Argumentation Context Systems: A Framework for Abstract Group Argumentation

Gerhard Brewka[1] and Thomas Eiter[2]

[1] Universität Leipzig, Augustusplatz 10-11, 04109 Leipzig, Germany,
brewka@informatik.uni-leipzig.de
[2] Vienna University of Technology, Favoritenstraße 9-11, A-1040 Vienna, Austria,
eiter@kr.tuwien.ac.at

Abstract. We introduce a modular framework for distributed abstract argumentation where the argumentation context, that is information about preferences among arguments, values, validity, reasoning mode (skeptical vs. credulous) and even the chosen semantics can be explicitly represented. The framework consists of a collection of abstract argument systems connected via mediators. Each mediator integrates information coming from connected argument systems (thereby handling conflicts within this information) and provides the context used in a particular argumentation module. The framework can be used in different directions; e.g., for hierarchic argumentation as typically found in legal reasoning, or to model group argumentation processes.

1 Introduction

In his seminal paper, Dung [10] introduced an abstract framework for argumentation (sometimes referred to as calculus of opposition) which proved to be extremely useful for analyzing various kinds of argumentation processes. His approach gives a convincing account of how to select a set of "acceptable" arguments out of a set of arguments which may attack each other.

Dung's approach is monolithic in the sense that there are no means to structure a set of arguments any further. This is at odds with real world argumentation scenarios, be they informal as in everyday conversation, or institutionalized as in the legal domain. In such scenarios, one typically finds *meta-arguments*, i.e., arguments about other arguments, which can be clearly distinguished from arguments about the domain at hand. Moreover, in multi-agent scenarios it is often important to keep track of where certain arguments came from, who put them forward, who opposed and the like.

For these reasons, our interest in this paper is on adding more structure to formal models of argumentation. In doing so, we want to stick as much as possible to the idea of abstract argumentation. However, instead of a single, unstructured set of arguments we consider clearly distinguishable, distributed argumentation modules and formalize ways in which they can possibly interact. This has several benefits.

1. Even in the single-agent case, the additional structure provided by argumentation modules gives a handle on complexity and diversity (e.g., in our framework it will be possible for an agent to be skeptical in critical parts, credulous in less critical ones);
2. a distributed framework provides a natural account of multi-agent argumentation scenarios including information flow and knowledge integration methods;
3. modules provide explicit means to model meta-argumentation where the arguments in one module are about the arguments in another module; this leads to more realistic accounts of, say, legal argumentation processes.

In the simplest multi-module situation, one module determines the context for another. By a *context* we mean the available meta-information including, for instance, arguments which should not be taken into account, preferences among arguments, values, reasoning modes (skeptical vs. credulous), or even the semantics to be used in a particular situation. However, we also want to be able to capture more general situations where modules form complex hierarchies, and even cycles where modules mutually influence each other. For these reasons, we consider arbitrary directed graphs of modules.

As different "parent" modules in the graph can contribute to the context of a single module, we face the difficulty that the context information may become inconsistent. Each module is equipped with a *mediator*[3] to deal with this issue.

To model the flow of information among argumentation modules, we use techniques developed in the area of multi-context systems ($\mathcal{MCS}$s), in particular the systems devised by Giunchiglia and Serafini [14] and their extensions to nonmonotonic $\mathcal{MCS}$s [17, 6]. An $\mathcal{MCS}$ describes the information available in a number of contexts and specifies the information flow between them.[4] So-called *bridge rules* play a key role: they are used to provide input for a context based on the beliefs held in other relevant contexts. Since different contexts may use different representation languages, this may include "translating" information from one language to another. In our case, the bridge rules are necessary to transform abstract arguments accepted in one module into context statements for a child module.

Our approach substantially generalizes a framework recently introduced by Modgil [15]. His framework consists of a *linear* hierarchy of argument systems $(\mathcal{A}_1, \ldots, \mathcal{A}_n)$ and allows *preferences* among arguments in $\mathcal{A}_i$ to be established in $\mathcal{A}_{i+1}$. Our approach is more general in at least two respects: (1) we provide means for argumentation not only about preferences, but also about values, acceptability of arguments and attacks, and even reasoning mode and semantics, and (2) we consider arbitrary directed graphs. Since our modules may have multiple parents, methods for information integration, as provided by our mediators, become essential. In summary, our contribution in this paper is twofold:

- We introduce *context-based argumentation* (Sect. 4). Here statements in a context language, which formally specify a context, allow us to control argu-

[3] Cf. Wiederhold's [19] classic notion in information systems.

[4] A context in an $\mathcal{MCS}$ is a local inference system. This is in contrast with the more specific meaning of the term in this paper.

mentation processes by determining their semantics, reasoning modality, and arguments.
• Based on it, we develop *argumentation context systems ($\mathcal{ACS}$)*, which are composed of *argumentation modules* and *mediators* (Sect. 5) in a graph structure. A mediator collects context information for a module based on the arguments accepted by its parent modules. It then applies a consistency handling method, possibly based on preference information regarding its parents, to select a consistent subset of the context statements.

Both context-based argumentation and ACSs are generic and can be instantiated in various ways (we consider several of them).

2 Background

Abstract argumentation: We assume some familiarity with Dung-style abstract argumentation [10] and just recall the essential definitions. An argumentation framework is a pair $\mathcal{A} = (AR, \mathit{attacks})$ where AR is a set of arguments, and *attacks* is a binary relation on AR (used in infix in prose). An argument $a \in AR$ is *acceptable with respect to a set S of arguments*, if each argument $b \in AR$ that attacks a is attacked by some $b' \in S$. A set S of arguments is *conflict-free*, if there are no arguments $a, b \in S$ such that a attacks b, and S is *admissible*, if in addition each argument in S is acceptable wrt. S.

Dung defined three different semantics for $\mathcal{A} = (AR, \mathit{attacks})$:

- A *preferred extension* of $\mathcal{A}$ is a maximal (wrt. $\subseteq$) admissible set of $\mathcal{A}$.
- A *stable extension of* $\mathcal{A}$ is a conflict-free set of arguments S which attacks each argument not belonging to S.
- The *grounded extension of* $\mathcal{A}$ is the least fixpoint of the operator $F_{\mathcal{A}} : 2^{AR} \rightarrow 2^{AR}$ where $F_{\mathcal{A}}(S) = \{a \in AR \mid a \text{ is acceptable wrt. } S\}$.

The unique grounded extension is a subset of the intersection of all preferred extensions, and each stable extension is a preferred extension, but not vice versa. While the grounded and some preferred extension are guaranteed to exist, $\mathcal{A}$ may have no stable extension.[5]

A *preference based argumentation framework (PAF)* $\mathcal{P} = (AR, C, \geq)$ [1, 8] is based on a relation C representing logical conflicts between arguments AR and a reflexive, transitive preference relation $\geq$ on AR for expressing that arguments are stronger than others; as usual, the associated strict preference $>$ is given by $a > b$ iff $a \geq b$ and $b \not\geq a$. The PAF $\mathcal{P}$ induces an ordinary argumentation framework $\mathcal{A}_P = (AR, \mathit{attacks})$ where $\mathit{attacks} = \{(a, b) \in C \mid b \not> a\}$. The grounded, preferred, and stable extensions of $\mathcal{P}$ are then the respective extensions of $\mathcal{A}_P$.

Value based argumentation frameworks (VAFs) [2, 3] derive preferences among arguments from the values they promote. An audience specific[6] VAF

[5] We confine the discussion here to Dung's original semantics. Recent proposals like semi-stable [7] and ideal [11] semantics can be easily integrated in our framework.

[6] We omit discussing audiences; in our framework, they are best modeled by modules.

$$\mathcal{V} = (AR, attacks, V, val, valprefs)$$

extends an ordinary argumentation framework $(AR, attacks)$ with a non-empty set of values V, a function $val : AR \to V$, and a strict preference relation *valprefs* on V; argument a is preferred over b whenever $(val(a), val(b)) \in valprefs$. The preferences are then treated as in Amgoud & Cayrol's approach.

Inconsistency handling: Let $F = (F_1, \ldots, F_n)$ be a sequence of sets of formulas. We will use 4 different methods to generate a consistent subset of $F_1 \cup \ldots \cup F_n$ from F: a skeptical method $sub_{\succ}$ which goes back to [5] and uses a partial preference order $\succ$ on the sets F_i, together with its skeptical variant $sub_{sk,\succ'}$, and a majority-based method maj, together with its skeptical variant maj_{sk}. The former methods are used to integrate information in cases where some parent modules are more important than others, the latter in peer-to-peer situations where voting is more appropriate. As here the technical details are of minor relevance, we leave them for a longer version of the paper.

Multi-context systems: Such systems model, in the tradition of [14] and their nonmonotonic extensions [17, 6], the information flow between different reasoning modules (called *contexts*) using so-called *bridge rules*, which may refer to other modules in their bodies. For our purposes, we only need to consider rules referring to a single other module that is implicitly given. Therefore, our bridge rules will look like ordinary logic programming rules of the form:

$$s \leftarrow p_1, \ldots, p_j, \mathbf{not}\ p_{j+1}, \ldots, \mathbf{not}\ p_m \tag{1}$$

where the head s is a context expression (defined in the Sect. 4) and the body literals are arguments p_i (possibly negated with **not**) from a parent argumentation framework.

3 Motivating Examples

Legal reasoning: Different proof standards determine which arguments are acceptable depending on the type of a trial: *beyond reasonable doubt* for criminal trials; *preponderance of evidence* for civil trials (we omit *scintilla of evidence* for simplicity). Consider a situation where a judge J has to decide which of the arguments put forward by a prosecutor P are acceptable. Assume also that criminal trials require grounded reasoning, whereas less skeptical reasoning methods may be acceptable in civil trials.[7]

The arguments and attacks put forward by P form an argumentation framework $\mathcal{A} = (\{a_1, \ldots, a_n\}, attacks)$. The judge has information about which of P's arguments are doubtful and the type of a trial. Using arguments dt_i, bd_i and dr_i for a_i is doubtful, beyond reasonable doubt and disregarded, respectively, the judge's argumentation framework may look as in Fig. 1, where *crl* means criminal trial and *civ* civil trial (arcs are according to *attacks*).[8] Optional arguments represent information the judge may or may not have are in blue.

[7] A similar example was discussed in [21].

[8] For simplicity, names of abstract arguments in $\mathcal{M}_2$ reflect their intended meaning.

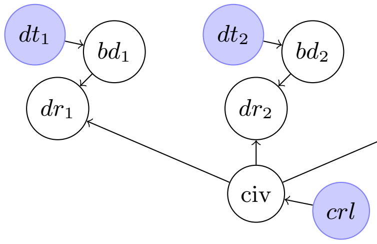

Fig. 1. The judge's argumentation framework for $n = 3$

The outcome of J's argumentation determines how P can argue, and sets a *context* for P; a language for representing contexts will be defined in the next section. As for now, let $\overline{\mathbf{arg}}(a_i)$ state that a_i is invalid for P, and that $\mathbf{sem}(grnd)$ fixes the semantics for P's argumentation framework to grounded semantics.

However, to build this context, we need to transform the relevant abstract arguments accepted by J into adequate context expressions for P that invalidate unusable arguments, pick the right semantics, etc. This is achieved by a *mediator*, which uses the following bridge rules $i \in \{1, \ldots, n\}$:

$$\{\ \overline{\mathbf{arg}}(a_i) \leftarrow dr_i, \quad \mathbf{sem}(grnd) \leftarrow \mathbf{not}\ civ\ \}.$$

Not all arguments of J need to be "visible" for P. Privacy issues (e.g. hiding or summarizing arguments) can be modeled by choosing adequate bridge rules.

The example involving judge, prosecutor and mediator is clearly simplistic and one might ask why the two argumentation frameworks are not simply merged into one. There are the following aspects, though: (1) It is non-obvious how, for instance, a statement fixing a particular semantics could be eliminated this way. (2) Even if we could compile a single argumentation framework out of the example, the distinction between arguments and meta-arguments and their origin would be blurred. (3) Most importantly, the framework we develop aims at capturing much more complex scenarios where such compilations appear neither doable nor fruitful. We now briefly discuss such a scenario.

Conference reviewing: Consider the paper review process for a modern AI conference. This process can typically be characterized as follows

- There is a hierarchy consisting of a PC chair, several area chairs, many reviewers, and even more authors.
- The PC chair determines the review criteria, acceptance rates etc.
- Area chairs make sure reviewers make fair judgements and eliminate unjustified arguments from their reviews.
- Authors give feedback on the preliminary reviews of their papers. Information flow is thus cyclic.
- Reviewers exchange arguments in a peer-to-peer discussion.
- Area chairs generate a consistent recommendation out of the final reviews.
- PC chair takes recommendations as input for final decision.

What we see here is a complex argumentation scenario including hierarchic (the PC chair setting criteria) as well as peer-to-peer (the reviewers discussing the same paper) forms of argumentation in groups. It is also evident that information flow is cyclic: even the authors nowadays are able to feed arguments back into higher levels of the reviewing hierarchy.

Examples like this call for a flexible framework allowing for cyclic structures encompassing a variety of information integration methods. Exactly this kind of framework is what we are going to develop in the rest of the paper.

4 Context-Based Argumentation

We now give a simple language for representing context and define what it means for a set of arguments to be acceptable for an argumentation framework $\mathcal{A}$ given a context C. In the context language, we want to specify various aspects, including: a) preferences among arguments; b) values and value orderings; c) validity/invalidity of specific arguments; d) addition/deletion of attack relationships among arguments; e) a reasoning mode (sceptical vs. credulous); and f) an argumentation semantics (stable, preferred, grounded).

Definition 1. *A* context expression *for a set of arguments AR and a set of values V has one of the following forms ($a, b \in AR$; $v, v' \in V$):*

$\boldsymbol{arg}(a)$ / $\overline{\boldsymbol{arg}}(a)$	*a is a valid / invalid argument*
$\boldsymbol{att}(a,b)$ / $\overline{\boldsymbol{att}}(a,b)$	*(a,b) is a valid / invalid attack*
$a > b$	*a is strictly preferred to b*
$\boldsymbol{val}(a,v)$	*the value of a is v*
$v > v'$	*value v is strictly better than v'*
$\boldsymbol{mode}(r)$	*the reasoning mode is $r \in \{skep, cred\}$*
$\boldsymbol{sem}(s)$	*the chosen semantics is $s \in \{grnd, pref, stab\}$*

A context *C is a set of context expressions (for given AR and V).*

The preference and value expressions together define a preference order on arguments as follows:

Definition 2. *For a context C, the preference order $>_C$* induced by C *is the smallest transitive relation such that $a >_C b$ if either (i) $a > b \in C$ or (ii) $\boldsymbol{val}(a, v_1) \in C$, $\boldsymbol{val}(b, v_2) \in C$, and (v_1, v_2) is in the transitive closure of $\{(v, v') \mid v > v' \in C\}$.*

A context C is *consistent*, if the following conditions hold:

1. $>_C$ is a strict partial order,
2. for no a both $\mathbf{arg}(a) \in C$ and $\overline{\mathbf{arg}}(a) \in C$,
3. for no (a,b) both $\mathbf{att}(a,b) \in C$ and $\overline{\mathbf{att}}(a,b) \in C$,
4. C contains at most one expression of the form $\mathbf{mode}(r)$; the same holds for $\mathbf{sem}(s)$ and for $\mathbf{val}(a,v)$, for each a.

Now we define the semantics of a consistent context. It acts as a modifier for an argumentation framework $\mathcal{A}$, which is evaluated under the argumentation semantics and reasoning mode specified in the context.

Definition 3. *Let $\mathcal{A} = (AR, attacks)$ be an argumentation framework, let V be a set of values, and let C be a consistent context for AR and V. The C-*modification *of $\mathcal{A}$ is the argumentation framework $\mathcal{A}^C = (AR^C, attacks^C)$, where*

- $AR^C = AR \cup \{\boldsymbol{def}\}$, *where* $\boldsymbol{def} \notin AR$.
- $attacks^C$ *is the smallest relation satisfying the following conditions:*
  1. *if* $\boldsymbol{att}(a, b) \in C$, *then* $(a, b) \in attacks^C$,
  2. *if* $(a, b) \in attacks$, $\overline{\boldsymbol{att}}(a, b) \notin C$ *and* $b \not>_C a$, *then* $(a, b) \in attacks^C$,
  3. *if* $\overline{\boldsymbol{arg}}(a) \in C$ *or* $(\boldsymbol{arg}(b) \in C \wedge (a, b) \in attacks^C)$ *then* $(\boldsymbol{def}, a) \in attacks^C$.

The basic idea is that the new, non-attackable argument **def** defeats invalid arguments as well as attackers of valid arguments. In this way, it is guaranteed that (in)valid arguments will indeed be (in)valid, independently of the chosen semantics. Moreover, the definition of $attacks^C$ guarantees that valid attacks are taken into account while invalid ones are left out. It also ensures that preferences among arguments are handled correctly, by disregarding any original attack (a, b) where b is more preferred than a. The preferences among arguments may be stated directly, or indirectly by the argument values and their preferences.

Example 1. Let $\mathcal{A} = (\{a, b, c, d\}, attacks)$ with $attacks = \{(a, b), (b, d), (d, b), (b, c)\}$. Moreover, let $C = \{\overline{\mathbf{arg}}(a), \mathbf{val}(b, v_1), \mathbf{val}(d, v_2), v_1 > v_2, c > b\}$. We obtain $\mathcal{A}^C = (\{a, b, c, d, \mathbf{def}\}, \{(\mathbf{def}, a), (a, b), (b, d)\})$.

Based on the C-modification, we define the sets of acceptable arguments. We adopt credulous reasoning and preferred semantics by default, i.e., whenever a context C contains no expression of the form $\mathbf{mode}(m)$ (resp., $\mathbf{sem}(s)$), we implicitly assume $\mathbf{mode}(cred) \in C$ (resp., $\mathbf{sem}(pref) \in C$).

Definition 4. *Let $\mathcal{A} = (AR, attacks)$ be an argumentation framework, let V be a value set, and let C be a consistent context for AR and V such that $\boldsymbol{sem}(s) \in C$, $\boldsymbol{mode}(m) \in C$. A set $S \subseteq AR$ is an* acceptable C-extension for $\mathcal{A}$, *if either*

- $m = cred$ *and* $S \cup \{\boldsymbol{def}\}$ *is an s-extension*[9] *of* $\mathcal{A}^C$, *or*
- $m = skep$ *and* $S \cup \{\boldsymbol{def}\}$ *is the intersection of all s-extensions of* $\mathcal{A}^C$.

We call a context *purely preferential* (resp., *purely value-based*), if it contains besides **mode**- and **sem**-statements only expressions of form $a > b$ (resp., $\mathbf{val}(a,v)$ or $v > v'$). The next proposition shows that our definition "does the right thing:"

Proposition 1. *Suppose C, $\mathcal{A}$ and V are as in Definition 4. Then,*

1. *if C is purely preferential, the acceptable C-extensions of $\mathcal{A}$ coincide with the s-extensions of the PAF $\mathcal{P} = (AR, attacks, \geq_{\mathcal{P}})$, if the strict partial order induced by $\geq_{\mathcal{P}}$ coincides with $>_C$.*
2. *if C is purely value-based and $AR = \{a \mid \boldsymbol{val}(a, v) \in C\}$, the acceptable C-extensions of $\mathcal{A}$ coincide with the s-extensions of the VAF $\mathcal{V} = (AR, attacks, V, val, valprefs)$ where $val(a) = v$ iff $\boldsymbol{val}(a, v) \in C$ and $(v, v') \in valprefs$ iff (v, v') is in the transitive closure of $\{(v, v') \mid v > v' \in C\}$.*
3. *if $\boldsymbol{arg}(a) \in C$ (resp., $\overline{\boldsymbol{arg}}(a) \in C$) and S is an acceptable C-extension, then $a \in S$ (resp., $a \notin S$).*

[9] s-extension means grounded, preferred, stable extension if $s = grnd, pref, stab$, resp.

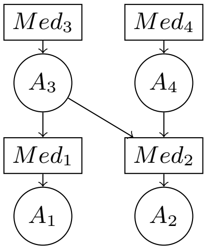

Fig. 2. Example argumentation context system $\mathcal{F} = \{(\mathcal{A}_1, Med_1) \mid 1 \leq i \leq 4\}$

5 Argumentation Context Systems

We now develop *argumentation context systems* ($\mathcal{ACS}$s) as a flexible framework for distributed argumentation, encompassing arbitrary directed graphs of *argumentation modules*. Such modules $\mathcal{M} = (\mathcal{A}, Med)$ consist of a Dung-style argument framework $\mathcal{A}$ and a *mediator* Med, which determines a context C for $\mathcal{A}$. To this end, it translates arguments accepted by $\mathcal{M}$'s parents (i.e., its direct ancestor modules) into context expressions using bridge rules and combines them with local context information. To ensure consistency of C, the mediator uses a *consistency handling method* that resolves any inconsistency between the local and the parent information (multiple parents are possible). Figure 2 shows the structure of an example $\mathcal{ACS}$. The module $\mathcal{M}_1$ has the single parent $\mathcal{M}_3$, and Med_1 receives input from $\mathcal{A}_3$; we say Med_1 is *based on* $\mathcal{A}_3$. The module $\mathcal{M}_2$ has the parents $\mathcal{M}_3$ and $\mathcal{M}_4$, and Med_2 is based on $\mathcal{A}_3$ and $\mathcal{A}_4$.

5.1 Mediators

We now define precisely what we mean by a mediator.

Definition 5. *Let $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_k$, $k \geq 1$, be argumentation frameworks. A* mediator for $\mathcal{A} = \mathcal{A}_1$ based on $\mathcal{A}_2, \ldots, \mathcal{A}_k$ *is a tuple*

$$Med = (E_1, R_2, \ldots, R_k, choice)$$

where

- *E_1 is a set of context expressions for $\mathcal{A}$;*
- *R_i, $2 \leq i \leq k$, is a set of rules of form (1) where s is a context expression for $\mathcal{A}$ and $p_1, \ldots, p_m$ are arguments in $\mathcal{A}_i$ (*bridge rules for $\mathcal{A}$ based on $\mathcal{A}_i$*);*
- *$choice \in \{\, sub_{\succ}, sub_{sk,\succ}, maj, maj_{sk}\}$, where $\succ$ is a strict partial order on $\{1, \ldots, k\}$.*

Intuitively, the $\mathcal{A}_i$, are from the parent modules $\mathcal{M}_i$ of $\mathcal{M}$. The set E_1 contains context statements for $\mathcal{A}$ which the mediator itself considers adequate; it is also used to provide the context for modules without parents. The bridge rules determine the relevant context expressions for $\mathcal{A}$ based on arguments accepted in $\mathcal{A}_i$ (i.e., in $\mathcal{M}_i$). This overcomes the following problem: argumentation in Dung-style systems is *abstract*, i.e., the content of arguments is not analyzed. In order to use arguments of a parent module in determining the context for a child module, this abstract view must be given up to some extent, as context expressions must be associated with arguments of the parent modules.

The parameter *choice* fixes a inconsistency handling strategy. The order $\succ$ is used to prioritize context expressions from different modules. Index 1 is included in the ordering; this makes it possible to specify whether statements in E_1 can be overridden by statements coming from parent modules or not. For the majority-based inconsistency methods *maj* and maj_{sk}, the order is not needed since the number of supporting modules is implicitly used as the preference criterion. Other consistency methods than those considered can be easily integrated.

Given sets of accepted arguments for all parent modules of $\mathcal{M}$, the mediator defines the consistent acceptable contexts for its argument system $\mathcal{A}$. There may be more than one acceptable context.

Let R_i be a set of bridge rules for $\mathcal{A}$ based on $\mathcal{A}_i$, and S_i a set of arguments of $\mathcal{A}_i$. $R_i(S_i)$ is the set of context expressions

$$\begin{aligned} R_i(S_i) = \{h \mid \;\; & h \leftarrow a_1, \ldots, a_j, \mathbf{not}\ b_1, \ldots, \mathbf{not}\ b_n \ \in R_i, \\ & \{a_1, \ldots, a_j\} \subseteq S_i,\ \{b_1, \ldots, b_n\} \cap S_i = \emptyset\,\}, \end{aligned}$$

Intuitively, $R_i(S_i)$ contains the context statements for $\mathcal{A}$ derivable through rules in R_i given the arguments S_i are accepted by $\mathcal{A}_i$.

The sets $E_1, R_2(S_2), \ldots, R_k(S_k)$ provide context information for $\mathcal{A}$ coming from the mediator for $\mathcal{A}$, respectively from the argument frameworks $\mathcal{A}_2, \ldots, \mathcal{A}_k$ in the parent modules. This information is integrated into a consistent context using the chosen consistency handling method.

Definition 6. *Let* $Med = (E_1, R_2, \ldots, R_k, choice)$ *be a mediator for* $\mathcal{A}$ *based on* $\mathcal{A}_2, \ldots, \mathcal{A}_k$. *A context* C *for* $\mathcal{A}$ *is* acceptable wrt. sets of arguments $S_2, \ldots, S_k$ of $\mathcal{A}_2, \ldots, \mathcal{A}_k$*, if* C *is a choice-preferred set for* $(E_1, R_2(S_2), \ldots, R_k(S_k))$.

Example 2. Consider a mediator $Med = (E_1, R_2, R_3, sub_{\succ})$ based on argumentation frameworks $\mathcal{A}_2$ and $\mathcal{A}_3$. Let $E_1 = \{\mathbf{sem}(grnd), v > v'\}$ and $1 \succ 2 \succ 3$. Let $R_2 = \{\mathbf{arg}(a_1) \leftarrow b_1,\ a_2 > a_3 \leftarrow b_2\}$ and $R_3 = \{\,\overline{\mathbf{arg}}(a_1) \leftarrow c_1,\ \mathbf{val}(a_2, v') \leftarrow c_2,$ $\mathbf{val}(a_3, v) \leftarrow c_3\,\}$. Suppose $S_2 = \{b_1, b_2\}$ is accepted in $\mathcal{A}_2$ and $S_3 = \{c_1,\ c_2,$ $c_3\}$ in $\mathcal{A}_3$. We obtain $R_2(S_2) = \{\mathbf{arg}(a_1),\ a_2 > a_3\}$ and $R_3(S_3) = \{\overline{\mathbf{arg}}(a_1),$ $\mathbf{val}(a_2, v'),\ \mathbf{val}(a_3, v)\}$. Note that $E_1 \cup R_2(S_2)$ is consistent. From the least preferred context information $R_3(S_3)$, only $\mathbf{val}(a_2, v')$ or $\mathbf{val}(a_3, v)$ can be consistently added to the more preferred context information. The acceptable contexts are $C_1 = E_1 \cup R_2(S_2) \cup \{\mathbf{val}(a_2, v')\}$ and $C_2 = E_1 \cup R_2(S_2) \cup \{\mathbf{val}(a_3, v)\}$.

5.2 The framework

We are now in a position to introduce our framework.

Definition 7. *An* module $\mathcal{M} = (\mathcal{A}, Med)$ *consists of an argument framework* $\mathcal{A}$ *and a mediator* Med *for* $\mathcal{A}$ *(based on some argumentation frameworks).*

An argument context system is a collection of modules that fulfills certain structural conditions.

Definition 8. *An* argumentation context system ($\mathcal{ACS}$) *is a set*[10]

$$\mathcal{F} = \{\mathcal{M}_1, \dots, \mathcal{M}_n\}$$

of modules $\mathcal{M}_i = (\mathcal{A}_i, Med_i)$*,* $1 \leq i \leq n$*, where* Med_i *is based on argumentation frameworks* $\mathcal{A}_{i_1}, \dots, \mathcal{A}_{i_{k_i}}$*,* $\{i_1, \dots, i_{k_i}\} \subseteq \{1, \dots, n\}$ (self-containedness)*.*

Self-containedness naturally induces the following graph and structural notion.

Definition 9. *The* module graph *of an* $\mathcal{ACS}$ $\mathcal{F}$ *as in Definition 8 is the directed graph* $G(\mathcal{F}) = (\mathcal{F}, E)$ *where* $\mathcal{M}_j \to \mathcal{M}_i$ *is in* E *iff* $\mathcal{A}_j$ *is among the* $\mathcal{A}_{i_1}, \dots, \mathcal{A}_{i_{k_i}}$ *on which* Med_i *is based. We call* $\mathcal{F}$ hierarchic*, if* $G(\mathcal{F})$ *is acyclic.*

We next define the acceptable states for our framework. Intuitively, such a state consists of a context and a set of arguments for all modules $\mathcal{M}_i$ such that in each case the chosen arguments form an acceptable set for $\mathcal{A}_i$ given the respective context, and this context (determined by Med_i) is acceptable with respect to the argument sets chosen for the parent modules of $\mathcal{A}_i$. More formally,

Definition 10. *Let* $\mathcal{F} = \{\mathcal{M}_1, \dots, \mathcal{M}_n\}$ *be an* $\mathcal{ACS}$*. A* state *of* $\mathcal{F}$ *is a function* $\mathcal{S}$ *that assigns each* $\mathcal{M}_i = (\mathcal{A}_i,\ Med_i)$ *a pair* $\mathcal{S}(\mathcal{M}_i) = (Acc_i, C_i)$ *of a subset* Acc_i *of the arguments of* $\mathcal{A}_i$ *and a set* C_i *of context expressions for* $\mathcal{A}_i$*.*

A state $\mathcal{S}$ *is* acceptable*, if (i) each* Acc_i *is an acceptable* C_i*-extension for* $\mathcal{A}_i$*, and (ii) each* C_i *is an acceptable context for* Med_i *wrt. all* Acc_j *such that* $\mathcal{M}_j \to \mathcal{M}_i$*.*

As stable semantics may be chosen for argumentation frameworks in modules, clearly an acceptable state is not guaranteed to exist. However, non-existence can arise even in absence of stable semantics and negation in bridge rules.

Example 3. Let $\mathcal{F} = (\mathcal{M}_1, \mathcal{M}_2)$ with modules $\mathcal{M}_i = (\mathcal{A}_i, Med_i)$, $i = 1, 2$, where

- $\mathcal{A}_1 = (\{a\}, \emptyset)$, $\quad Med_1 = (\{\mathbf{sem}(s_1)\}, \{\overline{\mathbf{arg}}(a) \leftarrow b_2\}, c_1)$;
- $\mathcal{A}_2 = (\{b_1, b_2\}, \{(b_1, b_2)\})$, $\quad Med_2 = (\{\mathbf{sem}(s_2)\}, \{\overline{\mathbf{arg}}(b_1) \leftarrow a\}, c_2)$.

Note that $G(\mathcal{F})$ is cyclic, as $\mathcal{M}_1 \to \mathcal{M}_2$ and $\mathcal{M}_2 \to \mathcal{M}_1$. Now for arbitrary s_1, c_1 and s_2, c_2, no acceptable state exists: assume $Acc_1 = \{a\}$. Using its single bridge rule, Med_2 will derive $\overline{\mathbf{arg}}(b_1)$, which is consistent with $\mathbf{sem}(s_2)$ and will belong to the context C_2, regardless of the chosen consistency method c_2. This, however, will lead to acceptance of b_2 independently of the semantics s_2, as the only potential attacker of b_2, namely b_1, is invalid. Now, if b_2 is in Acc_2, then the single bridge rule of Med_1 invalidates argument a; hence, there is no acceptable state with $Acc_1 = \{a\}$.

The other option, namely $Acc_1 = \emptyset$, also fails: now b_1 is not invalidated in $\mathcal{M}_2$ and thus defeats b_2. As a consequence, Med_1's bridge rule does not fire and there is no justification not to accept argument a in $\mathcal{A}_1$.

To guarantee existence of acceptable states, we need more conditions; for example, excluding stable semantics and cycles between modules.

[10] For multiple occurrences of the same module, this is viewed as an indexed set.

Proposition 2. *Suppose $\mathcal{F}$ is a hierarchic $\mathcal{ACS}$ and that* $\boldsymbol{sem}(stab)$ *does not occur in any mediator of $\mathcal{F}$. Then $\mathcal{F}$ possesses at least one acceptable state.*

Computing some acceptable state is still intractable in this setting, due to the intractability of recognizing preferred extensions [9]. However,

Proposition 3. *If $\mathcal{F}$ is a hierarchic $\mathcal{ACS}$ and all modules use grounded semantics and either $sub_{\succ}$ or maj for choice, then some acceptable state of $\mathcal{F}$ is computable in polynomial time.*

Regarding the complexity in the general case, we note the following result.

Proposition 4. *Deciding whether an arbitrary given $\mathcal{ACS}$ $\mathcal{F}$ has some acceptable state is Σ_3^p-complete.*

Intuitively, an acceptable state can be guessed and verified in polynomial time with the help of an Σ_2^p oracle; note that the reasoning tasks in Definition 4 are all feasible polynomial time with an Σ_2^p oracle. On the other hand, skeptical inference from all preferred extensions is Π_2^p-complete [12], which in combination with the module framework generates the Σ_3^p-hardness. This can be shown by a reduction from suitable quantified Booleans formulas, which uses no negation in bridge rules and and an arbitrary inconsistency handling method *choice*.

Depending on the various parameters and the graph structure, the complexity decreases. The complexity of C-extensions is dominated by the underlying argumentation framework, and in several situations, the mediator framework does not increase complexity. A more detailed analysis is left for further work.

5.3 Relationship to EAFs

Modgil [16] recently introduced an interesting extension of argumentation frameworks where arguments may not only attack other arguments, but also attacks. An *extended argumentation framework* (EAF) $\mathcal{E} = (AR, attacks, D)$ adds to a Dung-style argumentation framework $(AR, attacks)$ a set $D \subseteq AR \times attacks$ of *attacks against attacks*. Whenever $(a_1, (b, c)) \in D$ and $(a_2, (c, b)) \in D$ it is required that *attacks* contains both (a_1, a_2) and (a_2, a_1). $S \subseteq AR$ is conflict free iff for all $a, b \in S$: if $(a, b) \in attacks$, then $(b, a) \notin attacks$ and there is $c \in S$ such that $(c, (a, b)) \in D$. Argument a S-defeats b iff $(a, b) \in attacks$ and, for no $c \in S$, $(c, (a, b)) \in D$. S is a stable extension of $\mathcal{E}$ iff S is conflict free and each $c \in AR \setminus S$ is S-defeated by some element of S. We refer to [16] for numerous interesting applications of EAFs.

We can show that under stable semantics, any EAF $\mathcal{E}$ can be modeled as an $\mathcal{ACS}$ $\mathcal{F}(\mathcal{E})$ that consists of a single module with self-feedback. In detail, we construct $\mathcal{F}(\mathcal{E}) = \{ \mathcal{M}^* \}$, where $\mathcal{M}^* = (\mathcal{A}^*, Med^*)$ such that $\mathcal{A}^* = (AR, attacks)$ and $Med^* = (\{\mathbf{sem}(stab), \mathbf{mode}(cred)\}, R, choice)$ is based on $\mathcal{A}^*$, where

$$\begin{aligned} R = \; & \{ \overline{\mathbf{att}}(a, b) \leftarrow c, b \mid (c, (a, b)) \in D, (b, a) \notin attacks \} \cup \\ & \{ \overline{\mathbf{att}}(a, b) \leftarrow c, \mathbf{not}\ b \mid (c, (a, b)) \in D \}. \end{aligned}$$

and *choice* is arbitrary. The first type of rules in R handles conflict freeness, while the second ensures that an argument can only be defeated by an attack which is not successfully attacked.

Proposition 5. *Let $\mathcal{E}$ be an EAF. Then, (i) for each acceptable state S of $\mathcal{F}(\mathcal{E})$ with $S(\mathcal{M}^*) = (T, C)$, T is a stable extension of $\mathcal{E}$, and (ii) for each stable extension T of $\mathcal{E}$, $\mathcal{F}(\mathcal{E})$ has some acceptable state S such that $S(\mathcal{M}^*) = (T, C)$.*

The relationship under the other semantics is trickier; this is basically due to the fact that Modgil and Dung have very different notions of argument acceptability. A thorough investigation is an interesting topic for future work.

6 Related Work and Conclusion

We presented a flexible, modular framework for abstract argumentation. It builds on existing proposals extending them in various respects: argumentation is based on contexts described in a native language, integrating preference- and value-based argumentation, direct (in)validation of arguments and attacks, and specification of reasoning mode and semantics. Context information is integrated by a mediator. Arbitrary directed module graphs cover a wide range of applications involving multi-agent meta-argumentation.

Models of meta- and hierarchic argumentation are not new. The approaches by Modgil [15, 16] were already discussed. Wooldridge, McBurney and Parsons [20] develop a meta-level approach which shares motivation with our work, but has different focus: a meta-level defines arguments, attacks, extensions etc. for the lower level based on formal provability in its logic. In contrast, we take the basic notions for granted and determine, at a generic level, how a module in an arbitrary module graph influences argumentation in others by determining its context. The many parameter combinations yield then a range of concrete systems with different properties.

A framework for distributed argumentation was presented in [18], based on defeasible argumentation as in [13], where a moderator integrates the argumentation structures of a group of agents. In contrast, we stick to abstract argumentation and allow for more general relationships between modules.

Our approach also differs from recent work by Binas and McIlraith [4] on distributed query answering: while our mediators use “classical” preference based inconsistency methods to establish argumentation contexts, they use techniques based on prioritized argumentation to define distributed entailment for graphs of heterogeneous reasoners. Moreover, they focus on information integration rather than meta-reasoning as we do.

Our framework specializes the multi-context systems of Brewka and Eiter [6] by fixing Dung style argument systems as reasoning components. The use of mediators to integrate meta-information and the context language to control argumentation clearly goes beyond these systems.

Our future work includes an investigation of more expressive mediator languages (both in terms of constructs and bridge rules used), and a detailed study of computational aspects, comprising complexity and implementation.

References

1. Leila Amgoud and Claudette Cayrol. On the acceptability of arguments in preference-based argumentation. In *Proc. Fourteenth Conference on Uncertainty in Artificial Intelligence, UAI-98*, pages 1–7, 1998.
2. Trevor J. M. Bench-Capon. Value-based argumentation frameworks. In *Proc. 9th International Workshop on Non-Monotonic Reasoning, NMR-02, Toulouse, France*, pages 443–454, 2002.
3. Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.
4. Arnold Binas and Sheila McIlraith. Peer-to-peer query answering with inconsistent knowledge. In *Proc. 11th International Conference on Principles of Knowledge Representation and Reasoning, KR-2008, Sydney, Australia*, page 329339, 2008.
5. Gerhard Brewka. Preferred subtheories: An extended logical framework for default reasoning. In *Proc. IJCAI-89*, pages 1043–1048, 1989.
6. Gerhard Brewka and Thomas Eiter. Equilibria in heterogeneous nonmonotonic multi-context systems. In *Proc. AAAI-07*, pages 385–390, 2007.
7. Martin Caminada. Semi-stable semantics. In *Proc. Computational Models of Argument, COMMA-06*, pages 121–130, 2006.
8. Yannis Dimopoulos, Pavlos Moraitis, and Leila Amgoud. Theoretical and computational properties of preference-based argumentation. In *Proc. ECAI-08, Patras, Greece*, pages 463–467, 2008.
9. Yannis Dimopoulos and Alberto Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comput. Sci.*, 170(1-2):209–244, 1996.
10. Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
11. Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artif. Intell.*, 171(10-15):642–674, 2007.
12. Paul E. Dunne and Trevor J. M. Bench-Capon. Coherence in finite argument systems. *Artif. Intell.*, 141(1/2):187–203, 2002.
13. Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming: An argumentative approach. *TPLP*, 4(1-2):95–138, 2004.
14. Fausto Giunchiglia and Luciano Serafini. Multilanguage hierarchical logics, or: how we can do without modal logics. *Artificial Intelligence*, 65(1):29–70, 1994.
15. Sanjay Modgil. Hierarchical argumentation. In *Proc. JELIA-06*, pages 319–332, 2006.
16. Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artif. Intell.*, to appear, 2007.
17. Floris Roelofsen and Luciano Serafini. Minimal and absent information in contexts. In *Proc. IJCAI-05*, pages 558–563, 2005.
18. Matthias Thimm and Gabriele Kern-Isberner. A distributed argumentation framework using defeasible logic programming. In *Proc. Computational Models of Argument, COMMA-08*, pages 381–392, 2008.
19. Gio Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25(3):38–49, 1992.
20. Michael Wooldridge, Peter McBurney, and Simon Parsons. On the meta-logic of arguments. In *Proc. AAMAS-05*, pages 560–567, 2005.
21. Adam Zachary Wyner and Trevor J. M. Bench-Capon. Modelling judicial context in argumentation frameworks. In *Proceedings COMMA-08*, pages 417–428, 2008.