_2) = \{A_2\}, \texttt{DefRules}(A_2) = \varnothing$ and $\texttt{TopRule}(A_2) =$ undefined.

- $\texttt{Prem}(A_3) = \{f_2\}$, $\texttt{Conc}(A_3) = \neg \mathit{riskyOperation}$, $\texttt{Sub}(A_3) = \{A_2, A_3\}$, $\texttt{DefRules}(A_3) = \{r_3\}$ and $\texttt{TopRule}(A_3) = r_3$.
- $\texttt{Prem}(A_4) = \{f_1, f_2\}$, $\texttt{Conc}(A_4) = \mathit{negligence}$, $\texttt{Sub}(A_4) = \{A_1, A_2, A_3, A_4\}$, $\texttt{DefRules}(A_4) = \{r_2, r_3\}$ and $\texttt{TopRule}(A_4) = r_2$.
- $\texttt{Prem}(A_5) = \{f_1, f_2\}$, $\texttt{Conc}(A_5) = \mathit{compensation}$, $\texttt{Sub}(A_5) = \{A_1, A_2, A_3, A_4, A_5\}$, $\texttt{DefRules}(A_5) = \{r_1, r_2, r_3\}$ and $\texttt{TopRule}(A_5) = r_1$.
- $\texttt{Prem}(B_1) = \{f_3\}$, $\texttt{Conc}(B_1) = \mathit{medicalTests1}$, $\texttt{Sub}(B_1) = \{B_1\}$, $\texttt{DefRules}(B_1) = \varnothing$ and $\texttt{TopRule}(B_1) = \text{undefined}$.
- $\texttt{Prem}(B_2) = \{f_3\}$, $\texttt{Conc}(B_2) = \mathit{badCirculation}$, $\texttt{Sub}(B_2) = \{B_1, B_2\}$, $\texttt{DefRules}(B_2) = \{r_5\}$ and $\texttt{TopRule}(B_2) = r_5$.
- $\texttt{Prem}(B_3) = \{f_3\}$, $\texttt{Conc}(B_3) = \mathit{riskyOperation}$, $\texttt{Sub}(B_3) = \{B_1, B_2, B_3\}$, $\texttt{DefRules}(B_3) = \{r_4, r_5\}$ and $\texttt{TopRule}(B_3) = r_4$.
- $\texttt{Prem}(C_1) = \{f_4\}$, $\texttt{Conc}(C_1) = \mathit{medicalTests2}$, $\texttt{Sub}(C_1) = \{C_1\}$, $\texttt{DefRules}(C_1) = \varnothing$ and $\texttt{TopRule}(C_1) = \text{undefined}$.
- $\texttt{Prem}(C_2) = \{f_4\}$, $\texttt{Conc}(C_2) = \neg \mathit{badCirculation}$, $\texttt{Sub}(C_2) = \{C_1, C_2\}$, $\texttt{DefRules}(C_2) = \{r_6\}$ and $\texttt{TopRule}(C_2) = r_6$.

3. We have that $\texttt{LastDefRules}(A_3) = \{r_3\}$ while $\texttt{LastDefRules}(B_3) = \{r_4\}$ and since $r_3 < r_4$ we have that $\texttt{LastDefRules}(A_3) \triangleleft_{\texttt{Eli}} \texttt{LastDefRules}(B_3)$, so $A_3 \prec B_3$, so B_3 strictly defeats A_3.

   Moreover, we have that $\texttt{LastDefRules}(B_2) = \{r_5\}$ while $\texttt{LastDefRules}(C_2) = \{r_6\}$ and since $r_5 < r_6$ we have that $\texttt{LastDefRules}(B_2) \triangleleft_{\texttt{Eli}} \texttt{LastDefRules}(C_2)$, so $B_2 \prec C_2$, so C_2 strictly defeats B_2.

   The other attack relations succeed as defeats. The resulting defeat relations are shown in Figure 9.5.

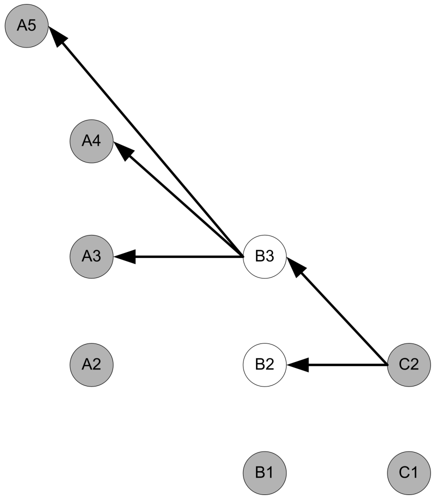


Figure 9.5: Abstract argumentation theory with grounded and unique preferred labelling

4. Figure 9.5 shows the grounded labelling: the arguments that are *in* are coloured gray, the arguments that are *out* are coloured white. The grounded extension consists of all arguments that are labelled *in*.

5. Since the abstract argumentation theory depicted in Figure 9.5 is finite and has no cycles, all semantics give the same result. So the grounded extension is also the unique preferred (and stable) extension.

6. We now also have that $\texttt{Prem}_p(A_3) = \varnothing$ while $\texttt{Prem}_p(B_3) = \{f_3\}$ so we have that $\texttt{Prem}_p(B_3) \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(A_3)$. Then we have that $A_3 \not\prec B_3$ and $B_3 \not\prec A_3$ so A_3 and B_3 now defeat each other.

   Moreover, we now also have that $\texttt{Prem}_p(B_2) = \{f_3\}$ while $\texttt{Prem}_p(C_2) = \{f_4\}$, so since $f_4 <' f_3$, we have that $\texttt{Prem}_p(C_2) \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(B_2)$. Then we have that $B_2 \not\prec C_2$ and $C_2 \not\prec B_2$ so B_2 and C_2 now also defeat each other. So the defeat relations now equal the attack relations as displayed in Figure 9.4. Then there are three preferred labellings: the original one displayed in Figure 9.5 and two new ones displayed in, respectively Figure 9.6 and Figure 9.7. The two new preferred extensions consist, respectively, of the sets of argument labelled *in* in these two preferred labellings.

   The grounded labelling now makes A_1, A_2, B_1 and C_1 *in* and the remaining arguments undecided. So the grounded extension is $\{A_1, A_2, B_1, C_1\}$.

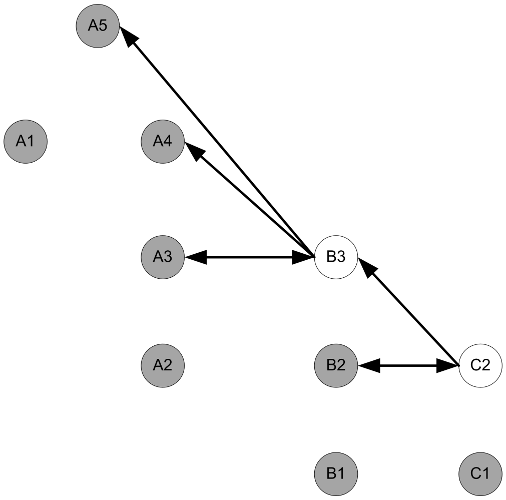


Figure 9.6: Abstract argumentation theory with a second preferred labelling

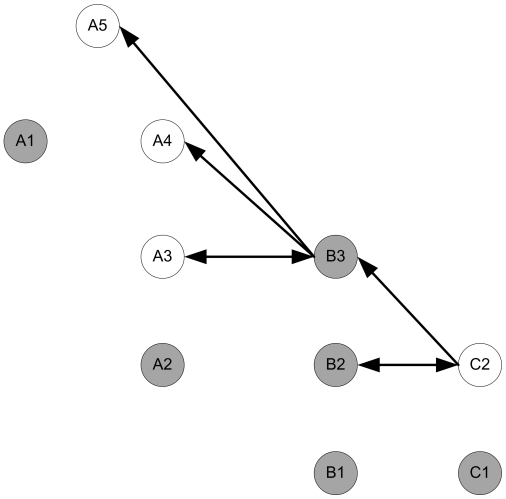


Figure 9.7: Abstract argumentation theory with a third preferred labelling

EXERCISE 6.8.7: see Figure 9.8.

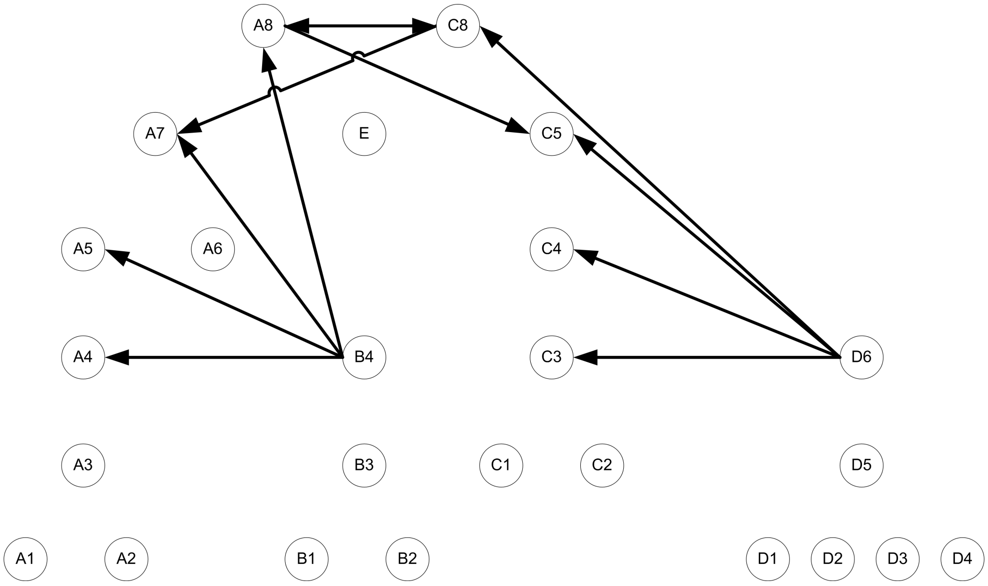


Figure 9.8: Abstract argumentation theory for Figure 6.4

EXERCISE 6.8.8

1. $\mathcal{K}_p$ consists of:

$\forall x(\texttt{BornInNL}(x) \rightsquigarrow \texttt{Dutch}(x))$
$\forall x(\texttt{NorwegianName}(x) \rightsquigarrow \texttt{Norwegian}(x))$
$\forall x((\texttt{Dutch}(x) \vee \texttt{Norwegian}(x)) \rightsquigarrow \texttt{LikesIceSkating}(x))$
$\texttt{BorninNL}(b)$
$\texttt{NorwegianName}(b)$
$\forall x \neg(\texttt{Dutch}(x) \wedge \texttt{Norwegian}(x))$

The following relevant arguments can be constructed:

$$\begin{array}{ll}
A_1\text{:} & \texttt{BorninNL}(b) \\
A_2\text{:} & \forall x\,(\texttt{BornInNL}(x) \leadsto \texttt{Dutch}(x)) \\
A_3\text{:} & A_2 \rightarrow \texttt{BornInNL}(b) \leadsto \texttt{Dutch}(b) \\
A_4\text{:} & A_1, A_3 \Rightarrow \texttt{Dutch}(b) \\
A_5\text{:} & A_4 \rightarrow \texttt{Dutch}(b) \vee \texttt{Norwegian}(b) \\
A_6\text{:} & \forall x((\texttt{Dutch}(x) \vee \texttt{Norwegian}(x)) \leadsto \texttt{LikesIceSkating}(x)) \\
A_7\text{:} & A_6 \rightarrow (\texttt{Dutch}(b) \vee \texttt{Norwegian}(b)) \leadsto \texttt{LikesIceSkating}(b) \\
A_8\text{:} & A_5, A_7 \Rightarrow \texttt{LikesIceSkating}(b)
\end{array}$$

$$\begin{array}{ll}
B_1\text{:} & \texttt{BorninNL}(b) \\
B_2\text{:} & \forall x\,(\texttt{BornInNL}(x) \leadsto \texttt{Dutch}(x)) \\
B_3\text{:} & B_2 \rightarrow \texttt{BornInNL}(b) \leadsto \texttt{Dutch}(b) \\
B_4\text{:} & B_1, B_3 \Rightarrow \texttt{Dutch}(b) \\
B_5\text{:} & \forall x \neg(\texttt{Dutch}(x) \wedge \texttt{Norwegian}(x)) \\
B_6\text{:} & B_4, B_5 \rightarrow \neg\texttt{Norwegian}(b)
\end{array}$$

$$\begin{array}{ll}
C_1\text{:} & \texttt{NorwegianName}(b) \\
C_2\text{:} & \forall x\,(\texttt{NorwegianName}(x) \leadsto \texttt{Norwegian}(x)) \\
C_3\text{:} & C_2 \rightarrow \texttt{NorwegianName}(b) \leadsto \texttt{Norwegian}(b) \\
C_4\text{:} & C_1, C_3 \Rightarrow \texttt{Norwegian}(b) \\
C_5\text{:} & C_4 \rightarrow \texttt{Dutch}(b) \vee \texttt{Norwegian}(b) \\
C_6\text{:} & \forall x((\texttt{Dutch}(x) \vee \texttt{Norwegian}(x)) \leadsto \texttt{LikesIceSkating}(x)) \\
C_7\text{:} & C_6 \rightarrow (\texttt{Dutch}(b) \vee \texttt{Norwegian}(b)) \leadsto \texttt{LikesIceSkating}(b) \\
C_8\text{:} & C_5, C_7 \Rightarrow \texttt{LikesIceSkating}(b)
\end{array}$$

$$\begin{array}{ll}
D_1\text{:} & \texttt{NorwegianName}(b) \\
D_2\text{:} & \forall x\,(\texttt{NorwegianName}(x) \leadsto \texttt{Norwegian}(x)) \\
D_3\text{:} & D_2 \rightarrow \texttt{NorwegianName}(b) \leadsto \texttt{Norwegian}(b) \\
D_4\text{:} & D_1, D_3 \Rightarrow \texttt{Norwegian}(b) \\
D_5\text{:} & \forall x \neg(\texttt{Dutch}(x) \wedge \texttt{Norwegian}(x)) \\
D_6\text{:} & D_4, D_5 \rightarrow \neg\texttt{Dutch}(b)
\end{array}$$

(If the example is formalised in a propositional language, then the steps A_7 and C_7 must be omitted.)

2. Note first that if no preference relation is specified, it does not hold. Then the relevant defeat relations are as follows:

   - B_6 defeats C_4 and thus also $C_5 - C_8$
   - D_6 defeats B_4 and thus also B_5 and B_6
   - D_6 defeats A_4 and thus also $A_5 - A_8$
   - B_6 defeats D_4 and thus also D_5 and D_6

   Let us first concentrate on B_6 and D_6. Since they defeat each other and have no other defeaters, it is possible to assign no status to them. Then in the grounded status assignments they have no status. But then the same holds for the arguments defeated by one of them. This includes A_8 and C_8. Hence the conclusion $\texttt{LikesIceSkating}(b)$ only has defensible arguments and is therefore itself defensible.

(The same answer in terms of the fixpoint definition: Since B_6 and D_6 defeat each other and have no other defeaters, they are in no F^i. But then the arguments defeated by one of them also are in no F^i.)

3. Let us again first concentrate on B_6 and D_6. Argument B_6 can be made *in* by making D_6 *out* and vice versa. Then there is a preferred status assignment in which B_6 is *in* and D_6 is *out*. In this status assignment also $C_4 - C_8$ are *out* and $A_1 - A_8$ are *in*. So an argument for the conclusion $\texttt{LikesIceSkating}(b)$ is *in*, namely, A_8. Conversely, there is also a preferred status assignment in which D_6 is *in* and B_6 is *out*. In this status assignment also $A_4 - A_8$ are *out* and $C_1 - C_8$ are *in*. So again an argument for the conclusion $\texttt{LikesIceSkating}(b)$ is *in* but this time it is not A_8 but C_8. So both A_8 and C_8 are defensible, so the conclusion $\texttt{LikesIceSkating}(b)$ is also defensible.

4. Since both preferred extensions contain an argument for the conclusion $\texttt{LikesIceSkating}(b)$, this conclusion is f-justified, even though there is no justified argument for it.

EXERCISE 6.8.9 The following formalisation is based on the intuition that the conclusion that Larry is not rich is justified. The undercutters in the example are based on the principle that statistical defaults about subclasses have priority over statistical defaults about superclasses.

$\mathcal{R}_s$ consists of all valid propositional and first-order inferences.

$\mathcal{R}_d$ consists of:

d_1. $\texttt{Lawyer}(x) \Rightarrow \texttt{Rich}(x)$
d_2. $\texttt{LivesInHollywood}(x) \Rightarrow \texttt{Rich}(x)$
d_3. $\texttt{PublicDefender}(x) \Rightarrow \neg\, \texttt{Rich}(x)$
d_4. $\texttt{RentsinHollywood}(x) \Rightarrow \neg\, \texttt{Rich}(x)$
d_5. $\texttt{PublicDefender}(x) \Rightarrow \neg d_1(x)$
d_6. $\texttt{RentsinHollywood}(x) \Rightarrow \neg d_2(x)$

$\mathcal{K}_p$ consists of

p_1. $\texttt{PublicDefender}(L)$
p_2. $\texttt{RentsInHollywood}(L)$

$\mathcal{K}_n$ consists of

n_1. $\forall x(\texttt{PublicDefender}(x) \supset \texttt{Lawyer}(x))$
n_2. $\forall x(\texttt{RentsInHollywood}(x) \supset \texttt{LivesInHollywood}(x))$

The following relevant arguments can be constructed:

A_1: $\texttt{PublicDefender}(L)$
A_2: $\forall x(\texttt{PublicDefender}(x) \supset \texttt{Lawyer}(x))$
A_3: $A_1, A_2 \rightarrow \texttt{Lawyer}(L)$
A_4: $A_3 \Rightarrow \texttt{Rich}(L)$

B_1: $\texttt{PublicDefender}(L)$
B_2: $B_1 \Rightarrow \neg\texttt{Rich}(L)$

C_1: $\texttt{RentsInHollywood}(L)$
C_2: $\forall x(\texttt{RentsInHollywood}(x) \supset \texttt{LivesInHollywood}(x))$
C_3: $C_1, C_2 \rightarrow \texttt{LivesInHollywood}(L)$
C_4: $C_3 \Rightarrow \texttt{Rich}(L)$

D_1: $\texttt{RentsInHollywood}(L)$
D_2: $B_1 \Rightarrow \neg\texttt{Rich}(L)$

E_1: $\texttt{PublicDefender}(L)$
E_2: $E_1 \Rightarrow \neg d_1(L)$

F_1: $\texttt{RentsInHollywood}(L)$
F_2: $F_1 \Rightarrow \neg d_2(L)$

Let us apply preferred semantics (but in grounded semantics the outcome is the same). Note first that E_2 undercuts A_4 and F_2 undercuts C_4. Moreover, neither E_2 nor F_2 has a defeater, so both of them are in all preferred extensions. But then A_4 and C_4 are not in any preferred extension, so that B_2 and D_2 are in all these extensions. So the conclusion $\neg\texttt{Rich}(L)$ is justified.

EXERCISE 6.8.10.

1. $Cl_{tp}(R_s) = R_s \cup \{-q \rightarrow -p;\ \ - r \rightarrow -p;\ p, -s \rightarrow -r;\ r, -s \rightarrow -p\}$.
2. Yes.
3. No.

EXERCISE 6.8.11. The point of this exercise is that closure under contraposition does not imply closure under transposition.

1. No: $\mathcal{R}_s$ contains $p \rightarrow q$ but not $\neg q \rightarrow \neg p$.
2. Yes. We have:

   $\{p\} \vdash q$ and $\{\neg q\} \vdash \neg p$
   $\{p\} \vdash \neg r$ and $\{r\} \vdash \neg p$
   $\{\neg r\} \vdash q$ and $\{\neg q\} \vdash r$
   $\{\neg q\} \vdash r$ and $\{\neg r\} \vdash q$

   So an argumentation theory with $\mathcal{R}_s$ satisfies contraposition.

EXERCISE 6.8.12.

1. The arguments (shown in Figure 9.9, with their conclusions at the bottom) are:

   $A' = a$,

   $A = A' \Rightarrow p$,

   $B_1 = \sim s$,

   $B_1' = B_1 \Rightarrow t$,

   $B_2 = r$,

   $B_2' = B_2 \Rightarrow q$,

   $B = B_1'$,

   $B_2' \rightarrow \neg p$,

   $C = [\neg r]$.

2. B rebuts A on A, C undermines B and B_2' on B_2, and C and B_2 undermine each other. Note that A does not rebut B since B has a strict top rule.
3. We have that $\texttt{LastDefRules}(B) = \{d_1, d_2\}$ and $\texttt{LastDefRules}(A) = \{d_3\}$ and since $d_2 < d_3$, we have that $\texttt{LastDefRules}(B) \triangleleft_{\texttt{Eli}} \texttt{LastDefRules}(A)$. So B does not defeat A. Moreover, we have that $\texttt{Prem}_p(C) = \{\neg r\}$ while $\texttt{Prem}_p(B_2) = \{r\}$ and $\neg r <' r$, so we also have that $\texttt{Prem}_p() \triangleleft_{\texttt{Eli}} \texttt{Prem}_p(B_2)$. So $C \prec B_2$ so B_2 strictly defeats C.
4. The transpositions are $p, t \rightarrow \neg q$ and $p, q \rightarrow \neg t$. This yields two new arguments:

   $D = A, B_1' \Rightarrow \neg q$,

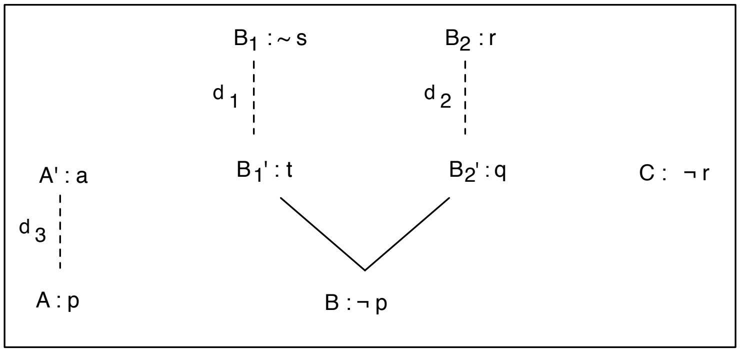


Figure 9.9: *ASPIC*$^+$ arguments and their conclusions, with dashed and solid lines respectively representing application of defeasible and strict inference rules.

$E = A, B_2' \Rightarrow \neg t$.

D rebuts B_2' while E rebuts B_1'.

We have that $\texttt{LastDefRules}(D) = \{d_1, d_3\}$ and $\texttt{LastDefRules}(B_2') = \{d_2\}$ and since $d_2 < d_3$, we have that $\texttt{LastDefRules}(B_2') \lhd_{\texttt{Eli}} \texttt{LastDefRules}(D)$. So D strictly defeats B_2'.

Moreover, we have that $\texttt{LastDefRules}(E) = \{d_2, d_3\}$ and $\texttt{LastDefRules}(B_1') = \{d_1\}$ and since $d_1 \not< d_2$ and $d_1 \not< d_2$ and $d_2 \not< d_3$, we have that these sets are incomparable in the $\lhd_{\texttt{Eli}}$ ordering. So E and B_1' defeat each other.

EXERCISE 6.8.13.

1. The arguments are

   A_1: $\sim a$
   A_2: $A_1 \rightarrow b$
   A_3: $A_2 \Rightarrow \neg c$

   B_1: $\Rightarrow c$
   B_2: $B_1 \Rightarrow a$

   Argument B_2 contrary-undermines A_1, A_2 and A_3 on A_1. Argument A_3 rebuts B_1 and B_2 on B_1. Finally, B_1 rebuts A_3 on A_3.

2. The attack of B_2 on A_1, A_2 and A_3 succeeds since contrary undermining is a preference-independent form of attack. Moreover, we have that $\texttt{DefRules}(B_1) = \{d_2\}$ and $\texttt{DefRules}(A_3) = \{d_1\}$ and since $d_2 < d_1$, we have that $\texttt{DefRules}(B_1) \lhd_{\texttt{Eli}} \texttt{DefRules}(A_3)$, so $B_1 \prec A_3$. So A_3 strictly defeats B_1 and B_2.

3. The grounded extension is empty, since there are no undefeated arguments.

4. There are two preferred extensions: the first is $\{A_1, A_2, A_3\}$ while the second is $\{B_1, B_2\}$.

EXERCISE 6.8.14.

1. C_2 rebuts D_2 and not vice versa. Since both arguments use defeasible rules and no preference relations hold between them, C_2 successfully rebuts and therefore defeats D_2. Argument C_2 in turn has two defeaters: its subarguments A_2 and B_2 defeat each other and thus also defeat C_2. Since there are no undefeated arguments that defeat A_2 or B_2, none of A_2, B_2, C_2 and D_2 are in the grounded extension. (In terms of status assignments: it is possible to give none of them a status so in the grounded extension, which maximises undecidedness, none of them have a status.) However, none of these arguments are defeated by an argument that is in the grounded extension, so they are all defensible.

2. Note that A_2 can be made *in* if B_2 is made *out* and vice versa. Then at least one preferred status assignment makes A_2 *in* and B_2 *out*, since such assignments minimise undecidedness. But since A_2 defeats C_2, this assignment also makes C_2 *out*. But then it makes D_2 *in*, since its only defeater is C_2. Conversely, a second preferred status assignment makes B_2 *in* and A_2 *out* so it also makes C_2 *out* and D_2 *in*. Since there are no other preferred status assignments, in all such assigments C_2 is *out* and D_2 is *in*. But then C_2 is overruled and D_2 is justified.

EXERCISE 6.8.15.

1. No. We explain this with the G-game. There is an argument for $guilty$, namely

   $A = murder, murder \supset guilty \rightarrow guilty.$

   Argument A has two strict defeaters, namely:

   $B = \neg ab, \neg ab \supset \neg guilty, murder \supset guilty \rightarrow \neg murder$

   $C = \neg ab, \neg ab \supset \neg guilty, murder \rightarrow \neg(murder \supset guilty)$

   Since $\mathcal{K}_p$ is minimally inconsistent (i.e., taking any element out makes $\mathcal{K}_p$ consistent), both B and C have underminers on any of their premises: these underminers can be formed by replacing the attacked premise with the remaining one. Since the argument ordering is simple, all these undermining attacks succeed as defeats. For example, B is defeated on $\neg ab$ by

   $D = murder, \neg ab \supset \neg guilty, murder \supset guilty \rightarrow ab$

   In the same way, any further argument moved in a G-game has defeaters, so the proponent does not have a winning strategy for A.

2. Any argument ordering in which ab is inferior to all other formulas in $\mathcal{K}_p$ will do, since then neither B not C defeats A, so the proponent wins the G-game after moving A.

3. Move all formulas except ab to $\mathcal{K}_n$. Then argument A has no attackers since all its premises are necessary.

9.5 Exercises Chapter 7

Exercise 7.4.1:

1. Yes, since there is just one preferred extension, namely, $\{B\}$.
2. No. if the attack from B to A is deleted, then the preferred extension is empty.

Exercise 7.4.2:

1. In (a) D is justified in all full resolutions. One full resolution deletes the attack from B to C and another full resolution deletes the attack from C to B. In both cases the grounded extension is $\{A, D\}$.

   In (b) D is justified in some but not all full resolutions. Any full resolution which deletes the attack from A to D makes D a member of the grounded extension. But a full resolution that deletes the attacks from D to A and B to A makes instead A a member of the grounded extension.

   In (e) D is also justified in some but not all full resolutions. If the attack from C to B is deleted, then D is in the grounded extension but if the attack from B to C is deleted then instead C is in the grounded extension.

2. All answers are the same for preferred semantics.

Exercise 7.4.3:

1. let $\mathcal{R}_s = \{p \to \neg q\}$, $\mathcal{R}_d = \mathcal{K}_n = \varnothing$ and $\mathcal{K}_p = \{q\}$. Then $A = p \to \neg q$ asymmetrically attacks $B = q$. With the elitist last- or weakest- link ordering, the resolution that adds $p <' q$ deletes this attack.

2. Let $(\mathcal{L}, ^-, \mathcal{R}, n)$ be an argumentation system where:

   - $\mathcal{L}$ is a language of propositional literals, composed from a set of propositional atoms $\{a, b, c, \ldots\}$ and the symbols $\neg$ and $\sim$ respectively denoting strong and weak negation (i.e., negation as failure). α is a strong literal if α is a propositional atom or of the form $\neg\beta$ where β is a propositional atom. α is a wff of $\mathcal{L}$, if α is a strong literal or of the form $\sim \beta$ where β is a strong literal.
   - For any wff α, α and $\neg\alpha$ are contradictories and α is a contrary of $\sim \alpha$.
   - $\mathcal{R}_s = \varnothing$, $\mathcal{R}_d = \{\neg c \Rightarrow \neg b; a, b \Rightarrow c\}$, and $\leq = \approx$ (since partial preorders are reflexive $\leq = \approx$ denotes $\{r \leq r | r \in \mathcal{R}_d\}$)

   $\mathcal{K}$ is the knowledge base such that $\mathcal{K}_n = \varnothing$, $\mathcal{K}_p = \{a, b, \neg c\}$, $\mathcal{K}_a = \varnothing$, and $\leq' = \{a <' \neg c <' b\}$.

   We obtain arguments $X = [\neg c; \neg c \Rightarrow \neg b]$ and $Y = [a; b; a, b \Rightarrow c]$. Then X attacks Y on $Y' = [b]$, and Y attacks X on $X' = [\neg c]$. Then the elitist weakest and last link principles give that $X \prec Y'$ and $Y \prec X'$. Hence neither X or Y defeat each other.

3. Consider any theory with $\mathcal{K}_p = \{p, \neg p\}$ and $p \approx \neg p$. Then the arguments p and $\neg p$ defeat each other and no preference extension can change this.

Exercise 7.4.4: Nothing changes. With the elitist last-link ordering the attack of B_3 on A_2 is preference independent. Moreover, only the preference between d_4 and d_2 is relevant for the conflict between B_2 and C_3 and we already have that $d_2 < d_4$. With elitist weakest-link two comparisons are relevant. The first is between $\{d_2\}$ and $\{d_3, d_4\}$ and here there are three strict preference relations between the three elements, which a preference extension cannot change. The second comparison is between u and s, between which a strict preference already exists.

Exercise 7.4.5:

1. Yes. A is undermined by $B = p, q \to (p \wedge q)$. We have $\texttt{Prem}_p(A) = \{\neg(p \wedge q)\}$ while $\texttt{Prem}_p(B) = \{p, q\}$ and since $p <' \neg(p \wedge q)$ but the relation between q and $\neg(p \wedge q)$ is undefined, we have that $\texttt{Prem}_p(B) \lhd_{\texttt{Eli}} \texttt{Prem}_p(A)$. So $B \prec A$, so B does not defeat A. Since A has no other defeaters, B is in the grounded extension.

   Note that we have the following orderings between the various premise sets:

   $\{p, q\} \lhd_{\texttt{Eli}} \{\neg(p \wedge q)\}$

   $\{p\} \lhd_{\texttt{Eli}} \{q, \neg(p \wedge q)\}$

   $\{p, \neg(p \wedge q)\} \lhd_{\texttt{Eli}} \{q\}$

2. There are three ways to extend $\leq$: with $q <' \neg(p \wedge q)$, with $\neg(p \wedge q) <' q$ and with $q \approx' \neg(p \wedge q)$. In all three cases the above orderings between the various premise sets does not change. So the set $\mathcal{D}$ of defeat relations does not change, so there exists no full preference-based resolution. So the answer is 'yes'.

9.6 Exercises Chapter 8

EXERCISE 8.6.1

- Topic language: arguments without structure.
- Communication language: utterances of arguments.
- Dialogue purpose: test the dialectical status of an argument.
- Participants: proponent and opponent. Both have knowledge of the same set of arguments and the same defeat relation. (More precisely, we assume that both have a knowledge base that gives rise to exactly this set of arguments.) They have no commitments.
- Logic: skeptical reasoning in grounded semantics the G-game, creduloous reasoning in preferred semantics for the P-game.
- Effect rules: none.
- Protocol: Definition 5.2.1(1) for the G-game and Definition 5.3.7(1) for the P-game.
- Outcome rules: Protocol: Definition 5.2.1(2) for the G-game and Definition 5.3.7(2) for the P-game.

EXERCISE 8.6.2

The elements of Definition 8.3.1 must be instantiated as follows (elements for which any instantiation is allowed are not listed):

- Dialogue purpose: T consists of two propositions t and $\neg t$.
- Participants: two players, one of whom is proponent of t and opponent of $\neg t$ and the other is proponent of $\neg t$ and opponent of t.

EXERCISE 8.6.3

1. - p is overruled. It has one argument, viz. $A_1 = (\{q, q \supset p\}, p)$, which has two attackers, viz. $A_2 = (\{\neg p, q \supset p\}, \neg q)$ and $A_3 = (\{q, \neg p\}, \neg(q \supset p))$, and A_3 makes A_1 overruled as follows. We have $\mathit{Level}(\{q, q \supset p\}) > \mathit{Level}(\{q, \neg p\})$, so A_3 defeatsA_1. Next, A_3 has two attackers, viz. A_1 and A_2. We already saw that A_3 is preferred over A_1 and, moreover, A_3 is also preferred over A_2, so A_3 strictly defeats both A_1 and A_2. Since A_3 has no other attackers, A_3 is justified and A_1 is overruled.

   - $\neg p$ is justified. The argument $A_4 = (\{\neg p\}, \neg p)$ has one attacker, viz. A_1, but $\mathit{Level}(\{q, q \supset p\}) > \mathit{Level}(\{\neg p\})$, so A_1 does not defeat A_4.

   - $q \supset p$ is overruled. It has one argument, viz. $A_5 = (\{q \supset p\}, q \supset p)$, which has one attacker, viz. A_3. We have $\mathit{Level}(\{q \supset p\}) > \mathit{Level}(\{\neg p, q\})$, so A_3 strictly defeats A_5. Since we saw under (1) that A_3 has no defeaters, A_5 is overruled.

2. There is just one legal dialogue, viz.

   W_1: *claim* p B_1: *why* p
   W_2: *claim* $\{q, q \supset p\}$ B_2: *concede* q; B_3: *claim* $\neg(q \supset p)$
   W_4: *concede* $\neg(q \supset p)$

   Let us explain why. At his first move, W must reason with Σ_W, which contains a justified argument for p (it has no attackers on the basis of Σ_W). Then B at her first move must reason with $\Sigma_B \cup \{p\}$. Then B cannot concede p: although she can construct an argument for p, viz. $A_6 = (\{p\}, p)$, it is not justified: B can construct $A_4 = (\{\neg p\}, \neg p)$, which defeats A_6 and is not defeated by other arguments on the basis of $\Sigma_B \cup \{p\}$. Can B claim $\neg p$? No, since her only argument for $\neg p$ is A_4, which is defeated by A_6 and since these arguments have no other attackers on the basis of $\Sigma_B \cup \{p\}$, they are both defensible on the basis of $\Sigma_B \cup \{p\}$. So B must challenge p. After W's reply with W_2 the information with which B must reason is $\Sigma_B \cup \{p, q, q \supset p\}$. On this basis B's only argument for $\neg q$ is $A_2 = (\{\neg p, q \supset p\}, \neg q)$ but we have $\mathit{Level}(\{q\}) < \mathit{Level}(\{\neg p, q \supset p\})$ so A_2 is overruled on the basis of $\Sigma_B \cup \{p, q, q \supset p\}$. So B must concede q. However, she has a justified argument against W_2's second premise, viz. A_3, so she must claim its negation. Then W must reason with $\Sigma_W \cup \{\neg(q \supset p)\}$: this supports a trivial argument for $q \supset p$ but since $q \supset p \prec \neg(q \supset p)$ we have that W must concede and the dialogue terminates.

   At termination, the commitment sets are:

   $C_W(d_6) = \{p, q, q \supset p, \neg(q \supset p)\}$, which is inconsistent;
   $C_B(d_6) = \{q, \neg(q \supset p)\}$, which is consistent.

On the basis of $\Sigma_W \cup C_W(d_6)$ we have that p and $\neg p$ are defensible because of the arguments A_6 and $A_8 = (\{\neg(q \supset p)\}, \neg p)$. Since p and $\neg(q \supset p)$ are at the same preference level, both arguments defeat each other and since they have no other defeaters, they are defensible.

On the basis of $\Sigma_B \cup C_B(d_6)$ we have that $\neg p$ is justified since it has two justified arguments A_4 and A_8.

In sum, even though on the basis of the players' joint beliefs p is overruled and $\neg p$ is justified, the players do not reach agreement on p.

3. The only legal dialogue now is

W_1: *claim p*	B_1: *claim* $\neg p$

Here the dialogue terminates since W cannot repeat *claim p*. At termination W is committed to p and B to $\neg p$. These sets are both internally consistent and consistent with the agents' own beliefs. Finally, p is justified on the basis of $\Sigma_W \cup C_W(d_2)$ while $\neg p$ is justified on the basis of $\Sigma_B \cup C_B(d_2)$.

EXERCISE 8.6.4

1. The only legal dialogue is

W_1: *claim p*	B_1: *why p*
W_2: *claim* $\{q, q \supset p\}$	B_2: *why q*
W_3: *claim q*	

This dialogue terminates without agreement, so B has learned nothing from W.

2. Any player can accept a proposition φ after a *claim* $\{\varphi\}$ move of the other player that was moved after a *why* φ move, provided that the player cannot construct an argument for $\neg\varphi$.

EXERCISE 8.6.5 Assuming the above answer to 8.6.4(2), the only legal dialogue is

W_1: *claim r*	B_1: *why r*
W_2: *claim* $\{p, p \supset q, q \supset r\}$	B_2: *why p*
W_3: *claim* $\{p\}$	B_3: *concede p*, B_4: *claim* $\neg(p \supset q)$
W_4: *claim* $p \supset q$	B_5: *why* $p \supset q$
W_5: *claim* $\{p \supset q\}$	B_6: *concede* $q \supset r$

This exercise illustrates a number of subtle features of the PWA protocol. Note first that black could make his counterclaim only after first conceding p! Next, at B_5 black could not claim $\neg(p \supset q)$ even though that is allowed by her assertion attitude, since this claim repeats B_4. So black had to challenge[1]. Finally, the reason why black must concede $q \supset p$ is that she has a justified argument for it with premises $\{s, s \supset \neg q\}$, which implies not only $\neg q$ but also $q \supset r$ for any r!

EXERCISE 8.6.6

[1] When read literally, PWA's termination condition "when the move required by the procedure cannot be made" implies that the dialogue terminates here, but we read it as meaning that only the 'sub-dialogue' about the first premise of W_2 terminates and the dialogue then continues about the second premise.

1. A counterexample is Example 8.4.4.

2. A counterexample is $\Sigma_W = \{p, p \supset q, r\}$ and $\Sigma_B = \{r \supset \neg p\}$, with topic q and all formulas of the same preference level. The only legal dialogue on q is:

W_1: *claim* q	B_1: *why* q
W_2: *claim* $\{p, p \supset q\}$	B_2: *why* p
W_3: *claim* $\{p\}$	B_3: *concede* p, B_4: *why* $p \supset q$
W_4: *claim* $\{p \supset q\}$	B_5: *concede* $\{p \supset q\}$

$C_W(d_9) \vdash q$ and $C_B(d_9) \vdash q$ but q is not justified on the basis of $\Sigma_W \cup \Sigma_B$ because of the counterargument $(\{r, r \supset \neg p\}, \neg p)$.

EXERCISE 8.6.7. This follows from result (2) of Section 4.4 of the reader, which implies that finite defeat graphs without cycles have a unique status assignments. (Note that dialogue trees have no such cycles through their reply relations.)

EXERCISE 8.6.8. A surrendered move is *in* by definition regardless of its other replies, so a new reply can never change any dialogical status.

EXERCISE 8.6.9

1. P_1, P_3 and P_7

2. O_2 and O_8

3. P_1, P_9

EXERCISE 8.6.10 For example:

$P_1 =$ *claim* q
$O_1 =$ *why* q
$P_2 = q$ *since* $p, p \supset q$
$O_2 = \neg p$ *since* $r, r \supset \neg p$
$P_3 = \neg(r \supset \neg p)$ *since* p, r
$O_3 =$ *why* r
$P_4 =$ *retract* q
or alternatively
$P_3 =$ *why* r
$O_3 = r$ *since* r
$P_4 =$ *retract* q

There are other examples.

Bibliography

Amgoud, L., and Besnard, P. (2009), "Bridging the gap between abstract argumentation systems and logic," in *Proceedings of the 3rd International Conference on Scalable Uncertainty Management (SUM'09)*, eds. L. Godo and A. Pugliese, no. 5785 in Springer Lecture Notes in AI, Berlin: Springer Verlag, pp. 12–27.

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

Amgoud, L., Bodenstaff, L., Caminada, M., McBurney, P., Parsons, S., Prakken, H., van Veenen, J., and Vreeswijk, G. (2006), "Final review and report on formal argumentation system," Deliverable D2.6, ASPIC IST-FP6-002307.

Amgoud, L., and Cayrol, C. (2002), "A model of reasoning based on the production of acceptable arguments," *Annals of Mathematics and Artificial Intelligence*, 34, 197–215.

Antoniou, G. (1999), "A tutorial on default logics," *ACM Computing Surveys*, 31, 337–359.

Baker, A., and Ginsberg, M. (1989), "A theorem prover for prioritized circumscription," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pp. 463–467.

Baroni, P., Dunne, P., and Giacomin, M. (2011), "On the resolution-based family of abstract argumentation semantics and its grounded instance," *Artificial Intelligence*, 175, 791–813.

Baumann, R. (2012), "What does it take to enforce an argument? Minimal change in abstract argumentation," in *Proceedings of the 20th European Conference on Artificial Intelligence*, pp. 127–132.

Baumann, R., and Brewka, G. (2010), "Expanding argumentation frameworks: Enforcing and monotonicity results," in *Computational Models of Argument. Proceedings of COMMA 2010* eds. P. Baroni, F. Cerutti, M. Giacomin and G. Simari, Amsterdam etc: IOS Press, pp. 75–86.

Besnard, P., and Hunter, A. (2009), "Argumentation based on classical logic," in *Argumentation in Artificial Intelligence* eds. I. Rahwan and G. Simari, Berlin: Springer, pp. 133–152.

Bisquert, P., Cayrol, C., Dupin de Saint-Cyr, F., and Lagasquie-Schiex, M.C. (2013), "Goal-driven changes in argumentation: a theoretical framework and a tool," in *Proceedings of the 25th International Conference on Tools with Artificial Intelligence (ICTAI 2013)*, pp. 610–617.

Bondarenko, A., Dung, P., Kowalski, R., and Toni, F. (1997), "An abstract, argumentation-theoretic approach to default reasoning," *Artificial Intelligence*, 93, 63–101.

Brewka, G. (1989), "Preferred Subtheories: An Extended Logical Framework for Default Reasoning," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pp. 1043–1048.

Caminada, M., *For the sake of the Argument. Explorations into argument-based reasoning*, Doctoral dissertation Free University Amsterdam (2004).

Caminada, M. (2006), "On the issue of reinstatement in argumentation," in *Logics in Artificial Intelligence. Proceedings of JELIA 2006*, eds. M. Fischer, W. van der Hoek, B. Konev and A. Lisitsa, no. 4160 in Springer Lecture Notes in AI, Berlin: Springer Verlag, pp. 111–123.

Caminada, M., and Amgoud, L. (2007), "On the evaluation of argumentation formalisms," *Artificial Intelligence*, 171, 286–310.

Carlson, L., *Dialogue Games: an Approach to Discourse Analysis*, Dordrecht: Reidel Publishing Company (1983).

Cayrol, C., Dupin de Saint-Cyr, F., and Lagasquie-Schiex, M.C. (2010), "Change in abstract argumentation frameworks: adding an argument," *Journal of Artificial Intelligence Research*, 38, 49–84.

Dung, P. (1995), "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and *n*–person games," *Artificial Intelligence*, 77, 321–357.

Dung, P., Kowalski, R., and Toni, F. (2009), "Assumption-based argumentation," in *Argumentation in Artificial Intelligence* eds. I. Rahwan and G. Simari, Berlin: Springer, pp. 199–218.

Grooters, D., and Prakken, H. (2016), "Two aspects of relevance in structured argumentation: minimality and paraconsistency," *Journal of Artificial Intelligence Research*, 56, 197–245.

Hamblin, C., *Fallacies*, London: Methuen (1970).

Hamblin, C. (1971), "Mathematical models of dialogue," *Theoria*, 37, 130–155.

Kraus, S., Lehmann, D., and Magidor, M. (1990), "Nonmonotonic reasoning, preferential models and cumulative logics," *Artificial Intelligence*, 44, 167–207.

Lifschitz, V. (1994), "Circumscription," in *Handbook of Logic in Artificial Intelligence and Logic Programming* eds. D. Gabbay, C. Hogger and J. Robinson, Oxford: Clarendon Press, pp. 297–352.

Loui, R. (1987), "Defeat among arguments: a system of defeasible inference," *Computational Intelligence*, 2, 100–106.

McCarthy, J. (1980), "Circumscription - a form of non-monotonic reasoning," *Artificial Intelligence*, 13, 27–39.

Modgil, S. (2006), "Hierarchical Argumentation," in *Logics in Artificial Intelligence. Proceedings of JELIA 2006*, eds. M. Fischer, W. van der Hoek, B. Konev and A. Lisitsa, no. 4160 in Springer Lecture Notes in AI, Berlin: Springer Verlag, pp. 319–332.

Modgil, S., and Prakken, H. (2012), "Resolutions in structured argumentation," in *Computational Models of Argument. Proceedings of COMMA 2012* eds. B. Verheij, S. Woltran and S. Szeider, Amsterdam etc: IOS Press, pp. 310–321.

Modgil, S., and Prakken, H. (2013), "A general account of argumentation with preferences," *Artificial Intelligence*, 195, 361–397.

Modgil, S., and Prakken, H. (2014), "The ASPIC+ framework for structured argumentation: a tutorial," *Argument and Computation*, 5, 31–62.

Parsons, S., Wooldridge, M., and Amgoud, L. (2003), "Properties and complexity of some formal inter-agent dialogues," *Journal of Logic and Computation*, 13, 347-376.

Pearl, J. (1992), "Epsilon-semantics," in *Encyclopedia of Artificial Intelligence* ed. S. Shapiro, New York: John Wiley & Sons, pp. 468–475.

Pollock, J., *Knowledge and Justification*, Princeton: Princeton University Press (1974).

Pollock, J. (1987), "Defeasible reasoning," *Cognitive Science*, 11, 481–518.

Pollock, J. (1994), "Justification and Defeat," *Artificial Intelligence*, 67, 377–408.

Pollock, J., *Cognitive Carpentry. A Blueprint for How to Build a Person*, Cambridge, MA: MIT Press (1995).

Pollock, J. (2009), "A recursive semantics for defeasible reasoning," in *Argumentation in Artificial Intelligence* eds. I. Rahwan and G. Simari, Berlin: Springer, pp. 173–197.

Prakken, H. (2005), "Coherence and flexibility in dialogue games for argumentation," *Journal of Logic and Computation*, 15, 1009–1040.

Prakken, H. (2006), "Formal systems for persuasion dialogue," *The Knowledge Engineering Review*, 21, 163–188.

Prakken, H. (2010), "An abstract framework for argumentation with structured arguments," *Argument and Computation*, 1, 93–124.

Prakken, H. (2012), "Some reflections on two current trends in formal argumentation," in *Logic Programs, Norms and Action. Essays in Honour of Marek J. Sergot on the Occasion of his 60th Birthday* Berlin/Heidelberg: Springer, pp. 249–272.

Prakken, H., and Sartor, G. (1997), "Argument-based extended logic programming with defeasible priorities," *Journal of Applied Non-classical Logics*, 7, 25–75.

Prakken, H., and Vreeswijk, G. (2002), "Logics for defeasible argumentation," in *Handbook of Philosophical Logic* (Vol. 4, Second ed.), eds. D. Gabbay and F. Günthner, Dordrecht/Boston/London: Kluwer Academic Publishers, pp. 219–318.

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

Vreeswijk, G. (1993), "Defeasible dialectics: a controversy-oriented approach towards defeasible argumentation," *Journal of Logic and Computation*, 3, 317–334.

Vreeswijk, G., *Studies in Defeasible Argumentation*, Doctoral dissertation Free University Amsterdam (1993).

Vreeswijk, G. (1997), "Abstract argumentation systems," *Artificial Intelligence*, 90, 225–279.

Vreeswijk, G., and Prakken, H. (2000), "Credulous and sceptical argument games for preferred semantics," in *Proceedings of the 7th European Workshop on Logics in Artificial Intelligence (JELIA'2000)*, no. 1919 in Springer Lecture Notes in AI, Berlin: Springer Verlag, pp. 239–253.

Walton, D., *Logical dialogue-games and fallacies*, Lanham, MD: University Press of America, Inc. (1984).

Walton, D., *Argumentation Schemes for Presumptive Reasoning*, Mahwah, NJ: Lawrence Erlbaum Associates (1996).

Walton, D., *Fundamentals of Critical Argumentation*, Cambridge: Cambridge University Press (2006).

Walton, D., and Krabbe, E., *Commitment in Dialogue. Basic Concepts of Interpersonal Reasoning*, Albany, NY: State University of New York Press (1995).

Wu, Y., *Between Argument and Conclusion. Argument-based Approaches to Discussion, Inference and Uncertainty*, Doctoral Dissertation Faculty of Sciences, Technology and Communication, University of Luxemburg (2012).