Handling Inconsistency on Ontologies Through a Generalized Dynamic Argumentation Framework

Martín O. Moguillansky Nicolás D. Rotstein Marcelo A. Falappa

Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
Department of Computer Science and Engineering (DCIC)
Artificial Intelligence Research and Development Laboratory (LIDIA)
Universidad Nacional del Sur (UNS), Bahía Blanca, ARGENTINA
E-MAIL: {mom,ndr,maf}@cs.uns.edu.ar

Abstract

In this article we present a generalized dynamic argumentation framework that handles arguments expressed in an abstract language assumed to be some first order logic fragment. Once the formalism is presented, we propose a reification to the description logic $\mathcal{ALC}$ with the intention to handle ontology debugging. In this sense, since argumentation frameworks reason over graphs that relate arguments through attack, our methodology is proposed to bridge ontological inconsistency sources to attack relations in argumentation. Finally, an argumentation semantics is proposed as a consistency restoration tool to cope with the ontology debugging.

Keywords: Knowledge Representation, Argumentation, Ontology Debugging, Description Logics, Semantic Web.

1 Introduction

In this article, the original dynamic argumentation frameworks (DAF) [16, 17] are generalized with the intention to handle knowledge represented in fragments of first order logic (FOL). It is important to mention that DAFs are extensions of the abstract argumentation framework proposed by Dung [5], enriched to cope with the dynamics of arguments. Given the preliminary state of the current investigations, we will analyze the generalized DAF wrt. a specific fragment of FOL named $\mathcal{L}^2$. Such a logic is the subset of first order predicate calculus (FOPC) with monadic and dyadic predicates allowing only two variable symbols and without considering functional letters. Decidability of $\mathcal{L}^2$ has been shown in [12]. Recall that monadic (dyadic) predicates, are predicates taking one (two) parameter(s).

Afterwards, the generalized DAF will be reified to the description logic (DL) $\mathcal{ALC}$ for ontologies, given that any concept description in such a logic can be translated into an $\mathcal{L}^2$ formula with one free variable. This result is given in several articles like [4] and [1]. Besides, in [8] an extension of the $\mathcal{ALC}$ DL is presented and proved to be equivalent to $\mathcal{L}^2$.

Considering an $\mathcal{ALC}$ ontology into an $\mathcal{ALC}$-Based DAF requires to relate some classical argumentation elements [5] like attack and support to identify different levels of inconsistency in ontologies

[6]. Thereafter, an acceptability semantics [3] applied to the $\mathcal{ALC}$-Based DAF is proposed to determine a methodology for ontology debugging.

It is important to note that the argumentation machinery here proposed is semantically determined –by effect of the semantic entailment. This would allow to propose further implementations relying on the reasoner used for such a matter. For instance, an $\mathcal{ALC}$-Based DAF may be handled via tableaux technics usually used to implement ontology reasoners. Consequently, the actual ontology debugging model could be performed by emulating the argumentation machinery, without using it explicitly. In this sense, this methodology could be considered as theoretical.

2 Generalized Dynamic Argumentation Framework

For the language $\mathcal{L}^2$ in consideration, we will use symbols $A, A_1, A_2, \ldots$ and $B, B_1, B_2, \ldots$ to denote monadic (unary) predicate letters, $R, R_1, R_2, \ldots$ to denote dyadic (binary) predicate letters, x, y to denote free variable objects, and a, b, c, d to denote individual names. Recall that formulae in $\mathcal{L}^2$ are those of FOPC that can be built with the help of predicate symbols with arity ≤ 2, including equality and constant symbols, but without function symbols. Note that constant symbols are not specified since they can be simulated through monadic predicates. Besides, we refer as $\mathcal{L}_{\mathcal{A}}$ to the sub-language of $\mathcal{L}^2$ identifying atomic formulae.

The logic $\mathcal{L}^2$ is interpreted as usual by means of interpretations of the form $\mathcal{I} = \langle \Delta^{\mathcal{I}}, A^{\mathcal{I}}, A_1^{\mathcal{I}}, \ldots, B^{\mathcal{I}}, B_1^{\mathcal{I}}, \ldots, R^{\mathcal{I}}, R_1^{\mathcal{I}}, \ldots \rangle$, where $\Delta^{\mathcal{I}}$ is the interpretation domain, $A^{\mathcal{I}}, A_1^{\mathcal{I}}, \ldots, B^{\mathcal{I}}, B_1^{\mathcal{I}}, \ldots, R^{\mathcal{I}}, R_1^{\mathcal{I}}, \ldots$ interpret $A, A_1, \ldots, B, B_1, \ldots, R, R_1, \ldots$, respectively. For an interpretation $\mathcal{I}$, some $a \in \Delta^{\mathcal{I}}$ and a formula $\varphi(x)$, we write $\mathcal{I} \models \varphi(a)$ if $\mathcal{I}, v \models \varphi(x)$, for the assignment v mapping x to a.

In order to specify a generalized DAF, it is necessary to abstract away from the language used to represent arguments. An (abstract) argument language, referred as $\mathbb{A}\mathtt{rgs}$, needs to be characterized regarding its inner components: the languages for claims and premises. Therefore, it is necessary to state some restrictions in order to abstractly characterize the argument language used into the generalized DAF.

Definition 1 (Argument Language) *Given a logic* $\mathbb{A}\mathtt{rgs} ::= 2^{\mathcal{L}_{\mathfrak{pr}}} \times \mathcal{L}_{\mathfrak{cl}}$ *with* $\mathcal{L}_{\mathfrak{cl}}$ *and* $\mathcal{L}_{\mathfrak{pr}}$ *two different fragments of* $\mathcal{L}^2$. $\mathbb{A}\mathtt{rgs}$ *is a legal* ***argument language*** *iff for every statement* $p \in \mathcal{L}_{\mathfrak{pr}}$ *there exists some set* $C \subseteq 2^{\mathcal{L}_{\mathfrak{cl}}}$ *such that* $C \models p$. *Thus, the languages* $\mathcal{L}_{\mathfrak{cl}}$ *and* $\mathcal{L}_{\mathfrak{pr}}$ *related by* $\mathbb{A}\mathtt{rgs}$ *are the languages for claims and premises in* $\mathbb{A}\mathtt{rgs}$*, respectively.*

Recall that an argument is considered an atomic (indivisible) piece of knowledge. To the argumentation machinery, an argument is a primitive element of reasoning supporting a claim from its set of premises. Usually, argumentation frameworks consider ground arguments, that is, a claim is directly inferred if the set of premises are conformed. In our framework, we consider two different kinds of arguments: ground and schematic. In this sense, a set of premises might consider free variables, meaning that the claim, and therefore the inference, will depend on them. When an argument has its premises supported, its variables may be instantiated as a result of that. These notions are carefully detailed throughout this section. Next we formalize the generalized notions of argument and the DAF.

Definition 2 (Argument) *An argument* $\mathcal{B} \subseteq \mathbb{A}\mathtt{rgs}$ *is a pair* $\langle P, c \rangle$, *where* $P \subseteq 2^{\mathcal{L}_{\mathfrak{pr}}}$ *is the finite set of premises from* $\mathcal{L}_{\mathfrak{pr}}$, *and* $c \in \mathcal{L}_{\mathfrak{cl}}$, *the claim. An argument* $\mathcal{B}$ *guarantees* $P \cup \{c\} \not\models \bot$ **(consistency)**. *Both premises and claims are represented as finite formulae from their respective language.*

Definition 3 (Generalized Dynamic Argumentation Framework) *Let* $T \subseteq 2^{\mathbb{A}\mathtt{rgs}} \times 2^{\mathbb{A}\mathtt{rgs}}$ *be a Generalized Dynamic Argumentation Framework* (DAF)*, specified by a pair* $\langle \mathbf{U}, \mathbf{A} \rangle$*, where* $\mathbf{U} \subseteq 2^{\mathbb{A}\mathtt{rgs}}$ *is* *the universal set of arguments, and $\mathbf{A} \subseteq \mathbf{U}$ is the framework's active set containing the unique set of arguments considered by the argumentation reasoning process.*

As usual, pairs of conflictive arguments may appear. Such pairs will be contained in an attack relation set $\mathbf{R}$, dynamically recognized from the current DAF specification. This notion will be made clear later. Besides, inactive arguments –ignored by the reasoning process– might be identified by means of a set $\mathbf{I} = \mathbf{U} \setminus \mathbf{A}$. We will consider *evidence* as the basic available piece of knowledge that needs no premises to support it. Thus, active arguments with no premises to be satisfied will be referred as evidence, enclosed in a set $\mathbf{E} \subseteq \mathbf{A}$. Those arguments with an empty set of premises that are inactive will be recognized as *non-evidential facts*, and will be held in a set $\mathbf{F} \subseteq \mathbf{I}$.

Definition 4 (Evidence & Non-evidential Fact) *Given a* DAF $\langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathtt{rgs}} \times 2^{\mathbb{A}\mathtt{rgs}}$*, an argument $\mathcal{B} \in \mathbf{U}$ such that $\mathcal{B} = \langle \{\}, c \rangle$ and $c \in \mathcal{L}_{\mathcal{A}}$, is referred either as:* ***Evidence*** *iff $\mathcal{B} \in \mathbf{A}$, or* ***Non-evidential Fact*** *iff $\mathcal{B} \notin \mathbf{A}$.*

Given an argument $\mathcal{B} \in \mathbb{A}\mathtt{rgs}$, its claim and set of premises are identified by the functions $\mathfrak{cl} : \mathbb{A}\mathtt{rgs} \rightarrowtail \mathcal{L}_{\mathfrak{cl}}$, and $\mathfrak{pr} : \mathbb{A}\mathtt{rgs} \rightarrowtail 2^{\mathcal{L}_{\mathfrak{pr}}}$, respectively. For instance, given $\mathcal{B} = \langle \{p_1, p_2\}, c \rangle$, its premises are $\mathfrak{pr}(\mathcal{B}) = \{p_1, p_2\}$, and its claim, $\mathfrak{cl}(\mathcal{B}) = c$.

An argument needs to find its premises supported as a functional part of the reasoning process to reach its claim. In this framework, due to the logic used to represent arguments derived from that of the ontology languages, a single argument is sometimes not enough to support a premise. This is the reason why we introduce the notion of coalitions: to identify a minimal set of arguments verifying some specific properties. For instance, a coalition $\widehat{\mathcal{C}} \subseteq 2^{\mathbb{A}\mathtt{rgs}}$ may provide support for an argument $\mathcal{B} \in \mathbb{A}\mathtt{rgs}$ through some of its premises. For that matter, we present the functions $\widehat{\mathfrak{clset}}(\widehat{\mathcal{C}}) = \{\mathfrak{cl}(\mathcal{B}) | \mathcal{B} \in \widehat{\mathcal{C}}\}$, and $\widehat{\mathfrak{prset}}(\widehat{\mathcal{C}}) = \bigcup_{\mathcal{B} \in \widehat{\mathcal{C}}} \mathfrak{pr}(\mathcal{B})$.

Definition 5 (Supporter) *Given a* DAF $\langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathtt{rgs}} \times 2^{\mathbb{A}\mathtt{rgs}}$*, and an argument $\mathcal{B} \in \mathbf{U}$ such that $p \in \mathfrak{pr}(\mathcal{B})$. A set of arguments $\widehat{\mathcal{C}} \subseteq \mathbf{U}$ is a supporting-coalition, or just a* ***supporter****, of $\mathcal{B}$ through p iff it guarantees:*

(support) $\widehat{\mathfrak{clset}}(\widehat{\mathcal{C}}) \models p$,

(consistency) $\widehat{\mathfrak{prset}}(\widehat{\mathcal{C}}) \cup \widehat{\mathfrak{clset}}(\widehat{\mathcal{C}}) \cup \mathfrak{pr}(\mathcal{B}) \cup \{\mathfrak{cl}(\mathcal{B})\} \not\models \bot$, *and*

(minimality) *no $\widehat{\mathcal{C}}' \subset \widehat{\mathcal{C}}$ is a supporter of $\mathcal{B}$ through p.*

Definition 6 (Free Premise) *Given a* DAF $\langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathtt{rgs}} \times 2^{\mathbb{A}\mathtt{rgs}}$ *and an argument $\mathcal{B} \in \mathbf{U}$, a premise $p \in \mathfrak{pr}(\mathcal{B})$ of $\mathcal{B}$ is* ***free*** *wrt. $\mathbf{U}$ iff there is no supporting-coalition $\widehat{\mathcal{C}} \subseteq \mathbf{U}$ of $\mathcal{B}$ through p.*

Definitions 5 and 6 are reviewed in the following examples. Just for simplicity, we will omit universal quantifiers for free variables in a given formula $\varphi(x)$ to refer to a formula $(\forall x)(\varphi(x))$.

Example 1 *Suppose we have a set $\mathbf{U} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3\}$, where arguments $\mathcal{B}_1 = \langle \{(\exists y)(A_1(y) \wedge R(x, y)),\ A_2(x)\},\ B(x) \rangle$, $\mathcal{B}_2 = \langle \{\}, R(a, b) \rangle$, and $\mathcal{B}_3 = \langle \{\}, A_1(b) \rangle$. The set $\widehat{\mathcal{C}} = \{\mathcal{B}_2, \mathcal{B}_3\}$ is a supporting-coalition of $\mathcal{B}_1$ given that $\{R(a, b), A_1(b)\} \models (\exists y)(A_1(y) \wedge R(x, y))$. Note that premise $A_2(x)$ is free wrt. $\mathbf{U}$.*

Example 2 *Assume $\mathbf{U} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4\}$, where $\mathcal{B}_1 = \langle \{A_1(x)\},\ B_1(x) \rangle$, $\mathcal{B}_2 = \langle \{A_1(x)\},\ B_2(x) \rangle$, $\mathcal{B}_3 = \langle \{A_2(x)\}, A_1(x) \vee B_1(x) \rangle$, and $\mathcal{B}_4 = \langle \{A_3(x)\}, \neg B_1(x) \rangle$. The set $\widehat{\mathcal{C}} = \{\mathcal{B}_3, \mathcal{B}_4\}$ is a supporting-coalition of $\mathcal{B}_2$. Note that $\widehat{\mathcal{C}}$ cannot be a supporting-coalition of $\mathcal{B}_1$ since it violates (supporter) consistency.*

A premise is said to be closed wrt. $\mathbf{U}$ if there exists some supporting-coalition for it, and recursively for each new appearing free premise. That means that the "iterated" support of a given premise p does ultimately end with an empty set of premises *iff* p is closed wrt. $\mathbf{U}$. To this matter, a supporting-chain is formally defined as follows.

Definition 7 (Supporting-Chain) *Given a* $\mathtt{DAF}\ \langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, *and a sequence* $\lambda = \mathcal{B} \xleftarrow{p} \widehat{\mathcal{C}}_1 \xleftarrow{p_1} \widehat{\mathcal{C}}_2 \xleftarrow{p_2} \cdots$, *where* $(\bigcup_{i \geq 1} \widehat{\mathcal{C}}_i) \cup \mathcal{B} \subseteq \mathbf{U}$, $p \in \mathfrak{pr}(\mathcal{B})$, $\widehat{\mathcal{C}}_1$ *is a supporting-coalition of* $\mathcal{B}$ *through p, and for every* $i > 1$, $p_{i-1} \in \widehat{\mathfrak{prset}}(\widehat{\mathcal{C}}_{i-1})$, *and* $\widehat{\mathcal{C}}_i$ *is a supporting-coalition of* $\widehat{\mathcal{C}}_{i-1}$ *through* p_{i-1}. *Thus,* λ *is referred as a (possible infinite)* ***supporting-chain for*** p *of* $\mathcal{B}$ *wrt.* $\mathbf{U}$.

Whenever λ *has* $\widehat{\mathcal{C}}_n$ *as its last element, it follows that every premise in* $\widehat{\mathfrak{prset}}(\widehat{\mathcal{C}}_n)$ *is free wrt.* $\mathbf{U}$, *or* $\widehat{\mathfrak{prset}}(\widehat{\mathcal{C}}_n) = \emptyset$. *In such a case,* λ *is said to be a* ***finite supporting-chain for*** p *of length* n *wrt.* $\mathbf{U}$.

The iterated aggregation of arguments via the support relation (*c.f.* Def. 5) may conform both, chains of supporting-coalitions for a premise in some argument (*c.f.* Def. 7), as well as sets of inter-related arguments (*c.f.* Def. 8). We will refer to such sets as structures, formally defined as follows.

Definition 8 (Structure) *Given a* $\mathtt{DAF}\ \langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, $\mathbb{S} \subseteq \mathbf{U}$ *is a* ***structure for*** c *iff it guarantees:*

(top argument) *there exists a unique* $\mathcal{B}^{top} \in \mathbb{S}$ *such that* $\mathfrak{cl}(\mathcal{B}^{top}) = c$,

(connectivity) *for every* $\mathcal{B} \in \mathbb{S} \setminus \{\mathcal{B}^{top}\}$, *there exists a unique subset* $\widehat{\mathcal{C}} \subseteq \mathbb{S}$ *such that* $\mathcal{B} \in \widehat{\mathcal{C}}$ *where* $\widehat{\mathcal{C}}$ *is a supporting-coalition of an argument in* $\mathbb{S}$,

(self-consistency) $\widehat{\mathfrak{prset}}(\mathbb{S}) \cup \widehat{\mathfrak{clset}}(\mathbb{S}) \not\models \bot$, *and*

(acyclicity) *every supporting-chain for every* p *of every* $\mathcal{B} \in \mathbb{S}$ *wrt.* $\mathbb{S}$ *is finite.*

The claim and premises of $\mathbb{S}$ *are determined by the functions* $\mathfrak{cl}(\mathbb{S}) = c$ *and* $\mathfrak{pr}(\mathbb{S}) = \{p \in \widehat{\mathfrak{prset}}(\mathbb{S}) \mid p$ *is a free premise wrt.* $\mathbb{S}\}$, *respectively.*

Note that functions "$\mathfrak{pr}$" and "$\mathfrak{cl}$" are overloaded and can be applied both to arguments and structures. This is not going to be problematic since either usage will be rather explicit. In addition to that, we will identify the top argument of a structure $\mathbb{S}$ using the function $\mathfrak{top} : 2^{\mathbb{A}\mathrm{rgs}} \rightarrowtail \mathbb{A}\mathtt{rgs}$. Note that $\mathfrak{cl}(\mathfrak{top}(\mathbb{S})) = \mathfrak{cl}(\mathbb{S})$.

Example 3 *Given two arguments* $\mathcal{B}_1 = \langle \{A(x)\}, B(x) \rangle$ *and* $\mathcal{B}_2 = \langle \{B(x)\}, A(x) \rangle$. *A set* $\{\mathcal{B}_1, \mathcal{B}_2\}$ *cannot be part of any structure since the infinite supporting-chain* $\lambda = \mathcal{B}_1 \xleftarrow{A(x)} \{\mathcal{B}_2\} \xleftarrow{B(x)} \{\mathcal{B}_1\} \xleftarrow{A(x)} \cdots$ *for* $A(x)$ *would violate (structure) acyclicity.*

A structure $\mathbb{S}$ trivially formed by a single argument is referred as *primitive iff* $|\mathbb{S}| = 1$. Thus, if $\mathbb{S} = \{\mathcal{B}\}$ then $\mathfrak{pr}(\mathcal{B}) = \mathfrak{pr}(\mathbb{S})$ and $\mathfrak{cl}(\mathcal{B}) = \mathfrak{cl}(\mathbb{S})$. However, not every single argument has an associated primitive structure. For instance, no structure could contain an argument $\langle \{A(x)\}, A(x) \rangle$ given that it would violate (structure) acyclicity. Depending on the condition of the set of premises in a structure we may identify two different kinds of structures.

Definition 9 (Schematic & Argumental Structure) *Given a* $\mathtt{DAF}\ \langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, *a structure* $\mathbb{S} \subseteq \mathbf{U}$ *is referred either as:* ***Argumental*** *iff* $\mathfrak{pr}(\mathbb{S}) = \emptyset$, *or* ***Schematic*** *iff* $\mathfrak{pr}(\mathbb{S}) \neq \emptyset$.

When no distinction is needed, we refer to primitive, schematic, or argumental structures, simply as structures. A **sub-structure relation** will be defined by $\trianglelefteq \in 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$. That is, given a structure $\mathbb{S}$ for a claim c, if it contains a subset $\mathbb{S}'$ verifying the conditions in Def. 8 for a claim c', then $\mathbb{S}'$ is a structure for c' and $\mathbb{S}' \trianglelefteq \mathbb{S}$. Finally, $\mathbb{S}'$ is said a sub-structure of $\mathbb{S}$. Note that $c = c'$ *iff* $\mathbb{S} = \mathbb{S}'$.

From a schematic structure and a supporting-coalition for it, a new structure is formed. If this new structure has no free premises, it means that a *variable substitution* was made over the schematic structure leading to an argumental structure. In general, a structure that adds some evidential argument about an individual name, say a, as part of the support for a schematic structure, provokes a variable substitution in the latter. In that case, the argumental structure ends up asserting some property – through its claim– about the individual a. Finally, it is clear that if a structure states a property about some element of the world by means of a free variable x then it is schematic.

Two argumental structures $\mathbb{S}_1$ and $\mathbb{S}_2$ are in conflict whenever they cannot be assumed together. This notion may be made extensive to sets of argumental structures, namely coalition of argumental structures. Coalition of structures is analogous to that of arguments; its formalization is not given due to lack of space. Therefore, the functions "$\widehat{\mathfrak{clset}}$" and "$\widehat{\mathfrak{prset}}$" are overloaded and can be applied both to coalitions $\widehat{\mathcal{C}}$ of arguments and to coalitions $\widehat{\mathbb{C}}$ of structures. Formally, $\widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}) = \{\mathfrak{cl}(\mathbb{S}) | \mathbb{S} \in \widehat{\mathbb{C}}\}$, and $\widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}) = \bigcup_{\mathbb{S} \in \widehat{\mathbb{C}}} \mathfrak{pr}(\mathbb{S})$. Next, we specify the notion of conflict between coalitions of structures as a generalization, since one of them has to be necessarily a singleton. This is required to preserve conflict minimality.

Definition 10 (Conflict) *Let $\langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$ be a* DAF, *and $\widehat{\mathbb{C}}_1$ and $\widehat{\mathbb{C}}_2$, two* ***minimal****, and* ***consistent*** *coalitions of structures in $\mathbf{U}$ verifying:*

a) $|\widehat{\mathbb{C}}_1| = 1$, *or* $|\widehat{\mathbb{C}}_2| = 1$, *and*

b) $\widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_1) \models \widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_2)$ ***(dependency)****, or* $\widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}_1) \models \widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_2)$ ***(support)****.*

Coalitions $\widehat{\mathbb{C}}_1$ and $\widehat{\mathbb{C}}_2$ are in ***conflict*** *iff every structure $\mathbb{S} \subseteq (\widehat{\mathbb{C}}_1 \cup \widehat{\mathbb{C}}_2)$, is the* ***smallest*** *$\mathbb{S}$ needed to guarantee either:*

(claim-conflict) $\widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}_1) \cup \widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}_2) \models \bot$, *or*

(premise-conflict) $\widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}_1) \cup \widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_2)) \models \bot$.

Example 4 *Assume we have arguments $\mathcal{B}_1 = \langle \{A_1(x), A_2(x)\}, B(x) \rangle$, $\mathcal{B}_2 = \langle \{A_3(x)\}, A_1(x) \rangle$, and $\mathcal{B}_3 = \langle \{A_3(x)\}, \neg A_2(x) \rangle$, then two structures $\mathbb{S}_1 = \{\mathcal{B}_1, \mathcal{B}_2\}$ and $\mathbb{S}_2 = \{\mathcal{B}_3\}$ appear. The trivial coalitions $\widehat{\mathbb{C}}_1 = \{\mathbb{S}_1\}$ and $\widehat{\mathbb{C}}_2 = \{\mathbb{S}_2\}$ model a* ***premise-conflict****. Note that $\widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_1) = \{A_3(x), A_2(x)\}$, $\widehat{\mathfrak{prset}}(\widehat{\mathbb{C}}_2) = \{A_3(x)\}$, and $\widehat{\mathfrak{clset}}(\widehat{\mathbb{C}}_2) = \{\neg A_2(x)\}$.*

Assume now that we have arguments $\mathcal{B}_1$ and $\mathcal{B}_2$, and a different $\mathcal{B}_3' = \langle \{A_3(x)\}, \neg B(x) \rangle$. It is easy to verify that a ***claim-conflict*** *will be modeled from $\widehat{\mathbb{C}}_1$ and $\{\{\mathcal{B}_3'\}\}$.*

Note that both conflicts in Ex. 4 come from *dependency* (*c.f.* Def. 10b). An example of *claim-conflict* from *support* appears in Ex. 2. It is clear that no *premise-conflict from support* is possible since both support and premise-conflict conditions cannot be mutually verified.

Deciding which coalition of structures succeeds between a conflicting pair, requires a comparison criterion. Such a criterion could be defined for instance, upon entrenchment of knowledge, *i.e.*, that the knowledge engineer may give different levels of importance to individual pieces of knowledge. In that sense, we will assume there exists a partial order of arguments called *argument comparison criterion* "$\succcurlyeq$", such that $\mathcal{B}_1 \succcurlyeq \mathcal{B}_2$ states that $\mathcal{B}_1$ has more priority than $\mathcal{B}_2$. Afterwards, two conflictive coalitions of structures $\widehat{\mathbb{C}}_1$ and $\widehat{\mathbb{C}}_2$ are assumed to be ordered by a function "$\mathfrak{pref}$" relying on "$\succcurlyeq$", where $\mathfrak{pref}(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) = (\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2)$ implies the attack relation $\widehat{\mathbb{C}}_1\mathbf{R}\widehat{\mathbb{C}}_2$, *i.e.*, $\widehat{\mathbb{C}}_1$ is a defeater of (or it defeats) $\widehat{\mathbb{C}}_2$. Note that when no pair of arguments is related by "$\succcurlyeq$", both $\widehat{\mathbb{C}}_1\mathbf{R}\widehat{\mathbb{C}}_2$ and $\widehat{\mathbb{C}}_2\mathbf{R}\widehat{\mathbb{C}}_1$ appear from any conflicting pair $\widehat{\mathbb{C}}_1$ and $\widehat{\mathbb{C}}_2$.

Definition 11 (Attack Relation Set) *The set* $\mathbf{R}$ *of attack relations is defined as* $\mathbf{R} = \{(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) \mid \widehat{\mathbb{C}}_1$ *and* $\widehat{\mathbb{C}}_2$ *are in conflict and* $\mathfrak{pref}(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) = (\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2)\}$.

Regarding the active condition of the components of the framework, a structure is active *iff* all its arguments are active. This notion is also extended to coalitions of structures by considering a coalition $\widehat{\mathbb{C}}$ active *iff* all its structures are active. Finally, an attack relation $\widehat{\mathbb{C}}_1\mathbf{R}\widehat{\mathbb{C}}_2$ is active *iff* both $\widehat{\mathbb{C}}_1$ and $\widehat{\mathbb{C}}_2$ are active. That is, if $(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) \in \mathbf{R}^{\mathbf{A}} \subseteq \mathbf{R}$ then $\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2 \subseteq \mathbf{A}$, where $\mathbf{R}^{\mathbf{A}}$ is the set standing for every active attack relation in $\mathbf{R}$.

Example 5 *Assume the* DAF $\langle \mathbf{U}, \mathbf{A} \rangle$ *is determined as* $\mathbf{U} = \mathbf{A} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5, \mathcal{B}_6, \mathcal{B}_7\}$, *where* $\mathcal{B}_1 = \langle \{\}, R(a,b) \rangle$, $\mathcal{B}_2 = \langle \{\}, R(b,c) \rangle$, $\mathcal{B}_3 = \langle \{\}, R(c,d) \rangle$, $\mathcal{B}_4 = \langle \{\}, A(a) \rangle$, $\mathcal{B}_5 = \langle \{\}, \neg A(c) \rangle$, $\mathcal{B}_6 = \langle \{\}, \neg A(d) \rangle$, $\mathcal{B}_7 = \langle \{A(x)\}, (\forall y)(R(x,y) \rightarrow A(y)) \rangle$.

The argumental structure $\mathbb{S}_1 = \{\mathcal{B}_4, \mathcal{B}_7\}$ *appears. Later on, the set* $\widehat{\mathcal{C}}_1 = \{\mathcal{B}_7, \mathcal{B}_1\}$ *is a supporting-coalition of* $\mathcal{B}_7$ *through* $A(b)$, $\widehat{\mathcal{C}}_2 = \{\mathcal{B}_7, \mathcal{B}_2\}$ *is a supporting-coalition of* $\mathcal{B}_7$ *through* $A(c)$, *and* $\widehat{\mathcal{C}}_3 = \{\mathcal{B}_7, \mathcal{B}_3\}$ *is a supporting-coalition of* $\mathcal{B}_7$ *through* $A(d)$. *Hence, the schematic structures* $\mathbb{S}_2 = \{\mathcal{B}_7, \mathcal{B}_1\}$, $\mathbb{S}_3 = \{\mathcal{B}_7, \mathcal{B}_2\}$, *and* $\mathbb{S}_4 = \{\mathcal{B}_7, \mathcal{B}_3\}$, *appear with* $\mathfrak{cl}(\mathbb{S}_2) = R(a,b) \rightarrow A(b)$, $\mathfrak{cl}(\mathbb{S}_3) = R(b,c) \rightarrow A(c)$, *and* $\mathfrak{cl}(\mathbb{S}_4) = R(c,d) \rightarrow A(d)$; *and premises* $\mathfrak{pr}(\mathbb{S}_2) = A(a)$, $\mathfrak{pr}(\mathbb{S}_3) = A(b)$, *and* $\mathfrak{pr}(\mathbb{S}_4) = A(c)$. *Thus, appear the related argumental structures* $\mathbb{S}_5 = \{\mathcal{B}_4, \mathcal{B}_7, \mathcal{B}_1\}$, $\mathbb{S}_6 = \{\mathcal{B}_4, \mathcal{B}_7, \mathcal{B}_1, \mathcal{B}_2\}$, *and* $\mathbb{S}_7 = \{\mathcal{B}_4, \mathcal{B}_7, \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3\}$, *where* $\mathbb{S}_2 \trianglelefteq \mathbb{S}_5$, $\mathbb{S}_3 \trianglelefteq \mathbb{S}_6$, *and* $\mathbb{S}_4 \trianglelefteq \mathbb{S}_7$, *as well as* $\mathbb{S}_5 \trianglelefteq \mathbb{S}_6$, *and* $\mathbb{S}_6 \trianglelefteq \mathbb{S}_7$. *Note also that,* $\mathbb{S}_1$ *is sub-structure of* $\mathbb{S}_5$, $\mathbb{S}_6$, *and* $\mathbb{S}_7$. *Besides, from* $\mathbb{S}_6$, *the supporting-chain for* $A(x)$ *in* $\mathcal{X}$ *is* $\mathcal{X} \overset{A(c)}{\longleftarrow} \{\mathcal{X}, \mathcal{B}_2\} \overset{A(b)}{\longleftarrow} \{\mathcal{X}, \mathcal{B}_1\} \overset{A(a)}{\longleftarrow} \{\mathcal{B}_4\}$.

Consider now the coalitions of structures $\widehat{\mathbb{C}}_1 = \{\{\mathcal{B}_2\}, \{\mathcal{B}_5\}\}$, *and* $\widehat{\mathbb{C}}_2 = \{\{\mathcal{B}_3\}, \{\mathcal{B}_6\}\}$. *Assuming* $\mathcal{B}_7 \succcurlyeq \mathcal{B}_i$, $i \in \{1, \ldots, 6\}$, *the following attack relations appear:* $\{\mathbb{S}_5\}\mathbf{R}\widehat{\mathbb{C}}_1$ *and* $\{\mathbb{S}_6\}\mathbf{R}\widehat{\mathbb{C}}_2$ *(refer to Fig. 1). Later on, considering also the coalitions of structures* $\widehat{\mathbb{C}}_3 = \{\mathbb{S}_5, \{\mathcal{B}_2\}\}$, *and* $\widehat{\mathbb{C}}_4 = \{\mathbb{S}_6, \{\mathcal{B}_3\}\}$, *attacks* $\widehat{\mathbb{C}}_3\mathbf{R}\{\{\mathcal{B}_5\}\}$ *and* $\widehat{\mathbb{C}}_4\mathbf{R}\{\{\mathcal{B}_6\}\}$ *appear.*

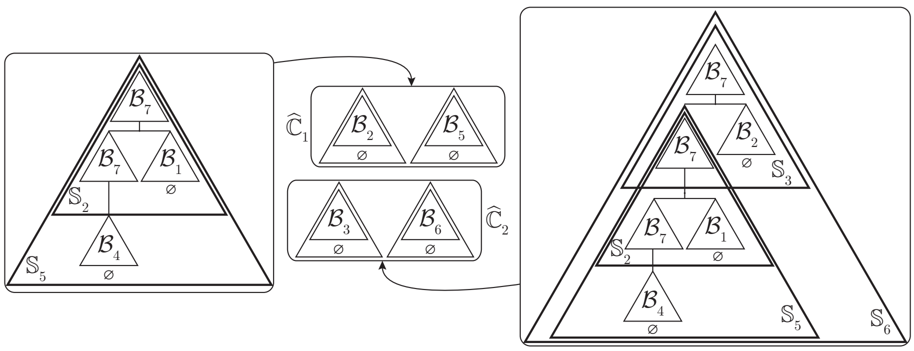


Figure 1: Some attacks from Ex. 5. Multiple occurrences of an argument within a structure refer to its different instances determined by every possible variable substitution.

3 Ontology Debugging Through the DAF

In what follows we propose a reification of the generalized DAF to the description logic $\mathcal{ALC}$. Afterwards, since some basic elements from argumentation like attack and support may allow to manage inconsistencies in ontologies, we propose an acceptability semantics for arguments in order to obtain a related maximal consistent ontology.

3.1 Reifying the Generalized DAF to $\mathcal{ALC}$ DL

Before presenting the reification subtleties, a very brief overview of the $\mathcal{ALC}$ DL will be given, for more detailed information refer to [2]. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a nonempty domain $\Delta^{\mathcal{I}}$, and an interpretation function $\cdot^{\mathcal{I}}$ that maps every concept to a subset of $\Delta^{\mathcal{I}}$, every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and every individual to an element of $\Delta^{\mathcal{I}}$.

Symbols $A, A_1, A_2, \ldots$ and $B, B_1, B_2, \ldots$ are used to denote atomic DL concepts, $C, C_1, C_2, \ldots$ and $D, D_1, D_2, \ldots$, to denote general DL concepts, and $R, R_1, R_2, \ldots$, to denote atomic DL roles. The description language $\mathcal{ALC}$ is formed by concept definitions according to the syntax $C, D ::= A|\bot|\top|\neg C|C \sqcap D|C \sqcup D|\forall R.C|\exists R.C$ where the interpretation function $\cdot^{\mathcal{I}}$ is extended to the universal concept as $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$; the bottom concept as $\bot^{\mathcal{I}} = \emptyset$; the full negation or complement as $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$; the intersection as $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$; the union as $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$; the universal quantification as $(\forall R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} | \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$; and the full existential quantification as $(\exists R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} | \exists b.(a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$.

An ontology is a pair $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T}$ represents the TBox, containing the terminologies (or axioms) of the application domain, and $\mathcal{A}$, the ABox, which contains assertions about named individuals in terms of these terminologies. Regarding the TBox $\mathcal{T}$, axioms are sketched as $C \sqsubseteq D$ and $C \equiv D$, therefore, an interpretation $\mathcal{I}$ satisfies them whenever $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ and $C^{\mathcal{I}} = D^{\mathcal{I}}$ respectively. An interpretation $\mathcal{I}$ is a model for the TBox $\mathcal{T}$ if $\mathcal{I}$ satisfies all the axioms in $\mathcal{T}$. Thus, the TBox $\mathcal{T}$ is said to be satisfiable if it admits a model. Besides, in the ABox $\mathcal{A}$, $\mathcal{I}$ satisfies $C(a)$ if $a \in C^{\mathcal{I}}$, and $R(a,b)$ if $(a,b) \in R^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is said to be a model of the ABox $\mathcal{A}$ if every assertion of $\mathcal{A}$ is satisfied by $\mathcal{I}$. Hence, the ABox $\mathcal{A}$ is said to be satisfiable if it admits a model. Finally, regarding the entire ontology, an interpretation $\mathcal{I}$ is said to be a model of $\mathcal{O}$ if every statement in $\mathcal{O}$ is satisfied by $\mathcal{I}$, and $\mathcal{O}$ is said to be satisfiable if it admits a model.

Moreover, the different classes of inconsistencies in an ontology may be characterized as follows. Given an ontology $\mathcal{O}$, a concept C is *unsatisfiable iff* for each interpretation $\mathcal{I} \in \mathcal{M}(\mathcal{O})$, $C^{\mathcal{I}} = \emptyset$. As stated in [6], an ontology $\mathcal{O}$ is *incoherent iff* there exists an unsatisfiable concept in $\mathcal{O}$. An incoherence may be considered a kind of inconsistency in the TBox. However, the incoherence does not replace the classical meaning of inconsistency, given that an incoherent ontology may admit models. Hence, an ontology $\mathcal{O}$ is *inconsistent iff* it admits no model.

An ontology contains implicit knowledge that is made explicit through inferences. The notion of semantic entailment is given by $\mathcal{O} \models \alpha$, meaning that every model of the ontology $\mathcal{O}$ is also a model of the statement α. Formally, **(semantic entailment)** $\mathcal{O} \models \alpha$ *iff* $\mathcal{M}(\mathcal{O}) \subseteq \mathcal{M}(\{\alpha\})$. Just for simplicity, we shall abuse notation writing $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$ (*eg.*, $\mathcal{O} = \{C \sqsubseteq D, A(a)\}$) to identify an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ (*eg.*, $\mathcal{O} = \langle \{C \sqsubseteq D\}, \{A(a)\} \rangle$).

The following grammars are proposed in order to specify the argument language used to represent $\mathcal{ALC}$-based ontologies into a dynamic argumentation framework (DAF) for DLs.

$$\phi ::= \top|A|\neg A|\forall R.\mathcal{L}_{disj}|\exists R.\mathcal{L}_{conj}$$

$$\mathcal{L}_{conj} ::= \phi|\mathcal{L}_{conj} \sqcap \mathcal{L}_{conj} \qquad \mathcal{L}_{disj} ::= \phi|\mathcal{L}_{disj} \sqcup \mathcal{L}_{disj}$$

$$\mathcal{L}_{\mathtt{pr}} ::= \phi(\mathcal{L}_{var}) \qquad \mathcal{L}_{\mathtt{cl}} ::= \mathcal{L}_{disj}(\mathcal{L}_{var})|R(\mathcal{L}_{var}, \mathcal{L}_{var})$$

$$\mathcal{L}_{var} ::= a|b|x|y \qquad \mathbb{A}\mathtt{rgs} ::= 2^{\mathcal{L}_{\mathtt{pr}}} \times \mathcal{L}_{\mathtt{cl}}$$

In order to obtain a DAF from an $\mathcal{ALC}$ ontology $\mathcal{O}$, it is needed to translate each axiom in $\mathcal{O}$ to *negation normal form*, so that negation appears only in front of atomic concepts. Afterwards, each axiom should turn to *disjunctive normal form* for the left-hand-side (*lhs*) part of the description, and to *conjunctive normal form* for its right-hand-side (*rhs*), conforming axioms *lhs* $\sqsubseteq$ *rhs* or *lhs* $\equiv$ *rhs*, where *lhs* $::= \bot|\mathcal{L}_{conj} \sqcup \ldots \sqcup \mathcal{L}_{conj}$ and *rhs* $::= \bot|\mathcal{L}_{disj} \sqcap \ldots \sqcap \mathcal{L}_{disj}$, referred as *pre-argumental normal form* (pANF). An ontology in pANF could trigger multiple arguments from each axiom, as states the following intuition: each *lhs* disjunction (in $\mathcal{L}_{conj}$) is interpreted as a set of premises $\mathcal{L}_{\mathfrak{pr}}$– one for each conjunction– and each *rhs* conjunction (in $\mathcal{L}_{disj}$), as a claim in $\mathcal{L}_{\mathfrak{cl}}$ (*c.f.* Ex. 6). Concept equivalences as $C_1 \equiv C_2$, are assumed as pairs $C_1 \sqsubseteq C_2$ and $C_2 \sqsubseteq C_1$. Inclusions $\bot \sqsubseteq C$ and $C \sqsubseteq \bot$, are assumed as $\neg C \sqsubseteq \top$ and $\top \sqsubseteq \neg C$, respectively, given that arguments cannot accept $\bot$ in any of their components (*c.f. consistency* in Def. 2). Finally, any assertion $A(a)$ triggers an evidence $\langle\{\}, A(a)\rangle$. A formal specification of a systematic translation was left out due to space reasons.

Example 6 *Let* $(A_1 \sqcap A_2) \sqcup (\forall R_1.A_3 \sqcap \exists R_2.\forall R_3.\neg A_4) \sqsubseteq (A_1 \sqcup A_2) \sqcap A_5$ *be an axiom conforming the* pANF. *Four arguments appear in the related* DAF*:* $\langle\{A_1(x), A_2(x)\}, (A_1 \sqcup A_2)(x)\rangle$, $\langle\{(\forall R_1.A_3)(x), (\exists R_2.\forall R_3.\neg A_4)(x)\}, (A_1 \sqcup A_2)(x)\rangle$, $\langle\{A_1(x), A_2(x)\}, A_5(x)\rangle$, *and* $\langle\{(\forall R_1.A_3)(x), (\exists R_2.\forall R_3.\neg A_4)(x)\}, A_5(x)\rangle$.

Given an $\mathcal{ALC}$ ontology $\mathcal{O}$, a function $\mathfrak{daf} : \mathcal{ALC} \rightarrowtail 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$ is the mapping $\mathfrak{daf}(\mathcal{O}) = \langle \mathbf{U}, \mathbf{A}\rangle$ following the translation methodology described before. That is, $\mathcal{O}$ is turned into pANF, and thus the DAF is obtained where every argument identified appears active, *i.e.*, $\mathbf{A} = \mathbf{U}$. Furthermore, we define as $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ to the logic for ontologies $\mathcal{O} \subseteq \mathcal{L}_{\mathcal{T}} \times \mathcal{L}_{\mathcal{A}}$, using $\mathcal{L}_{\mathcal{T}} ::= \mathcal{L}_{conj} \sqsubseteq \mathcal{L}_{disj}$ for axioms and $\mathcal{L}_{\mathcal{A}} ::= A(\mathcal{L}_{var})|\neg A(\mathcal{L}_{var})|R(\mathcal{L}_{var}, \mathcal{L}_{var})$ for assertions. It is clear that any $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ ontology conforms the $\mathcal{ALC}$ DL, and it is always in pANF. Moreover, we will assume a function $\mathfrak{af} : \mathcal{ALC} \rightarrowtail \mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$, the *argumental-DL function* that translates any $\mathcal{ALC}$ ontology $\mathcal{O}$ into an equivalent $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ ontology $\mathfrak{af}(\mathcal{O})$. A desirable property of an $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ ontology is that each statement in it generates a single argument in its related DAF, except for obvious unsatisfiable inclusions as $A \sqsubseteq \neg A$, which are filtered by *consistency* in Def. 2 –triggering no related argument in the DAF.

Proposition 1 [1] *Let* $\mathcal{O}$ *and* $\mathcal{O}'$ *be two ontologies. If* $\mathcal{O}$ *conforms the logic* $\mathcal{ALC}$*, and* $\mathcal{O}'$ *conforms* $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ *then*

$a)$ $\mathcal{O}'$ *conforms the logic* $\mathcal{ALC}$ *and is in* pANF*,*

$b)$ *If* $\mathfrak{af}(\mathcal{O}) = \mathcal{O}'$ *then* $\mathcal{O}$ *is equivalent to* $\mathcal{O}'$*, and*

$c)$ *If* $\mathcal{O}$ *conforms the logic* $\mathcal{ALC}^{\mathbb{A}\mathrm{rgs}}$ *then* $|\mathcal{O}| \geq |\mathbf{U}|$ *where* $\mathfrak{daf}(\mathcal{O}) = \langle \mathbf{U}, \mathbf{A}\rangle$.

Notice that, from an ontology $\mathcal{O} = \{A \equiv B\}$ a situation like that in Ex. 3 occurs, where (structure) acyclicity manages to handle a cyclic terminology. A similar case occurs with an axiom as $A \sqsubseteq A$.

3.2 An Argumentation Semantics as Debugging Tool

In an ontology, inconsistency implies that there are contradictory concept definitions, or assertions that will lead to conflicting arguments within the equivalent DAF. Thus, once the translation is performed, each inconsistency in the original ontology will be reflected as an attack in the resulting DAF. Since the objective of converting an ontology to an argumentation framework is to remove inconsistency from the former, there is a need for a mechanism that allows us to obtain those arguments that

[1] In this work, proofs will be omitted due to space reasons.

prevail over the rest. That is, those arguments that can be consistently assumed together, following some policy. For instance, structures with no defeaters should always prevail, since there is nothing strong enough to be posed against them. The tool we need to resolve inconsistency of concept definitions via an argumentation framework is the notion of *acceptability of arguments*, which is defined on top of an *argumentation semantics* [3]. There are several well-known argumentation semantics, such as the grounded, the stable, and the preferred semantics [5]. These semantics ensure the obtention of a consistent set of arguments, namely an extension. That is, the set of accepted arguments calculated following any of these semantics is such that no pair of conflictive arguments appears in that same extension. Finally, when we translate an ontology to a DAF, all what is left to do to resolve inconsistencies is to calculate the set of accepted arguments following some semantics, which is going to be translated back to a consistent ontology. It is important to notice that the chosen semantics will greatly affect the resulting ontology. Moreover, problems like multiple extensions from semantics like both the *stable* and the *preferred* may appear, requiring to make a choice among them. On the other hand, the outcome of the *grounded semantics* is always a single extension, which could be empty. Finally, since dealing with multiple extensions is a problem that falls outside the scope of this article, we will choose the grounded semantics, which can be implemented with a simple algorithm. Consequently, we define a mapping $\mathfrak{sem} : 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}} \rightarrowtail 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, that intuitively behaves as follows.

For every pair of active attack $(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) \in \mathbf{R}^{\mathbf{A}}$, if there is no active coalition of structures defeating $\widehat{\mathbb{C}}_1$ **(undefeat)**, then we deactivate some argument from some structure in $\widehat{\mathbb{C}}_2$ **(deactivation)**. As a side-effect, any attack $(\widehat{\mathbb{C}}_2, \widehat{\mathbb{C}}_3)$ will disappear. This process is recursively applied on $\mathbf{R}^{\mathbf{A}}$ until every attack relation is deactivated. As stated before, the outcome of a grounded semantics could be an empty extension. Such an issue arises when there is a loop in the structures attack graph. To overcome this, if *undefeat* is not verified for any $(\widehat{\mathbb{C}}_1, \widehat{\mathbb{C}}_2) \in \mathbf{R}^{\mathbf{A}}$, then *deactivation* is applied to some active attack. Thus, the loop is broken, and the process determined by applying "$\mathfrak{sem}$" can be reconsidered.

Proposition 2 *Given a* DAF $T \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, *if* $\mathfrak{sem}(T) = \langle \mathbf{U}, \mathbf{A} \rangle$ *then* $\mathbf{R}^{\mathbf{A}} = \emptyset$.

Let "$\mathfrak{ont}$" be a mapping from a DAF $\langle \mathbf{U}, \mathbf{A} \rangle \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$ to an $\mathcal{ALC}$ ontology $\mathfrak{ont}(\langle \mathbf{U}, \mathbf{A} \rangle)$, following backwards the intuitions given to obtain a DAF by "$\mathfrak{daf}$". Consistency-coherency of the $\mathfrak{ont}$-outcome is related to the attacks in the DAF by Prop. 3.

Proposition 3 *Given a* DAF $\langle \mathbf{U}, \mathbf{A} \rangle$ *where* $\mathbf{A} \subseteq \mathbf{U} \subseteq 2^{\mathbb{A}\mathrm{rgs}}$, $\mathbf{R}^{\mathbf{A}} = \emptyset$ *iff* $\mathfrak{ont}(\langle \mathbf{U}, \mathbf{A} \rangle)$ *is a consistent-coherent* $\mathcal{ALC}$ *ontology.*

Lemma 1 *Given a* DAF $T \subseteq 2^{\mathbb{A}\mathrm{rgs}} \times 2^{\mathbb{A}\mathrm{rgs}}$, $\mathfrak{ont}(\mathfrak{sem}(T))$ *is a consistent-coherent* $\mathcal{ALC}$ *ontology.*

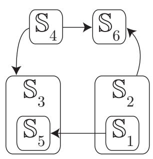
The relation stated in Prop. 3 along with that in Prop. 2 motivates Lemma 1. Theorem 1 states the main contribution of the $\mathcal{ALC}$-Based DAF regarding ontology debugging. Afterwards, Corollary 1 relates that result through "$\mathfrak{af}$". Finally, in examples 7 and 8, the methodology here proposed for ontology debugging is applied.

Theorem 1 *Given an* $\mathcal{ALC}$ *ontology* $\mathcal{O}$, *if* $\mathcal{O}$ *is inconsistent and\or incoherent then* $\mathfrak{ont}(\mathfrak{sem}(\mathfrak{daf}(\mathcal{O})))$ *is a related consistent-coherent* $\mathcal{ALC}$ *ontology.*

Corollary 1 *Given an inconsistent-incoherent* $\mathcal{ALC}$ *ontology* $\mathcal{O}$, *there exists a related consistent-coherent ontology* $\mathcal{O}'$, *such that* $\mathfrak{af}(\mathcal{O}') \subseteq \mathfrak{af}(\mathcal{O})$.

Example 7 *Let* $\mathcal{O} = \{A_1 \sqsubseteq B_1 \sqcap B_2,\ A_2 \sqsubseteq A_1 \sqcap \neg B_2,\ A_1(a),\ B_1(a),\ \neg B_2(a),\ A_2(a)\}$ *be an* $\mathcal{ALC}$ *ontology, we want to debug* $\mathcal{O}$ *to obtain a related consistent-coherent ontology* $\mathcal{O}^R$. *Applying* $\mathfrak{daf}(\mathcal{O})$, *a* DAF $\langle \mathbf{U}, \mathbf{A} \rangle$, *where* $\mathbf{U} = \mathbf{A}$ *appears:*

Statement	*Args.*
$A_1 \sqsubseteq B_1 \sqcap B_2$	$\{\mathcal{B}_1, \mathcal{B}_2\}$
$A_2 \sqsubseteq A_1 \sqcap \neg B_2$	$\{\mathcal{B}_3, \mathcal{B}_4\}$
$A_1(a)$	$\{\mathcal{B}_5\}$
$B_1(a)$	$\{\mathcal{B}_6\}$
$\neg B_2(a)$	$\{\mathcal{B}_7\}$
$A_2(a)$	$\{\mathcal{B}_8\}$



$$\left\{\begin{array}{l} \mathcal{B}_1 = \langle\{A_1(x)\}, B_1(x)\rangle \\ \mathcal{B}_2 = \langle\{A_1(x)\}, B_2(x)\rangle \\ \mathcal{B}_3 = \langle\{A_2(x)\}, A_1(x)\rangle \\ \mathcal{B}_4 = \langle\{A_2(x)\}, \neg B_2(x)\rangle \\ \mathcal{B}_5 = \langle\{\}, A_1(a)\rangle \\ \mathcal{B}_6 = \langle\{\}, B_1(a)\rangle \\ \mathcal{B}_7 = \langle\{\}, \neg B_2(a)\rangle \\ \mathcal{B}_8 = \langle\{\}, A_2(a)\rangle \end{array}\right\}$$

Consider the structures $\mathbb{S}_1 = \{\mathcal{B}_3, \mathcal{B}_2\}$, $\mathbb{S}_2 = \{\mathcal{B}_8\} \cup \mathbb{S}_1$, $\mathbb{S}_3 = \{\mathcal{B}_8, \mathcal{B}_4\}$, *and* $\mathbb{S}_4 = \{\mathcal{B}_5, \mathcal{B}_2\}$*; and the primitive structures* $\mathbb{S}_5 = \{\mathcal{B}_4\}$ *and* $\mathbb{S}_6 = \{\mathcal{B}_7\}$*.Assuming* $\mathcal{B}_2 \succcurlyeq \mathcal{B}_4$ *and* $\mathcal{B}_2 \succcurlyeq \mathcal{B}_7$*, the attack relation set is* $\mathbf{R} = \{(\{\mathbb{S}_1\}, \{\mathbb{S}_5\}), (\{\mathbb{S}_2\}, \{\mathbb{S}_6\}), (\{\mathbb{S}_4\}, \{\mathbb{S}_6\}), (\{\mathbb{S}_4\}, \{\mathbb{S}_3\})\}$ *(see the graph depicted above). Note that* $(\{\mathbb{S}_2\}, \{\mathbb{S}_3\})$ *is not in* $\mathbf{R}$ *given that* $\mathbb{S}_1 \trianglelefteq \mathbb{S}_2$, $\mathbb{S}_5 \trianglelefteq \mathbb{S}_3$*, and* $(\{\mathbb{S}_1\}, \{\mathbb{S}_5\}) \in \mathbf{R}$ *(c.f. Def. 10).*

The acceptability analysis determines $\mathbb{S}_3$, $\mathbb{S}_5$*, and* $\mathbb{S}_6$ *to be deactivated, and since* $\mathbb{S}_5 \trianglelefteq \mathbb{S}_3$*, deactivating* $\mathcal{B}_4$ *and* $\mathcal{B}_7$ *is enough. Afterwards,* $\mathfrak{sem}(\mathfrak{daf}(\mathcal{O}))$ *determines the new set of active arguments* $\{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_5, \mathcal{B}_6, \mathcal{B}_8\}$*. Finally, following the table above, the operation* $\mathfrak{ont}(\mathfrak{sem}(\mathfrak{daf}(\mathcal{O})))$ *constructs the repaired ontology* $\mathcal{O}^R = \{A_1 \sqsubseteq B_1 \sqcap B_2,\ A_2 \sqsubseteq A_1,\ A_1(a),\ B_1(a),\ A_2(a)\}$*.*

Note that, assuming $\mathcal{B}_7 \succcurlyeq \mathcal{B}_2 \succcurlyeq \mathcal{B}_4$*, conflicts involving* $\mathbb{S}_6$ *are inverted leading to* $(\{\mathbb{S}_6\}, \{\mathbb{S}_2\})$ *and* $(\{\mathbb{S}_6\}, \{\mathbb{S}_4\})$*. In such a case, only* $\mathcal{B}_2$ *would be deactivated.*

Example 8 (Ex. 5 cont.) *Assuming an ontology* $\mathcal{O} = \{R(a,b), R(b,c), R(c,d), A(a), \neg A(c), \neg A(d), A \sqsubseteq \forall R.A\}$*, and applying* $\mathfrak{daf}(\mathcal{O})$*, the* `DAF` $\langle \mathbf{U}, \mathbf{A} \rangle$ *is determined as stated in Ex. 5, changing argument* $\mathcal{B}_7$ *to* $\langle\{A(x)\}, (\forall R.A)(x)\rangle$*. The rest of the example coincides. Consequently, the acceptability analysis determines coalitions* $\widehat{\mathbb{C}}_1$, $\widehat{\mathbb{C}}_2$, $\{\{\mathcal{B}_5\}\}$*, and* $\{\{\mathcal{B}_6\}\}$ *to deactivate. Later on, the deactivation of* $\mathcal{B}_5$ *and* $\mathcal{B}_6$ *deactivates every attack. Afterwards,* $\mathfrak{sem}(\mathfrak{daf}(\mathcal{O}))$ *determines the set of active arguments as* $\{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_7\}$*. Finally, the operation* $\mathfrak{ont}(\mathfrak{sem}(\mathfrak{daf}(\mathcal{O})))$ *constructs the debugged ontology* $\mathcal{O}^R = \{A \sqsubseteq \forall R.A,\ R(a,b),\ R(b,c),\ R(c,d),\ A(a)\}$*.*

Assuming also $\mathcal{B}_5 \succcurlyeq \mathcal{B}_2$ *and* $\mathcal{B}_6 \succcurlyeq \mathcal{B}_6$*,* $\{\{\mathcal{B}_5\}\}\mathbf{R}\widehat{\mathbb{C}}_3$ *and* $\{\{\mathcal{B}_6\}\}\mathbf{R}\widehat{\mathbb{C}}_4$ *appear along with the attacks from Fig. 1. Hence, only* $\mathcal{B}_2$ *and* $\mathcal{B}_3$ *would be deactivated.*

4 Related and Future Work

Debugging of terminologies is usually focused on the recognition of sources of concept-unsatisfiability. In this sense, the union of conflictive coalitions of structures presented in this work, may be related to constructions like *minimal inconsistent preserving sub-terminologies* (MIPS) [19], which have been previously used in ontology debugging [18] and change [13]. MIPS may be also related to works in ontology integration [9], and debugging like [10], where *maximally concept-satisfiable subsets* (MCSS) were proposed for that matter.

It is interesting to extend this proposal beyond the scope of ontology debugging to that of the entire ontology change. For instance, ontology evolution could benefit from this approach. To this matter, $\bot$-Kernel Sets [7] (minimal sets inferring $\bot$), may be also related to the union of conflictive coalitions of structures. Moreover, in works like [15, 13, 11], incision functions are used to cut the appropriate piece of knowledge from every kernel such that every source of inconsistency would disappear. In that sense, the function "$\mathfrak{sem}$" deactivates the appropriate argument from each attack in order to deactivate every possible argument conflict from the `DAF`, just like incision functions do.

Further implementations of the model here presented could be done (1) as a module to be incorporated to the DL reasoner, or (2) as a DL-argumentation reasoner. For the second option, a DL reasoner based on argumentation could be an interesting alternative to those like RACER, FaCT, and FaCT++. Although a negotiation based approach was proposed in [14], to our knowledge there is no approach of reasoning about DLs based directly on argumentation. Such an approach would cope "on the fly" with the decision of what to keep or discard from different sources of information without applying any changes to them. Moreover, an ontology may keep inconsistencies leaving its resolution up to the argumentation reasoning process, that is, the ontology reasoning machinery would manage to dynamically handle inconsistency. This exposes an interesting proposal to incorporate to the semantic web the most characteristic feature of argumentation reasoners: to keep inconsistency while managing to reason on top of it.

As mentioned before, the grounded semantics [5] could return empty extensions. For instance, refer to Ex. 7 assuming an empty comparison criterion "$\succcurlyeq$". Thus, the usage of different semantics [3] could be studied to overcome this issue. Future work also involves a deep investigation on the applicability of the generalized DAF wrt. higher expressive fragments of FOL.

5 Conclusion

The proposal of a generalized DAF appears interesting to keep the advantages of usual DAFs [17, 16], along with the facility of abstracting away from the formal specification of the logic used to represent knowledge in arguments. In this sense, the notion of coalitions was introduced to characterize sets of arguments supporting a premise. A coalition of structures was also formalized in order to recognize some set of structures that in conjunction introduce a conflict wrt. another argument in the DAF. Such notions allow to generalize classic argumentation elements [5] like attack and support.

Besides (ground) arguments from usual DAFs, schematic arguments are also presented. This is a necessary extension to provide an analogy to basic FOL elements like polyadic predicates, which may consider several parameters. Recall that structures in usual DAFs are equivalent to classical arguments in an abstract framework like that of Dung [5]. In this proposal, since schematic structures are proposed, the notion of structures is slightly modified by allowing them to exist despite they keep free premises. Therefore, only argumental structures are comparable to classical arguments, given that both share the same property: the claim is reached from a satisfied set of premises.

Different classes of attack were also proposed, and in particular, the attack of schematic structures, which recognize a conflict in advance. That is, conflicts are identified wrt. the smallest possible structures, irrespective of any bigger argumental structure.

Dynamic argumentation frameworks add its most important feature to classic argumentation frameworks: dynamism. In this sense, a DAF may be defined specially to cope with specific logics for arguments, but as a consequence of the dynamism it handles, a tool to deal with change in the framework is provided. In this sense, a reification of the generalized DAF to the $\mathcal{ALC}$ DL, renders an interesting methodology to handle ontology change. As a preliminary result, we proposed a novel theoretical approach to cope with ontology debugging through argumentation, although it seems to be useful to other subareas of ontology change like ontology evolution, as well as integration.

References

[1] F. Baader. Logic-Based Knowledge Representation. In *Artificial Intelligence Today*, pages 13–41. 1999.

[2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *Description Logic Handbook: Theory, Implementation and Application*. Cambridge University Press, Cambridge, 2003.

[3] P. Baroni and M. Giacomin. On Principle-Based Evaluation of Extension-Based Argumentation Semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.

[4] A. Borgida. On the relative expressiveness of description logics and predicate logics. *Artif. Intell.*, 82(1-2):353–367, 1996.

[5] P. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning and Logic Programming and n-person Games. *Artificial Intelligence*, 77:321–357, 1995.

[6] G. Flouris, Z. Huang, J. Z. Pan, D. Plexousakis, and H. Wache. Inconsistencies, Negations and Changes in Ontologies. In *AAAI*, pages 1295–1300, 2006.

[7] S. O. Hansson. Kernel Contraction. *Journal of Symbolic Logic*, 59:845–859, 1994.

[8] C. Lutz, U. Sattler, and F. Wolter. Description Logics and the Two-Variable Fragment. In *Description Logics*, 2001.

[9] T. Meyer, K. Lee, and R. Booth. Knowledge Integration for Description Logics. In *AAAI*, pages 645–650, 2005.

[10] T. Meyer, K. Lee, R. Booth, and J. Z. Pan. Finding Maximally Satisfiable Terminologies for the Description Logic ALC. In *AAAI*, 2006.

[11] M. Moguillansky, M. Falappa, and G. Simari. Model-Based Contractions for Description Logics. In *NMR*, 2008 (to appear).

[12] M. Mortimer. On Languages with Two Variables. *Zeitschr. f. math. Logik und Grundlagen d. Math.*, 21:135–140, 1975.

[13] G. Qi, P. Haase, Z. Huang, and J. Z. Pan. A Kernel Revision Operator for Terminologies. In *DL*, 2008.

[14] G. Qi, W. Liu, and D. A. Bell. Combining multiple prioritized knowledge bases by negotiation. *Fuzzy Sets and Systems*, 158(23):2535–2551, 2007.

[15] M. M. Ribeiro and R. Wassermann. Base Revision in Description Logics - prelimimnary results. In *IWOD*, 2007.

[16] N. Rotstein, M. Moguillansky, M. Falappa, A. García, and G. Simari. Argument Theory Change: Revision Upon Warrant. In *COMMA*, pages 336–347, 2008.

[17] N. Rotstein, M. Moguillansky, A. García, and G. Simari. An Abstract Argumentation Framework for Handling Dynamics. In *NMR*, 2008 (to appear).

[18] S. Schlobach. Debugging and Semantic Clarification by Pinpointing. In *ESWC*, pages 226–240, 2005.

[19] S. Schlobach and R. Cornet. Non-Standard Reasoning Services for the Debugging of Description Logic Terminologies. In *IJCAI*, pages 355–362, 2003.