STUDIES IN LOGIC, GRAMMAR AND RHETORIC 23 (36) 2011

Floris Bex and Chris Reed
University of Dundee

SCHEMES OF INFERENCE, CONFLICT, AND PREFERENCE IN A COMPUTATIONAL MODEL OF ARGUMENT

Abstract: Argumentation demands that various non-deductive patterns of reasoning are accounted for from a strong theoretical foundation. The theory of argumentation schemes has provided such a theoretical foundation, and has led to a significant programme of research not only in epistemological and metaphysical philosophy but also in knowledge representation and multi-agent systems in artificial intelligence. More recently, work on computational models of argument has posited that not only inference, but also conflict, might be governed by more sophisticated relationships that just propositional negation. And finally, work on developing a standard computational ontology for handling argument has suggested that preference too demands such schematization. This paper shows how schematic templates can be designed to capture these stereotypical patterns of inferring, conflicting and preferring, and furthermore, demonstrates the strong representational and constitutive similarities between these apparently very different phenomena.

Keywords: argumentation schemes, Argument Interchange Format, inference, conflict, preference

1. Introduction

The theory of argument is a rich, interdisciplinary area with insights from diverse disciplines such as philosophy, law, psychology, communication studies and artificial intelligence. This paper explores the ways in which parts of arguments can be connected together.

Recent research in philosophy has shown that the broad range of ways in which inference is performed in natural texts can be understood by taxonomizing and classifying 'argumentation schemes', which capture stereotypical patterns of reasoning (Walton 1996; Walton et al. 2008). These argumentation schemes have been demonstrated to be not only powerful tools for scholarly investigation of argument, but also of practical use both in pedagogy and in computational settings (Reed and Walton, 2005). In addition to inference, however, argument makes fundamental use of two further types

ISBN 978–83–7431–***–* ISSN 0860-150X

of relation: conflict and preference. Conflict acts as a driver for argumentative discourse, and for many authors is a defining feature of such linguistic behaviour. Preference, in turn, is the key to resolving conflict, particularly where (as is very common) the conflict is rooted not just in propositional disagreement, but in mismatches in values.

This paper argues for an approach that tackles inference, conflict and preference as genera of a more abstract class of schematic relationships. This allows the three types of relationship to be treated in more or less the same way, meaning that the logical and semantic machinery required for handling them is greatly simplified. The context of the work is a method for representing argument structures which aims simultaneously to provide a language that is rich enough to talk about the enormous variety of naturally occurring argument, whilst at the same time enforcing a level of specificity and clarity that allows for computational interpretation. This context is the Argument Interchange Format, AIF, which serves as an interlingua between various software tools and systems in the burgeoning community in computational models of argument (Chesñevar et al., 2006).

The rest of this paper is organised as follows. In section 2 we provide a brief and very general introduction to the most important concepts in argumentation theory. Section 3 introduces the language of the AIF; 3.1 discusses the basic concepts and 3.2 concentrates on the various schematic relations. Section 4–6 discuss inference, conflict and preference schemes, respectively. These sections start with a short introduction to the representation of inference, conflict and preference in models of computational argument. After this, the modelling of these concepts in the language of the AIF is presented. Section 7 concludes the paper.

2. Argumentation

In an argument, a defeasible inference leads from premises to a conclusion; associated with a defeasible inference is a generalization, usually in a conditional form, which justifies or *warrants* the inference link between premises and conclusion. Generalizations are generalized statements about how we think the world around us works; they can express generally accepted patterns (e.g. ("If a witness testifies that 'P' then P is the case") or they can be more case-specific (e.g. "Chris is usually at work before 8 o' clock"). Very often, generalizations are left implicit in natural argument, but explicitly expressing the generalization can help in determining the relevance and force of the inference. Take, as a simple example the argument for the conclusion that Harry was in Dundee based on Bob's testimony, visualised as a diagram in the style of Toulmin (2003):

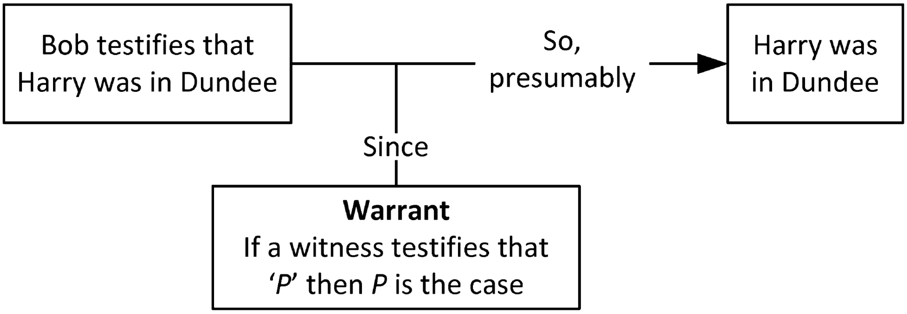

Figure 1: a Toulmin-style argument for the claim that Harry was in Dundee

Generalizations that occur often in natural argument have been studied in the form of *argumentation schemes* (Walton et al. 2008), stereotypical patterns of reasoning.[1] As an example, take the scheme for "argument from appeal to witness testimony", which is similar to the above generalization (adapted from Bex et al. 2003):

> Witness *W* asserts that *P* is true (false).
> *Therefore*, *P* may plausibly be taken to be true (false).

Associated with each argumentation scheme are critical questions that point to standard sources of doubt. Standard sources of doubt with regards to witness testimony are, for example, the witness' bias, whether he is lying or whether he correctly remembers what he observed; critical questions for the argumentation scheme are hence, for example, 'Is the witness biased?' or 'Is there a chance that the witness misremembers?'.

Most everyday arguments are defeasible, in that new information can cast doubt on information previously taken to be true. For example, 'witness Peter testifies that Harry was in Amsterdam' is a reason for the fact that Harry was in Amsterdam. This provides a counterargument to the original conclusion that Harry was in Dundee. In addition to attacking conclusions (called rebuttal in the literature, see Pollock 1994, Prakken 2010), we may also attack the defeasible inference (this type of attack is often called *under-cutting*). Recall that the generalizations or schemes that justify inferences

---

[1] As Prakken (2010) has shown, argumentation schemes are often (but not always) generalized rules of inference.

express a stereotypical pattern of everyday reasoning: normally we expect that people bear witness only to events they actually observed. However, it is not unthinkable that in any particular case the witness misremembers or lies (cf. the critical questions for the argument from witness testimony). In such a case we are dealing with an exception to the general rule. Such an exception does not deny the premise or conclusion of the argument but attacks the inference from premise to conclusion: if, in the example, the witness is lying, this does not mean that Harry was *not* in Dundee; it just shows that this particular witness testimony is not a good reason for believing this conclusion.

Undercutting and rebutting are just ways to express conflict in argumentation. Conflict is just as important as inference in argumentation: the dialectical process is essentially a process of argument and counterargument. However, while some of the mathematical properties of conflict have been extensively studied,[2] (context-)specific types of conflict have not received much attention in (computational) argumentation theory.[3]

Inference and conflict allow us to build arguments and provide counterarguments. In many contexts, a choice then needs to be made as to which of the arguments one decides to believe or, in other words, which of the arguments is *preferred*. This preference is naturally tied to the applicable "rules" of the discussion (e.g. a judge or jury cannot decide for an argument on inadmissible evidence, even if she prefers this argument). In general, however, this preference is intimately tied to the beliefs of the person doing the evaluation. For example, we can only accept that Harry was in Dundee if we believe that Bob (who stated Harry was in Dundee) is a more trustworthy witness than Peter (who stated that Harry was in Amsterdam). Formal models of argumentation have long enjoyed rich, mature models of preference and priority. Bench-Capon (2003), for example, has shown how one's values might influence the choice of beliefs and Modgil (2009) has extended Dung's (1995) abstract argumentation frameworks with reasoning about preferences. However, as with conflict, more context-specific patterns of preference (outside the value orderings of Bench-Capon) have not been widely examined in (computational) argumentation theory.

---

[2] See the large body of work on argumentation theoretic semantics in the style of (Dung 1995), e.g. (Caminada 2006, Dunne 2009).

[3] However, some argumentation schemes, such as the scheme for ad hominem arguments, seem to have more to do with conflict rather than inference.

3. The Argument Interchange Format

Argumentation is a large and diverse field stretching from analytical philosophy to communication theory and social psychology. The computational investigation of the space has multiplied that spectrum by a diversity of its own in semantics, logics and inferential systems. One of the problems associated with the diversity and productivity of the field, however, is fragmentation. With many researchers from various backgrounds focusing on different aspects of argumentation, it is increasingly difficult to reintegrate results into a coherent whole; for the plethora of methods, processes and tools for argumentation, there are just as many individual languages for argumentation, ranging from logical to visual to natural language. This fragmentation makes it difficult to present new ideas which can be adapted across the board and difficult for new research to build upon old. To tackle this problem, the computational argument community has initiated an effort aimed at building a common ontology for argument: the Argument Interchange Format (AIF).

The AIF is a communal project which aims to consolidate some of the defining work on (computational) argumentation (Chesñevar et al. 2006). The AIF project aims to present a common vision and consensus on the concepts and technologies in the field, thus promoting research and development of new argumentation tools and techniques. A main aspiration of the AIF is to facilitate data interchange among various tools and methods for argument analysis, manipulation and visualization. To this end, the AIF project aims to develop a commonly agreed-upon *core ontology* that specifies the basic concepts used to express argumentative information and relations. The purpose of this ontology is not to replace other languages for expressing argument but rather to serve as an abstract *interlingua* that acts as the centrepiece to multiple individual languages. These argument languages may be, for example, logical languages (e.g. ASPIC, see Prakken 2010), visual diagramming languages (e.g. Araucaria, see Reed and Rowe 2004) or natural languages (e.g. pragma-dialectics, see van Eemeren and Grootendorst 2004). The idea is that an interlingua drastically reduces the number of translation functions that are needed for the different argumentation languages to engage with each other; only translation functions to the AIF core ontology have to be defined (i.e., n instead of n^2 functions for n argumentation languages).

3.1. The AIF core ontology

In the AIF ontology, arguments and their mutual relations are described by conceiving of them as a an *argument graph*. The ontology falls into two natural halves: the Upper Ontology and the Forms Ontology. The Upper Ontology, introduced in (Chesñevar et al. 2006), describes the language of different types of nodes and edges with which argument graphs can be built. The Forms Ontology, introduced by (Rahwan et al. 2007), allows for the conceptual definition of the elements of the graphs, that is, it describes the argumentative concepts instantiated by the specific nodes in a graph. Figure 2 visually renders part of the ontological structure of the AIF; the explanation of the different elements is below Figure 2. Note that here, only a part of the ontology is shown; as we will show in this paper, for example, conflict schemes also have descriptions of the elements that are in conflict. For readability, however, only the elements connected to the defeasible and deductive inference schemes are shown.

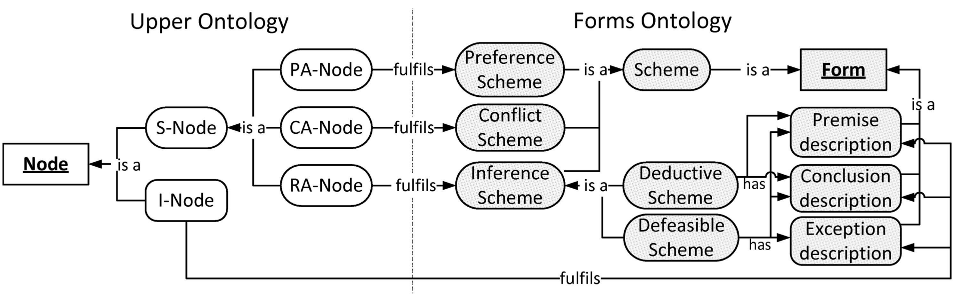

Figure 2: The AIF core ontology

The Upper Ontology places at its core a distinction between *information*, such as propositions and sentences, and *schemes*, general patterns of reasoning such as inference or conflict, which are used to relate pieces of information (I-nodes) to each other. Accordingly, there are two types of nodes for building argument graphs: information nodes, *I-nodes*, and scheme nodes, *S-nodes*. Individual nodes can have various attributes (e.g. "creator", "date"). In a graph, I-nodes can only be connected to other I-nodes via S-nodes, that is, there must be a specific scheme application that expresses the rationale behind the relation between I-nodes. In the basic AIF ontology, scheme nodes can be rule application nodes (*RA-nodes*), which denote specific inference relations, conflict application nodes (*CA-nodes*), which denote specific conflict relations, and preference application nodes (*PA-nodes*), which denote specific preference relations. Different S-nodes can be connected to each other; for example, we can express that two preference applications are in conflict with each other (e.g., $x > y$ and $y > x$) by connecting the two PA-nodes through a CA-node.

3.2. Scheme application in the AIF

The Upper Ontology defines the basic building blocks of argument-graphs (in a sense, it defines the "syntax" for our abstract language). In contrast, the Forms Ontology defines what these individual nodes mean in argumentative terms It defines the forms of the schemes that are used in reasoning, that is, the inference schemes, conflict schemes and preference schemes. Informally, inference schemes are criteria for inferring (deductively, inductively or presumptively), conflict schemes are criteria (declarative specifications) defining conflict (which may be logical or non-logical) and preference schemes express (possibly abstract) criteria of preference. These main scheme types can be further classified. For example, inference schemes can be deductive or defeasible. Defeasible inference schemes can be further subdivided into more specific argumentation schemes, such as Expert Opinion, Practical Reasoning and so on (see, for example, Walton et al. 2008).[4] Accordingly, the AIF ontology has a Schemes Ontology, which is a sub-ontology of the Forms Ontology. This Schemes Ontology contains specific inference schemes and may vary from very simple (containing only the basic deductive and defeasible schemes) to extensive (containing a large number of specific deductive and defeasible argumentation schemes).

As can be seen in Figure 2, the Forms Ontology and the Upper Ontology are intimately connected because a specific applications of schemes (denoted by RA-, CA- and PA-nodes) are instantiations of general (inference-, conflict- and preference-) schemes; in other words, the S-nodes *fulfil* the schemes expressed in the Forms Ontology. Like argument-graphs from the abstract language of the AIF, schemes can also be translated into a more concrete language; for example, Rahwan et al. (2010) define schemes as combinations of classes of statements in Description Logic, with object level arguments then being instances of those classes. In this paper, like in (Rahwan et al. 2007), we will represent the Forms Ontology and the schemes contained in it as graphs.[5]

RA-, CA- and PA-nodes capture the passage or the process of inferring, conflicting and preferring, respectively, whilst the inference schemes, conflict schemes and preference schemes embody the general principles expressing how it is that A is inferable to B, A is contrastable to B ('conflictable' is too

[4] It is important to note that the AIF ontology does not (and should not) legislate as to which schemes or forms are the correct ones; different schemes are each plausible according to particular theoretical assumptions.

[5] Note that these graphs simply express concepts (i.e. Forms) and the ontological relations between them; they are *not* AIF argument graphs, which exist at the object level.

cumbersome a term), and A is preferable to B, respectively. RA-nodes thus correspond mostly closely to what a traditional system of formal logic would regard as entailment, i.e. where φ is a premise for an RA to a conclusion ψ, the RA corresponds to $\varphi \vdash \psi$. In contrast, conditionals such as $\varphi \rightarrow \psi$ are available as I-nodes – in instances of (defeasible) modus ponens, for example. Of course it is possible to formulate – in natural language – a proposition corresponding to the fact that $\varphi \vdash \psi$, so in principle we can also represent such a proposition as an I-node. But this I-node can be handled as a special type of 'calculated property' (Reed 2010): a propositional result of running some (arbitrary) process over an AIF structure. This proposition could use the RA itself as a basis for establishing the entailment proposition, but as Bex et al. (2010) have argued, this exact connection between an AIF graph and the properties calculated on the basis of the graph cannot be captured in the core AIF ontology itself (and nor should they be, for otherwise the AIF would swell to some general purpose programming language).

This contrast between propositions expressing implicative relationships and propositions expressing entailment relationships is important because for inference, we have strong intuitions and mature theory to guide the way in which the AIF should handle them. For conflict and preference, we need to develop strong analogues to inferential components. If we say then that $\varphi \rightarrow \psi$ expresses that ψ is inferable from φ, we might similarly say that $\varphi \succ \psi$ expresses that φ is preferable to ψ, and $\varphi \, {-\!\times} \, \psi$ expresses that φ is contrastable with ψ. These could all be captured by I-nodes and could all serve as foundations for RA-, CA- and PA-nodes respectively. In contrast, $\varphi \vdash \psi$ captures that ψ is (in fact) inferred from φ, and so similarly we might say that $\varphi \mid\!\succ \psi$ corresponds to the fact that φ is (in fact) preferred to ψ, and that $\varphi \mid\!\times \psi$ corresponds to the fact that φ does (in fact) conflict with ψ. Again, these can all be captured by I-nodes, but their connection to RA-, CA- and PA-nodes is tenuous and is governed by the process which determines these calculated properties, and not by the AIF per se. These strong analogies between the three schematic classes are very useful in developing accounts of scheme usage through the AIF as a whole.

4. Schemes of Inference

One of the main issues of argumentation discussed in section 2 concerns the generalizations warranting the defeasible inferences. In a logic, conditional generalizations of the form 'if φ then ψ' (or'φ therefore ψ') can be modelled both as an object-level rule (φ *implies* ψ, formally represented as

$\varphi \rightarrow \psi$) or as a metalinguistic rule of inference (φ *entails* ψ, formally represented as $\varphi \vdash \psi$). The various argumentation logics (see Chesñevar et al. 2000, Prakken and Vreeswijk 2002 for overviews) have different stances on how these generalizations should be modelled. For example, Prakken (2010) and Pollock (1994) model them as rules of inference, whereas (Bondarenko et al. 1997) and Verheij (2003) model them as (defeasible) implications in the object language and have a (defeasible) modus ponens inference rule for reasoning with these implications. The most important difference between the two ways of modelling them is that conditionals in the object language can be reasoned about in a natural way; they can, for example, be denied by arguing that "$\neg(\varphi \rightarrow \psi)$" or they can serve as the conclusions of arguments. A problem for the argumentation logics that model generalizations as inferences is that it is often unclear how statements like $\varphi \vdash \psi$ can be rendered in the object language and what, *in the object language*, their relation to $\varphi \rightarrow \psi$ is. In the AIF ontology, the fact that in the AIF $\varphi \vdash \psi$ is represented by its own RA-node in the object language and $\varphi \rightarrow \psi$ is represented by its own I-node in the object language disambiguates this relationship between the two.

Now, as was argued above conditional generalizations can be modelled either in the object language (as an I-node) or in the metalanguage, as a Scheme in the Forms Ontology. Figure 3a models the conditional expressing the generalization as a premise (I-node) and connects this premise together with another premise (I-node) representing the antecedent of the conditional to the conclusion by way of a (defeasible) modus ponens inference (RA-node).

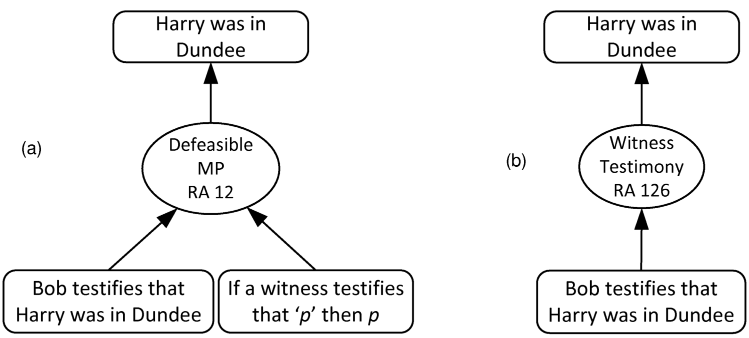

Figure 3: Two ways of modelling defeasible inference

The inference rule that is applied is explicitly shown in the AIF structure. In the case of the argument in Figure 3a, the generalization justifying the inference ("If a witness testifies that 'P' then P") is made explicit as an

I-node and can be questioned. However, no further information about the generalization is provided; if an arguer or an analyst wishes to critique the inference step (e.g. by undercutting it), it remains for them to introduce sufficient contextual knowledge to form an attack. One of the advantages of the scheme-based approach advocated by, among others, Walton et al. (2008) is that it provides a theoretically principled way of structuring this contextual knowledge. So the Argument Scheme from Witness Testimony provides not just a characterisation of the minor premise and conclusion, but also a raft of implicit premises (presumptions) which may be taken to hold, and exceptions, which may be taken not to hold. These presumptions and exceptions are part of the Scheme Ontology. A scheme-based analysis (Figure 3b) shows that the premise and conclusion are connected by this specific type of Witness Scheme inference. The general form of this scheme gives the implicit premises and exceptions are part of the scheme and which can be used to critique the scheme. Figure 4 shows both the abstract Defeasible Modus Ponens (a) and the specific Witness Testimony Scheme (b). Notice that the Witness Testimony scheme shows the (implicit) presumptions and exceptions; exactly how these can be used to attack an argument that uses the scheme will be discussed below in section 5.

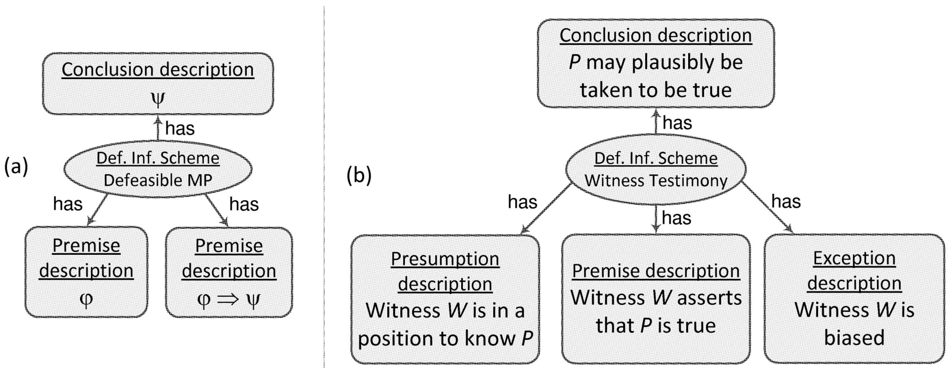

Figure 4: The Defeasible Modus Ponens and Witness Testimony schemes as represented in the AIF

So modelling generalizations as conditional premises as in Figure 4a allows for a lot of flexibility, whereas modelling them as schemes as in Figure 4b provides a firm grounding to the rules of inference that are being used in our reasoning. This can be seen in the case of Toulmin's characterisation of backing, a reason for why we should believe the warranting generalization. In at least some of Toulmin's examples, backing serves to justify a general rule, rather than its specific application. This is possible in the case of the argument in Figure 3a: reasons can be given for the conditional premise that expresses the generalization. In the case of the argument in Figure 3b, a backing can only be given if it is explicitly encoded in the Forms Ontology (i.e. if the scheme for Witness Testimony has a Backing description). It is not possible to give a backing in an object-level argument, as this would require Scheme Forms from the Scheme Ontology to be able to stand as the conclusions of arguments.

It is important to note that AIF ontology does not (and should not) legislate as to which analysis in Figure 3 is correct. They are each plausible according to particular theoretical assumptions. Similarly, the AIF ontology does not (and should not) legislate as to which schemes or forms are the correct ones; different schemes are each plausible according to particular theoretical assumptions. Argument analysis needs, like many techniques applicable to naturally occurring language, to be flexible, and to admit of alternative views. AIF's job is to make such alternative analyses clear and unambiguous in a common language.

5. Schemes of Conflict

Conflict is a central notion in dialectical argumentation and it can take many forms. For example, two claims may be in conflict because they express opposing points of view or because they were uttered by people from different political parties. In logical models of argument, conflict is often equated with logical conflict, i.e., the contradiction between φ and $\neg\varphi$. Some frameworks for formal argumentation (e.g. Bondarenko et al. 1997, Prakken 2010) generalize this to a contrariness relation, where φ is in conflict with its contrary $\bar{\varphi}$. Thus, other non-logical conflict relations can be expressed.

An important concept in logical models of argument, which is closely related to conflict, is that of *attack*. Attack expresses that one argument is somehow a counterargument to another.[6] However, conflict is not the same as attack. First, the fact that two propositions are in conflict does not mean they attack each other, as this depends on one's definition of attack. For example, in the ASPIC framework (Prakken 2010) a proposition φ only attacks another proposition $\neg\varphi$ if $\neg\varphi$ is not a necessary premise. If this is

[6] Not to be confused with "defeat". Attack and defeat are different concepts: attacking your enemy does not guarantee their defeat, only a successful attack defeats. So attack expresses that one argument is a counterargument to another, whilst defeat says that an argument is a counterargument and is preferred (Garcia and Simari 2004).

the case, φ is in conflict with $\neg\varphi$ but it does not attack it. Here, attack is *based* on conflict, it is a calculated property. Second, attack is often defined over arguments, where conflict is usually only defined over propositions and, in some cases, inference applications (Figure 7).

In the AIF ontology, conflict is expressed using *conflict schemes* in the Forms Ontology and applications of these schemes in the object layer, *conflict application* or *CA-nodes*. Conflict schemes are similar to (but certainly not analogous to) inference schemes, in that they are patterns of reasoning which are often used in argumentation. Like inference schemes, conflict schemes may denote abstract, logical patterns (e.g. logical conflict) as well as more concrete patterns of conflict dependent on, for example, legal or linguistic conventions (e.g. a bachelor is not married, a man is not a woman). Like inference schemes, conflict schemes can be strict (no exceptions to the scheme; e.g., φ and $\neg\varphi$ are always in conflict) or defeasible (there are exceptions to the scheme; e.g. a man is not a woman unless (s)he is androgynous).

Like inference, conflict is often expressed as a generalization; for instance, "people cannot be in two places at the same time" or "it is impossible for both the Tories and Labour to both be in government". Where generalizations that warrant inference are often rephrased as conditionals of the form "if φ then ψ", generalizations that express conflict can be rephrased as "φ conflicts with ψ". These conflict generalizations can be represented as information (I-nodes) in the object layer, or in the layer of the Schemes Ontology, as conflict schemes. Take, for example, the conflict between the British Labour Party and the British Conservative Party ("the Tories") being in government. Generally, the two parties are not in the same government (the last time was during the Second World War). Now, we can make the generalization (Figure 5a), or we can model the conflict generalization a separate conflict scheme (Figure 5b and Figure 6b).

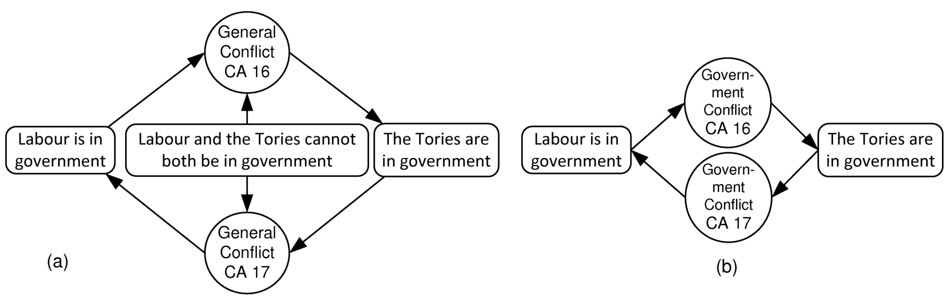

Figure 5: Two ways of modelling conflict generalizations

Figure 5 shows an important difference between conflict and inference, namely that often (but not always), conflict is a symmetrical relation, whilst inference is certainly not. That is, if φ is in conflict with ψ, then ψ is also in conflict with φ. For inference, this is not the case. One of the reasons for this is that with inference, we can *gain* new information (e.g. we have information that a witness testified that Harry was in Dundee, so we can infer the new information that Harry was in Dundee). Conflict schemes have no such generative function, as they only allow us to represent conflict between existing propositions.

In Figure 5, two conflict schemes are used, a general and a specific one. These two schemes are rendered in Figure 6. The general conflict scheme (Figure 6a) takes a generalization from an I-node and uses this generalization to warrant the application of a conflict. In this sense, it can be likened to the inference scheme for (Defeasible) Modus Ponens (Figure 4a), which warrants the *inference application* with a generalization from an I-node.

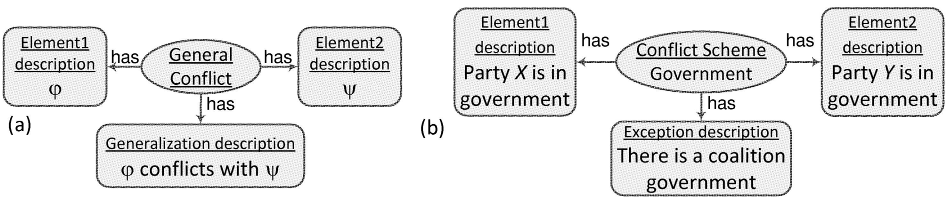

Figure 6: Conflict schemes in the AIF Forms Ontology

An advantage of modelling conflict generalizations as I-nodes is that they can be reasoned about. For example, we can give reasons for why, in general, Labour and Tories cannot be in the same government, having the I-node that contains this generalization in Figure 5 as the conclusion of an RA-node. When conflict generalizations are modelled as schemes in the Forms Ontology, it is not possible to provide them with a "backing" in this way. However, representing a conflict generalization as a scheme allows us to specify implicit presumptions and exceptions to the scheme. For instance, an exception to the generalization that Labour and Conservatives are not in the same government is that there is a coalition government, as was the case during the Second World War (the exception basically says that the elements 1 and 2 are in conflict *unless* there is a coalition government of party X and Y). Thus, the implicit presumptions and exceptions to conflict relations can be incorporated in a principled way.

Conflict does not just exist between I-nodes. There are cases in which, for example, some information is in conflict with an inference or a preference, or two inferences or preferences are in conflict. Take the example in

Figure 7. Here, the information that Bob is biased conflicts with the application of the Witness Testimony inference scheme. This type of conflict, called undercutting by Pollock (1994), is quite common in argumentation. It allows us to attack the way in which some information has been derived rather than the information itself (that is, we attack $\varphi \vdash \psi$). In the example, knowing that Bob is biased is not a reason for the opposite conclusion, that Harry was *not* in Dundee, but rather it is a reason to believe that we might not be justified in inferring Harry's whereabouts from Bob's testimony. Figure 7b shows the conflict scheme used in the argument. Note how this conflict scheme connects an inference scheme with one of its exceptions.

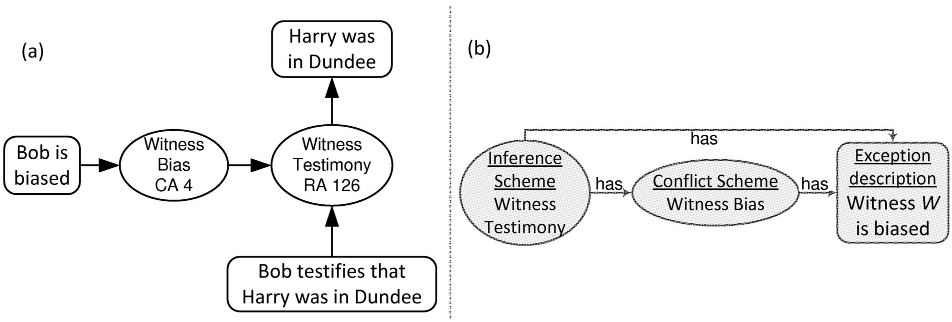

Figure 7: Conflict between an I-node and an RA-node

6. Preference Schemes in Argumentation

In addition to inference and conflict, the we treat preference as a basic concept of argumentation. Inference and conflict allow us to build arguments and provide counterarguments. In many contexts, a choice then needs to be made as to which of the arguments one decides to believe. Based on the arguments for the prosecution and the defence, does the jury rule the suspect to be guilty or innocent? After a long election campaign, who do we decide to vote for? After comparing the pros and cons, which car (if any) do we buy? The thought that one argument (or set of arguments) is considered better or stronger than another can be expressed using preferences. For example, the jury can argue that the witnesses for the prosecution were more convincing than those for the defence. Which argument we believe may depend on personal preferences; for instance, someone who prefers equality to enterprise and red to blue will generally vote social democrats and buy red cars.

Formal models of argumentation have long enjoyed rich, mature models of preference and priority. For example, (Amgoud and Cayrol 2002, Garcia and Simari 2004) define in, for example, systems of preference-based argumentation, where preferences are used to determine whether an argument that attacks another argument actually defeats the attacked argument. Bench-Capon (2003) further extends this by basing the preferences between arguments on value orderings. Modgil (2009) has proposed Extended Argumentation Frameworks in the style of Dung (1995), where preferences are modelled as attacks on attacks: if an argument A is preferred to another argument B, any attack from B on A is itself attacked. Recently, Prakken (2010) has incorporated preferences in his framework for structured argumentation. Here, the preferences are not between arguments but rather between premises or inference rules. If desired, preferences between arguments can be calculated on the basis of these preferences (Modgil and Prakken 2010).

In line with the now-familiar pattern, the AIF ontology expresses preferences by using *preference schemes* and applications of these schemes in the object layer, *preference application* or *PA-nodes*. Like conflict schemes, preference schemes are similar to inference schemes, again with some important differences, which will be highlighted below. Like inference schemes, preference schemes are patterns of reasoning which are often used in argumentation, which may be abstract, logical patterns as well as more concrete and context-dependent patterns. Preference schemes can also be strict (no exceptions to the scheme) or defeasible (there are exceptions to the scheme).

In argumentation (as in most everyday language use), the preferences themselves can be expressed as generalizations of the form "φ is preferred to ψ". As with inference and conflict, these generalizations (which can be said to warrant a particular preference) can be explicitly rendered in the object layer, that is, as I-nodes, or they can be modelled as a concrete preference scheme in the Schemes Ontology. So, for example, say that we have a generalization that, in general, government policies that promote equality are preferred over policies that promote enterprise. Figure 8a shows this generalization as an I-node that warrants the application of a general preference scheme and Figure 8b shows this generalization as a specific preference scheme. The preference schemes used in Figure 8 are shown in Figure 9.

Note the similarities with conflict and inference: while modelling the generalization as in Figure 8a allows us to further reason about this generalization, incorporating it as a scheme allows us to provide possible exceptions to this generalization. One example of reasoning about preference generalizations is to base them on one's ideals, one's values (Bench-Capon 2003).

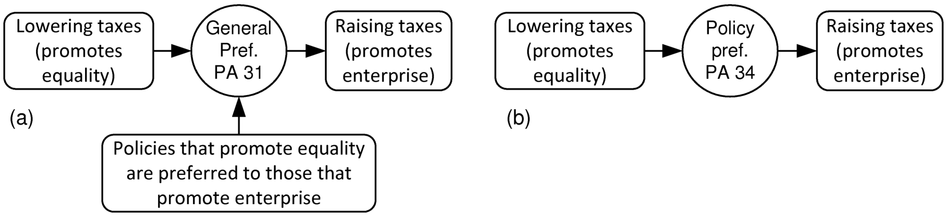

Figure 8: Two ways of modelling preference generalizations

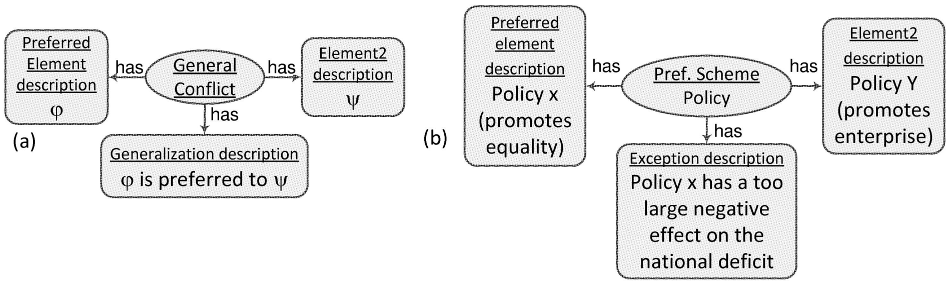

Figure 9: Preference schemes in the AIF Forms Ontology

Figure 10 shows how this can be done. The scheme that corresponds to RA74 is not rendered, but will be something along the lines of "if one prefers value A to value B, one should amend one's policies accordingly".

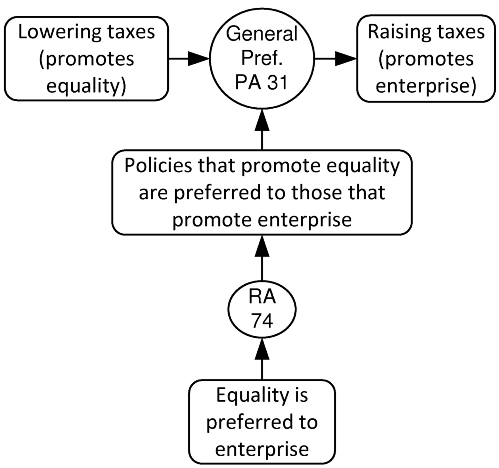

Figure 10: Two ways of modelling preference generalizations

As was already discussed in section 4, it is of course possible to provide such a "backing" for the policy preference scheme in the Forms Ontology. As for inference generalizations, rendering them as I-nodes provides flexibility, as the generalization can easily be denied or argued for. Rendering a generalization as a scheme, however, is that it structures contextual knowledge in a principled way. Whilst argumentation schemes for inference are a subject of much study, conflict and preference schemes have not yet been fully developed. Hence, the examples in this paper of such schemes (Figure 6b and Figure 9b) might seem somewhat far-fetched. More intuitive examples the sorts of contextual knowledge preference schemes in the Forms Ontology can express are perhaps the irreflexivity and antisymmetry properties of a particular preference relation. Take, for example, the preference relation $\succ$ as described by (Prakken 2010). A scheme for this relation can be incorporated in the Forms Ontology (Figure 11).

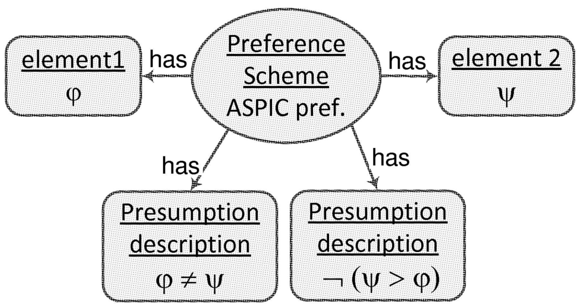

Figure 11: The ASPIC preference relation as a scheme in the Forms Ontology

Here, the properties of irreflexivity and antisymmetry have been incorporated into the scheme as implicit presumptions.

7. Concluding remarks

In this paper, we have shown how the apparently very different relationships of inference, conflict and preference can be captured analogously in a common language. The approach provides an ontologically parsimonious way of handling a diverse and sophisticated range of argumentation components. Schematising all of these relationships offers particular advantages in terms of explicit characterisation of the constitution of different forms of inference, conflict and preference; spelling out missing or implicit parts (such as assumptions and presumptions), and capturing stereotypical ways of evaluating and critiquing.

We have also shown for the first time how scheme instances can interact with propositional statements that capture expressions of inference, preference and conflict, by virtue of the distinction between, on the one hand, the inferring/preferring/conflicting relation captured by RA/PA/CA-nodes and on the other hand, the inferability/preferability/contrastability captured by I-nodes. Whereas in the current logics for argumentation the distinction between $\varphi \rightarrow \psi$ and $\varphi \vdash \psi$ is fairly well developed, these distinctions are often not explicitly made for preference (i.e. between $\varphi \succ \psi$ and $\varphi \mid\!\succ \psi$) or for conflict (i.e. between $\varphi - \!\times \psi$ and $\varphi \mid\!\times \psi$).[7] As an increasing number of research groups and systems start to take advantage of what the AIF has to offer, and thereby, what other teams have already achieved, it becomes vital that a thorough understanding of schematic argument relations and their inter-connections is established, and it is this that the current paper has laid out.

References

Amgoud, L., and Cayrol, C. (2002). A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence* 34:1, 97–215.

Bex, F. J., Prakken, H. and Reed, C. A. (2010). A formal analysis of the AIF in terms of the ASPIC framework. In P. Baroni, F. Cerutti, M. Giacomin & G. R. Simari (eds.): *Computational Models of Argument. Proceedings of COMMA 2010.* 99–110. Amsterdam: IOS Press.

Bex, F. J., Prakken, H., Reed, C., and Walton, D. N. (2003). Towards a formal account of reasoning about evidence: argumentation schemes and generalisations. *Artificial Intelligence and Law* 11, 125-165.

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

Bondarenko, A., Dung, P. M., Kowalski, R. A., and Toni, F. (1997). An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence* 93:1–2, 63–101.

[7] One could argue that in Bench-Capon (2003), the value ordering expresses the preferability relation which is used to found the actual preference between propositions (in this case propositions expressing particular government policies). However, as we showed in Figure 10, the value ordering is actually a reason for the preferability between the policies, on which ultimately the actual preference between the policies is founded.

Caminada, M. W. A. (2006). Semi-Stable Semantics. *Computational Models of Argument. Proceedings of COMMA 2006*, 121–130. Amsterdam: IOS Press.

Chesñevar, C. I., Maguitman, A. G. and Loui, R. P. (2000). Logical models of argument. *ACM Computing Surveys (CSUR)* 32:4, 337–383.

Chesñevar, C., McGinnis, J., Modgil, S., Rahwan, I., Reed, C., Simari, G., South, M., Vreeswijk, G. and Willmott, S. (2006). Towards an Argument Interchange Format, *Knowledge Engineering Review* 21 (4), 293–316.

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

Dunne, P. E. (2009). The computational complexity of ideal semantics. *Artificial Intelligence*, 173:1559–1591.

Eemeren, F. H. van, and Grootendorst, R. (2004). *A Systematic Theory of Argumentation – The pragma-dialectical approach.* Cambridge: Cambridge University Press.

Freeman, J. B. (1991). *Dialectics and the Macrostructure of Arguments: A Theory of Argument Structure.* Berlin: Foris Publications.

García, A. J., and Simari, G. R. (2004). Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Programming* 4:1, 95–138.

Modgil, S. (2009). Reasoning About Preferences in Argumentation Frameworks. *Artificial Intelligence* 173:9–10, 901–1040.

Modgil S. and Prakken, H. (2010). Reasoning about preferences in structured extended argumentation frameworks. In P. Baroni, F. Cerutti, M. Giacomin & G. R. Simari (eds.): *Computational Models of Argument. Proceedings of COMMA 2010.* 347–358. Amsterdam: IOS Press.

Pollock, J. L. (1994). Justification and defeat. *Artificial Intelligence* 67 (2), 377–408.

Prakken, H. (2010). An abstract framework for argumentation with structured arguments. *Argument and Computation* 1, 93–124.

Prakken, H. (2010). On the nature of argument schemes. In C. A. Reed and C. Tindale (eds.) *Dialectics, Dialogue and Argumentation. An Examination of Douglas Walton's Theories of Reasoning and Argument*, 167–185. London: College Publications.

Prakken, H., and Vreeswijk, G. (2002). Logics for defeasible argumentation. In Goebel, R. and Guenthner, F. (eds.), *Handbook of Philosophical Logic*, 219–318, Dordrecht: Kluwer Academic Publishers.

Rahwan, I., Zablith, F. and Reed, C. (2007). Laying the Foundations for a World Wide Argument Web, *Artificial Intelligence* 171, 897–921.

Reed, C. (2010). "Walton's theory of argument and its impact on computational models" in Reed, C. & Tindale, C. W. (eds.) *Dialectics, Dialogue and Argumentation. An Examination of Douglas Walton's Theories of Reasoning and Argument*, 73–84. London: College Publications.

Reed, C. and Rowe, G. (2004). Araucaria: Software for Argument Analysis, Diagramming and Representation, *International Journal of AI Tools* 13 (4), 961–980.

Reed, C. & Walton, D. (2005). "Towards a Formal and Implemented Model of Argumentation Schemes in Agent Communication", Autonomous Agents and Multi-Agent Systems, 11 (2), pp. 173–188.

Toulmin, S. E. (2003). *The Uses of Argument*, Updated edition (originally published in 1958), Cambridge: Cambridge University Press.

Verheij, B. (2003). DefLog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation* 13:3, 319–346.

Walton, D. N. (1996). *Argumentation Schemes for Presumptive Reasoning*, Mahwah (NJ): Lawrence Erlbaum Associates.

Walton, D., Reed, C. and Macagno, F. (2008). *Argumentation Schemes*, Cambridge University Press.

Floris Bex
Argumentation Research Group
School of Computing
University of Dundee
Dundee DD1 4HN Scotland, UK
florisbex@computing.dundee.ac.uk

Chris Reed
Argumentation Research Group
School of Computing
University of Dundee
Dundee DD1 4HN Scotland, UK
chris@computing.dundee.ac.uk