Abstract Argumentation via Monadic Second Order Logic*

Wolfgang Dvořák[1], Stefan Szeider[2], and Stefan Woltran[2]

[1] Theory and Applications of Algorithms Group, University of Vienna
[2] Institute of Information Systems, Vienna University of Technology

Abstract. We propose the formalism of Monadic Second Order Logic (MSO) as a unifying framework for representing and reasoning with various semantics of abstract argumentation. We express a wide range of semantics within the proposed framework, including the standard semantics due to Dung, semi-stable, stage, cf2, and resolution-based semantics. We provide building blocks which make it easy and straight-forward to express further semantics and reasoning tasks. Our results show that MSO can serve as a *lingua franca* for abstract argumentation that directly yields to complexity results. In particular, we obtain that for argumentation frameworks with certain structural properties the main computational problems with respect to MSO-expressible semantics can all be solved in linear time. Furthermore, we provide a novel characterization of resolution-based grounded semantics.

1 Introduction

Starting with the seminal work by Dung [18] the area of argumentation has evolved to one of the most active research branches within Artificial Intelligence (see, e.g., [6]). Dung's abstract argumentation frameworks, where arguments are seen as abstract entities which are just investigated with respect to how they relate to each other, in terms of "attacks", are nowadays well understood and different semantics (i.e., the selection of sets of arguments which are jointly acceptable) have been proposed. In fact, there seems to be no single "one suits all" semantics, but it turned out that studying a particular setting within various semantics and to compare the results is a central research issue within the field. Different semantics give rise to different computational problems, such as deciding whether an argument is acceptable with respect to the semantics under consideration, that require different approaches for solving these problems.

This broad range of semantics for abstract argumentation demands for a *unifying framework* for representing and reasoning with the various semantics. Such a unifying framework would allow us to see what the various semantics have in common, in what they differ, and ideally, it would offer generic methods for solving the computational problems that arise within the various semantics. Such a unifying framework should be general enough to accommodate most of the significant semantics, but simple enough to be decidable and computationally feasible.

* Dvořák's and Woltran's work has been funded by the Vienna Science and Technology Fund (WWTF) through project ICT08-028 and Szeider's work has been funded by the European Research Council (ERC), grant reference 239962 (COMPLEX REASON)

In this paper we propose such a unifying framework. We express several semantics within the framework, and we study its properties. The proposed unifying framework is based on the formalism of Monadic Second Order Logic (MSO), which is a fragment of Second Order logic with relational variables restricted to unary. MSO provides higher expressiveness than First Order Logic while it has more appealing algorithmic properties than full Second Order logic. Furthermore, MSO plays an important role in various parts of Computer Science. For instance, by Büchi's Theorem, a formal language is regular if and only if it can be expressed by MSO (this also provides a link between MSO and finite automata); furthermore, by Courcelle's Theorem, MSO expressible properties can be checked in linear time on structures of bounded treewidth.

Main Contributions. The results in this paper can be summarized as follows:

(1) We express a wide range of semantics within our proposed framework, including the standard semantics due to Dung, semi-stable, stage, cf2, and resolution-based semantics. For the latter, we present a new characterization that admits an MSO-encoding without quantification over sets of attacks and thus provides additional algorithmic implications.
(2) We provide MSO-building blocks which make it easy and straight-forward to express other semantics or to create new ones or variants.
(3) We also illustrate that any labeling-based semantics can be canonically expressed within our framework. We show that the main computational problems can be solved in linear time for all semantics expressible in our framework when restricted to argumentation frameworks of certain structures. This includes decision problems such as skeptical and brave acceptance, but also counting problems, for instance, determining how many extensions contain a given argument.

Our results show that MSO is indeed a suitable unifying framework for abstract argumentation and can serve as a *lingua franca* for further investigations. Furthermore, recent systems [28,29] showed quite impressive performance for evaluating MSO formulas over graphs, thus the proposed framework can be exploited as a rapid-prototyping approach to experiment with established and novel argumentation semantics.

Finally, we want to emphasise that in contrast to existing work [19,20,21,22] our goal is not to provide new complexity results for a particular argumentation semantics, but we propose MSO as a general logical framework for specifying argumentation semantics, having fixed-parameter tractability results as a neat side effect (compared to other approaches discussed below). Thus, in contrast to previous work, where MSO techniques were used as an auxiliary tool for achieving tractability results for particular semantics, our approach intends to raise MSO to a new conceptual level.

Related Work. Using MSO as a tool to express AI formalisms has been advocated in [26,27]. In terms of abstract argumentation MSO-encodings were given in [19,22] and implications in terms of parameterized complexity also appeared in [20,21].

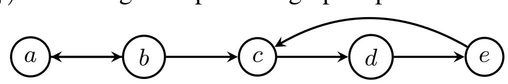
Finding a uniform logical representation for abstract argumentation has been subject of several papers. While [7] used propositional logic for this purpose, [24] showed that quantified propositional logic admits complexity-adequate representations. Another branch of research focuses on logic programming as common grounds for different argumentation semantics, see [32] for a survey. Finally also the use of constraint satisfaction techniques was suggested [1,8]. All this research was mainly motivated by implementation issues and led to systems such as ASPARTIX [23]. As mentioned above, also MSO can serve this purpose, but in addition yields further results "for free", in particular in terms of complexity.

2 Background

We start this section by introducing (abstract) argumentation frameworks [18] and recalling the semantics we study in this paper (see also [4]).

Definition 1. *An* argumentation framework (AF) *is a pair $F = (A, R)$ where A is a set of arguments and $R \subseteq A \times A$ is the attack relation. The pair $(a, b) \in R$ means that a attacks b. We say that an argument $a \in A$ is* defended *(in F) by a set $S \subseteq A$ if, for each $b \in A$ such that $(b, a) \in R$, there exists a $c \in S$ such that $(c, b) \in R$.*

Example 1. In the following we use the AF $F = (\{a, b, c, d, e\}, \{(a, b), (b, a), (b, c), (c, d), (d, e), (e, c)\})$ as running example. The graph representation is given as follows:



Semantics for argumentation frameworks are given via a function σ which assigns to each AF $F = (A, R)$ a set $\sigma(F) \subseteq 2^A$ of extensions. We first consider for σ the functions $naive$, stb, adm, com, prf, grd, stg, and sem which stand for naive, stable, admissible, complete, preferred, grounded, stage, and semi-stable semantics, respectively. Towards the definition of these semantics we introduce two more formal concepts.

Definition 2. *Given an AF $F = (A, R)$, the characteristic function $\mathcal{F}_F : 2^A \Rightarrow 2^A$ of F is defined as $\mathcal{F}_F(S) = \{ x \in A \mid x \text{ is defended by } S \}$. For a set $S \subseteq A$ and an argument $a \in A$, we write $S \rightarrowtail^R a$ (resp. $a \rightarrowtail^R S$) in case there is an argument $b \in S$, such that $(b, a) \in R$ (resp. $(a, b) \in R$). Moreover, for a set $S \subseteq A$, we denote the set of arguments attacked by S as $S^{\oplus}_R = \{ x \mid S \rightarrowtail^R x \}$, and resp. $S^{\ominus}_R = \{ x \mid x \rightarrowtail^R S \}$, and define the* range *of S as $S^+_R = S \cup S^{\oplus}_R$.*

Example 2. In our running example $\mathcal{F}_F(\{a\}) = \{a\}$, $\mathcal{F}_F(\{b\}) = \{b, d\}$, $\{a\}^{\oplus}_R = \{b\}$, $\{b\}^{\oplus}_R = \{a, c\}$ and $\{a\}^+_R = \{a, b\}$, $\{b\}^+_R = \{a, b, c\}$.

Definition 3. *Let $F = (A, R)$ be an AF. A set $S \subseteq A$ is* conflict-free (in F)*, if there are no $a, b \in S$, such that $(a, b) \in R$. $cf(F)$ denotes the collection of conflict-free sets of F. For a conflict-free set $S \in cf(F)$, it holds that*

- *$S \in naive(F)$, if there is no $T \in cf(F)$ with $T \supset S$;*
- *$S \in stb(F)$, if $S^+_R = A$;*
- *$S \in adm(F)$, if $S \subseteq \mathcal{F}_F(S)$;*
- *$S \in com(F)$, if $S = \mathcal{F}_F(S)$;*
- *$S \in grd(F)$, if $S \in com(F)$ and there is no $T \in com(F)$ with $T \subset S$;*
- *$S \in prf(F)$, if $S \in adm(F)$ and there is no $T \in adm(F)$ with $S \subset T$;*

– *$S \in sem(F)$, if $S \in adm(F)$ and there is no $T \in adm(F)$ with $S_R^+ \subset T_R^+$;*
– *$S \in stg(F)$, if there is no $T \in cf(F)$, with $S_R^+ \subset T_R^+$.*

We recall that for each AF F, $stb(F) \subseteq sem(F) \subseteq prf(F) \subseteq com(F) \subseteq adm(F)$ holds, and that each of the considered semantics σ except stb satisfies $\sigma(F) \neq \emptyset$. Moreover grd yields a unique extension for each AF F (in what follows identified by $grd(F)$), which is the least fix-point of the characteristic function $\mathcal{F}_F$.

Example 3. Our running example F has four admissible sets, i.e. $adm(F) = \{\emptyset, \{a\}, \{b\}, \{b,d\}\}$, with $\{a\}$ and $\{b,d\}$ being the preferred extensions. The grounded extension is the empty set, moreover $com(F) = \{\emptyset, \{a\}, \{b,d\}\}$ and $stb(F) = sem(F) = stg(F) = \{\{b,d\}\}$.

On the base of these semantics one can define the family of resolution-based semantics [3], with the resolution-based grounded semantics being its most popular instance.

Definition 4. *Given AF $F = (A,R)$, a resolution $\beta \subset R$ of F is a $\subseteq$-minimal set of attacks such that for each pair $\{(a,b),(b,a)\} \subseteq R$ ($a \neq b$) either $(a,b) \in \beta$ or $(b,a) \in \beta$. We denote the set of all resolutions of an AF F by $\gamma(F)$. Given a semantics σ, the corresponding resolution-based semantics σ^* is given by $\sigma^*(F) = \min_{\subseteq} \bigcup_{\beta \in \gamma(F)} \{\sigma((A, R \setminus \beta))\}$.*

Example 4. For our example AF F we get the two resolutions $\{(a,b)\}$ and $\{(b,a)\}$. In the case of resolution based grounded semantics this yields two candidates for extensions $grd((A, R \setminus \{(a,b)\})) = \{b,d\}$ and $grd((A, R \setminus \{(b,a)\})) = \{a\}$. As they are not in $\subseteq$-relation both are resolution-based grounded extensions and thus $grd^*(F) = \{\{a\}, \{b,d\}\}$.

Finally, let us consider the semantics cf2, which was introduced in [5] as part of a general schema for argumentation semantics. cf2 semantics gained some interest as it handles even and odd length cycles of attacks in a similar way. Towards a definition of cf2 semantics we need the following concepts.

Definition 5. *Given an AF $F = (A,R)$ and a set $S \subseteq A$. By $SCC(F)$ we denote the set of all strongly connected components of F. $D_F(S)$ denotes the set of arguments $a \in A$ attacked by an argument $b \in S$ occurring in a different component. Finally, for $F = (A,R)$ and a set S of arguments, $F|_S := (A \cap S, R \cap (S \times S))$ and $F - S := F|_{A \setminus S}$.*

Example 5. To illustrate D_F consider our example AF and the set $\{b,d\}$. We have two components $\{a,b\}$ and $\{c,d,e\}$ and that b attacks c. Hence $c \in D_F(\{b,d\})$. Also b attacks a and d attacks e but as both conflicts are within one component they do not add to the set $D_F(\{b,d\})$ and we have $D_F(\{b,d\}) = \{c\}$.

Definition 6. *Given an AF $F = (A,R)$, for $S \subseteq A$ we have that $S \in cf2(F)$ if one of the following conditions holds: (i) $|SCC(F)| = 1$ and $S \in naive(F)$; (ii) $\forall C \in SCC(F) : C \cap S \in cf2(F|_C - D_F(S))$.*

Example 6. For our example AF we obtain $cf2(F) = \{\{a,e\}, \{a,d\}, \{a,c\}, \{b,d\}\}$.

Labeling-based semantics. So far we have considered so-called extension-based semantics. However, there are several approaches defining argumentation semantics via certain kind of labelings instead of extensions. As an example we consider the complete labelings from [11].

Definition 7. *Given an AF $F = (A, R)$, a function $\mathcal{L} : A \rightarrow \{\text{in}, \text{out}, \text{undec}\}$ is a* complete labeling *iff the following conditions hold: (i) $\mathcal{L}(a) = \text{in}$ iff for each b with $(b, a) \in R$, $\mathcal{L}(b) = \text{out}$; (ii) $\mathcal{L}(b) = \text{out}$ iff there exists b with $(b, a) \in R$, $\mathcal{L}(b) = \text{in}$.*

There is a one-to-one mapping between complete extensions and complete labelings, such that the set of arguments labeled with "in" corresponds to a complete extension.

Example 7. The example AF has three complete labelings corresponding to the three complete extensions: the labeling $\mathcal{L}_1$ corresponding to $\emptyset$ with $\mathcal{L}_1(a) = \mathcal{L}_1(b) = \mathcal{L}_1(c) = \mathcal{L}_1(d) = \mathcal{L}_1(e) = \text{undec}$; the labeling $\mathcal{L}_2$ corresponding to $\{a\}$ with $\mathcal{L}_2(a) = \text{in}$, $\mathcal{L}_2(b) = \text{out}$, and $\mathcal{L}_2(c) = \mathcal{L}_2(d) = \mathcal{L}_2(e) = \text{undec}$; and the labeling $\mathcal{L}_3$ corresponding to $\{b, d\}$ with $\mathcal{L}_1(b) = \mathcal{L}_1(d) = \text{in}$ and $\mathcal{L}_3(a) = \mathcal{L}_3(c) = \mathcal{L}_3(e) = \text{out}$.

Monadic Second Order Logic. Informally, Monadic Second Order Logic can be seen as an extension of First Order Logic that admits quantification over sets. First Order Logic is built from variables $x, y, z, \ldots$ referring to elements of the universe, atomic formulas $R(t_1, \ldots, t_k)$, $t_1 = t_2$, with t_i being variables or constants, the usual Boolean connectives, and quantification $\exists x$, $\forall x$. MSO_1 extends the language of First Order Logic by set variables $X, Y, Z, \ldots$, atomic formulas $t \in X$ with t a variable or constant, and quantification over set variables. We further consider MSO_2 an extension of MSO_1 which is only defined on graphs (which is perfectly fine for our purposes). MSO_2 adds variables $X^E, Y^E, Z^E, \ldots$ ranging over sets of edges of the graph and quantification over such variables. In the following when talking about MSO we refer to MSO_2.

For an MSO formula ϕ we usually write $\phi(x_1, \ldots, x_p, X_1, \ldots X_q)$ to denote that the free variables of ϕ are $x_1, \ldots, x_p, X_1, \ldots X_q$. For a graph $G = (V, E)$, $v_1, \ldots, v_p \in V$, and $A_1, \ldots A_q \subseteq V$, we write $G \models \phi(v_1, \ldots, v_p, A_1, \ldots A_q)$ to denote that the formula ϕ holds true for G if x_i is instantiated with v_i and X_j is instantiated with A_j, $1 \leq i \leq p$, $1 \leq j \leq q$.

3 Encoding Argumentation Semantics in MSO

Building Blocks. We first introduce some shorthands simplifying notation when dealing with subset relations and the range of extensions.

$$
\begin{array}{ll}
X \subseteq Y = \forall x\,(x \in X \rightarrow x \in Y) & x \notin X = \neg(x \in X) \\
X \subset Y = X \subseteq Y \wedge \neg(Y \subseteq X) & x \in X_R^+ = x \in X \vee \exists y(y \in X \wedge (y, x) \in R) \\
X \not\subseteq Y = \neg(X \subseteq Y) & X \subseteq_R^+ Y = \forall x\,(x \in X_R^+ \rightarrow x \in Y_R^+) \\
X \not\subset Y = \neg(X \subset Y) & X \subset_R^+ Y = X \subseteq_R^+ Y \wedge \neg(Y \subseteq_R^+ X)
\end{array}
$$

Another important notion that underlies argumentation semantics is the notion of a set being conflict-free. The following MSO formula encodes that a set X is conflict-free w.r.t. the attack relation R:

$$cf_R(X) = \forall x, y\,((x, y) \in R \rightarrow (\neg x \in X \vee \neg y \in X))$$

Next we give a building block for maximizing extensions using an (MSO expressible) order $\sqsubseteq$:

$$\max{}_{A,P(.),\sqsubseteq}(X) = P(X) \land \neg\exists Y\big(Y \subseteq A \land P(Y) \land X \sqsubset Y\big)$$

Clearly we can also implement minimization by inverting the order, i.e., $\min{}_{A,P(.),\sqsubseteq}(X) = \max{}_{A,P(.),\sqsupseteq}(X)$.

Standard Encodings. In the following we provide MSO-characterizations for the different argumentation semantics. The characterizations for adm, stb, prf are borrowed from [19] while those for sem, stg are borrowed from [22].

$$\begin{aligned}
naive_{A,R}(X) &= \max{}_{A,cf_R(.),\subseteq}(X)\\
adm_R(X) &= cf_R(X) \land \forall x,y\ \big(((x,y) \in R \land y \in X) \rightarrow\\
&\qquad\qquad\qquad\qquad \exists z(z \in X \land (z,x) \in R)\big)\\
com_{A,R}(X) &= adm_R(X) \land \forall x((x \in A \land x \notin X) \rightarrow\\
&\qquad\qquad\qquad\qquad \exists y((y,x) \in R \land \neg\exists z(z \in X \land (z,y) \in R)))\\
grd_{A,R}(X) &= \min{}_{A,com_{A,R}(.),\subseteq}(X)\\
stb_{A,R}(X) &= cf_R(X) \land \forall x(x \in A \rightarrow x \in X^+_R)\\
prf_{A,R}(X) &= \max{}_{A,adm_R(.),\subseteq}(X)\\
sem_{A,R}(X) &= \max{}_{A,adm_R(.),\subseteq^+_R}(X)\\
stg_{A,R}(X) &= \max{}_{A,cf_R(.),\subseteq^+_R}(X)
\end{aligned}$$

These characterisations are straight-forward translations of the definitions and thus can be easily checked to be correct.

Based on the above characterizations, we proceed with encodings for the resolution-based semantics as follows. Via $res_R(X^E)$, given as

$$\begin{aligned}
\forall x,y\big(&X^E \subseteq R \land (x,x) \in R \rightarrow (x,x) \in X^E \land\\
&(x \neq y \land (x,y) \in R) \rightarrow ((x,y) \in X^E \leftrightarrow (y,x) \notin X^E)\big),
\end{aligned}$$

we express modified frameworks $(A, R \setminus \beta)$ where β is a resolution according to Definition 4. Now resolution-based semantics are characterised by

$$\begin{aligned}
\sigma^*_{A,R}(X) = \exists X^E(&res_R(X^E) \land \sigma_{A,X^E}(X) \land\\
&\forall Y \forall Y^E(res_R(Y^E) \land \sigma_{A,Y^E}(Y) \rightarrow Y \not\subset X)).
\end{aligned} \tag{1}$$

Labeling-based semantics. There are several approaches to define argument semantics via different kind of argumentation labelings and almost all argumentation semantics admit a characterization via argument labelings. The general concept behind labelings is to use a fixed set of labels and assign to each argument a subset of them, or just a single label. Such labelings are valid if for each argument the assigned labels satisfy certain (qualitative) conditions concerning the labels of attacking arguments and the labels of the attacked arguments. Additionally one might demand that the set of arguments labeled by a specific label is maximal or minimal. All these properties can be easily expressed in MSO, which we illustrate for complete labelings. We encode an in, out, undec labeling $\mathcal{L}$ as a triple $(\mathcal{L}_{\text{in}}, \mathcal{L}_{\text{out}}, \mathcal{L}_{\text{undec}})$ where $\mathcal{L}_l := \{ a \in A \mid \mathcal{L}(a) = l \}$. To have these three sets disjoint, one uses the formula $\varphi = \forall x \in A((x \in \mathcal{L}_{\text{in}} \vee x \in \mathcal{L}_{\text{out}} \vee x \in \mathcal{L}_{\text{undec}}) \wedge (x \notin \mathcal{L}_{\text{in}} \vee x \notin \mathcal{L}_{\text{out}}) \wedge (x \notin \mathcal{L}_{\text{in}} \vee x \notin \mathcal{L}_{\text{undec}}) \wedge (x \notin \mathcal{L}_{\text{undec}} \vee x \notin \mathcal{L}_{\text{out}}))$. Now we can give an MSO formula $com_{A,R}(\mathcal{L}_{\text{in}}, \mathcal{L}_{\text{out}}, \mathcal{L}_{\text{undec}})$ expressing whether such a triple is a complete labeling:

$$\varphi \wedge \forall x \in X(x \in \mathcal{L}_{\text{in}} \leftrightarrow (\forall y \in X((y,x) \in R \rightarrow y \in \mathcal{L}_{\text{out}})))$$
$$\wedge \forall x \in X(x \in \mathcal{L}_{\text{out}} \leftrightarrow (\exists y \in X((y,x) \in R \wedge y \in \mathcal{L}_{\text{in}})))$$

Further, one can directly encode preferred labelings, which are defined as complete labelings with maximal $\mathcal{L}_{\text{in}}$.

$$prf_{A,R}(\mathcal{L}_{\text{in}}, \mathcal{L}_{\text{out}}, \mathcal{L}_{\text{undec}}) = com_{A,R}(\mathcal{L}_{\text{in}}, \mathcal{L}_{\text{out}}, \mathcal{L}_{\text{undec}}) \wedge \neg\exists \mathcal{L}'_{\text{in}}, \mathcal{L}'_{\text{out}}, \mathcal{L}'_{\text{undec}}$$
$$(\mathcal{L}_{\text{in}} \subset \mathcal{L}'_{\text{in}} \wedge com_{A,R}(\mathcal{L}'_{\text{in}}, \mathcal{L}'_{\text{out}}, \mathcal{L}'_{\text{undec}}))$$

MSO-characterization for $cf2$. The original definition of $cf2$ semantics is of recursive nature and thus not well suitable for a direct MSO-encoding. Hence we use an alternative characterisation of $cf2$ [25]. For this purpose we need the following definitions.

Definition 8. *Given an AF $F = (A, R)$, $B \subseteq A$, and $a, b \in A$, we define $a \Rightarrow_F^B b$ if and only if there exists a sequence $(b_i)_{1 \le i \le n}$ with $b_i \in B$, $b_1 = a$, $b_n = b$ and $(b_i, b_{i+1}) \in R$.*

The relation $\Rightarrow_F^B$ can be encoded in MSO by first defining a relation $\hat{R}_{R,B}(u,v) = (u,v) \in R \wedge u \in B \wedge v \in B$ capturing the allowed attacks and borrowing the following MSO-encoding for reachability [12]: $reach_R(x,y) = \forall X(x \in X \wedge [\forall u, v(u \in X \wedge R(u,v) \rightarrow v \in X)] \rightarrow y \in X)$. Finally we obtain $\Rightarrow_R^B (x,y) = reach_{\hat{R}_{R,B}}(x,y)$.

Definition 9. *For AF $F = (A, R)$ and sets $D, S \subseteq A$ we define: $\Delta_{F,S}(D) = \{ a \in A \mid \exists b \in S : b \neq a, (b,a) \in R, a \not\Rightarrow_F^{A \setminus D} b \}$. $\Delta_{F,S}$ denotes the least fixed-point of $\Delta_{F,S}(.)$.*

Example 8. Consider our example AF F and the set $\{b, d\}$. Towards the least fixed-point $\Delta_{F,\{b,d\}}$ first consider $\Delta_{F,\{b,d\}}(\emptyset)$. The arguments attacked by $\{b, d\}$ are a, c, e, but a and e having paths back to their attackers and thus $\Delta_{F,\{b,d\}}(\emptyset) = \{c\}$. Next consider $\Delta_{F,\{b,d\}}(\{c\})$. Still d attacks e but $e \not\Rightarrow_F^{A \setminus \{c\}} d$. Thus $\Delta_{F,\{b,d\}}(\{c\}) = \{c, e\}$ which is also the least fixed-point $\Delta_{F,\{b,d\}}$ of $\Delta_{F,\{b,d\}}(.)$.

One can directly encode whether an argument x is in the operator $\Delta_{F,S}(D)$ by $\Delta_{A,R,S,D}(x) = x \in A \wedge \exists b \in S(b \neq x \wedge (b,x) \in R \wedge \neg \Rightarrow_F^{A \setminus D} (x,b))\}$ and thus also whether x is in the least fixed-point $\Delta_{F,S}$, by $\Delta_{A,R,S}(x) = \exists X \subseteq A(x \in X \wedge \forall a(a \in X \leftrightarrow \Delta_{A,R,S,X}(a)) \wedge \neg\exists Y \subset X(\forall b(b \in Y \leftrightarrow \Delta_{A,R,S,Y}(b))))$.

Definition 10. *For AF F we define the separation of F as $[[F]] = \bigcup_{C \in SCCs(F)} F|_C$.*

Example 9. To obtain the separation of our example AF we have to delete all attacks that are not within an single SCC. That is we simple remove the attack (b, c) and obtain the AF $(\{a, b, c, d, e\}, \{(a, b), (b, a), (c, d), (d, e), (e, c)\})$ as the separation of our example AF.

The attack relation of the separation of an AF (A, R) is given by $R_{[[(A,R)]]}(x, y) = x \in A \wedge y \in A \wedge (x, y) \in R \wedge \Rightarrow_R^A (y, x)$.

The following result provides an alternative characterization for *cf2* semantics.

Proposition 1 ([25]). *For any AF F, $S \in cf2(F)$ iff $S \in cf(F) \cap naive([[F - \Delta_{F,S}]])$.*

Example 10. For example consider the *cf2* extension $\{a, d\}$ of our running example. Clearly $\{a, d\} \in cf(F)$ and as illustrated before $\Delta_{F,\{b,d\}} = \{c, e\}$. We obtain $[[F - \Delta_{F,S}]] = (\{a, b, d\}, \{(a, b), (b, a)\})$ and thus also $\{a, d\} \in naive([[F - \Delta_{F,S}]])$.

Using the above Proposition we obtain the following MSO characterisation of *cf2*.

$$cf2(X) = cf_R(X) \wedge naive_{\hat{A}, R_{[[(\hat{A},R)]]}}(X) \quad \text{where} \quad \hat{A}(x) = x \in A \wedge \neg\Delta_{A,R,X}(x)$$

4 Algorithmic Implications

Most computational problems studied for AFs are computationally intractable (see, e.g., [19]), while the importance of efficient algorithms is evident. An approach to deal with intractable problems comes from parameterized complexity theory and is based on the fact, that many hard problems become polynomial-time tractable if some problem parameter is bounded by a fixed constant. In case the order of the polynomial bound is independent of the parameter one speaks of *fixed-parameter tractability* (FPT).

One popular parameter for graph-based problems is *treewidth* [9] which intuitively measures how tree-like a graph is. One weakness of treewidth is that it only captures sparse graphs. The parameter *clique-width* [17] generalizes treewidth, in the sense that each graph class of bounded treewidth has also bounded clique-width, but clique-width also captures a wide range of dense graphs.[3]

Both parameters have already been considered for abstract argumentation [19,20,21] and are closely related to MSO by means of meta-theorems. One such meta-theorem is due to [13] and shows that one can solve any graph problem that can be expressed in MSO_1 in linear time for graphs of clique-width bounded by some fixed constant k, when given together with a certain algebraic representation of the graph, a so called k-expression. A similar result is Courcelle's seminal meta-theorem [15,16] for MSO_2 and treewidth (which is also based on a certain structural decomposition of the graph, a so called tree-decomposition). Together with results from [10,30] stating that also k-expressions and tree-decompositions can be computed in linear time if k is bounded by a constant we get the following meta-theorem.

[3] As we do not make direct use of them, we omit the formal definitions of treewidth and clique-width here; the interested reader is referred to other sources [19,20]. We just note that these parameters are originally defined for undirected graphs, but can directly be used for AFs, as well.

Theorem 1. *For every fixed MSO formula* $\phi(x_1,\dots,x_i,X_1,\dots X_j,\ X_1^E,\dots X_l^E)$ *and integer* c*, there is a linear-time algorithm that, given a graph* (V,E) *of treewidth* $\le c$*,* $v_k \in V$*,* $A_k \subseteq V$*, and* $B_k \subseteq E$ *decides whether* $(V,E) \models \phi(v_1,\dots,v_i,A_1,\dots A_j,B_1,\dots B_l)$*. If* ϕ *is in MSO*$_1$*, then this also holds for graphs of clique-width* $\le c$*.*

The theorem can be extended to capture also counting and enumeration problems [2,14].

In the next theorem we give fixed-parameter tractability results w.r.t. the parameters treewidth and clique-width for the main reasoning problems in abstract argumentation.

Theorem 2. *For each argumentation semantics* σ *that is expressible in MSO, the following tasks are fixed-parameter tractable w.r.t. the treewidth of the given AF:*

- *Deciding whether an argument* $a \in A$ *is in at least one* σ*-extension (Credulous acceptance).*
- *Deciding whether an argument* $a \in A$ *is in each* σ*-extension (Skeptical acceptance).*
- *Verifying that a set* $E \subseteq A$ *is a* σ*-extension (Verification).*
- *Deciding whether there exists a* σ*-extension (Existence).*
- *Deciding whether there exists a non-empty* σ*-extension (Nonempty).*
- *Deciding whether there is a unique* σ*-extension (Unique).*

If σ *is expressible in MSO*$_1$*, then the above tasks are also fixed-parameter tractable w.r.t. the clique-width of the AF.*

Proof. The result follows by Theorem 1 and the following MSO-encodings: Credulous acceptance: $\phi^\sigma_{\mathsf{Cred}}(x) = \exists X\ (x \in X \wedge \sigma_R(X))$; Skeptical acceptance: $\phi^\sigma_{\mathsf{Skept}}(x) = \forall X\ (\sigma_R(X) \rightarrow x \in X)$; Verification: $\phi^\sigma_{\mathsf{Ver}}(X) = \sigma_R(X)$; Existence: $\phi^\sigma_{\mathsf{Exists}} = \exists X \sigma_R(X)$; Nonempty: $\phi^\sigma_{\mathsf{Exists}^{\neg\emptyset}} = \exists X \exists x (\sigma_R(X) \wedge x \in X)$; and Unique: $\phi^\sigma_{\mathsf{U}} = \exists X \sigma_R(X)) \wedge \neg\exists Y (Y \neq X \wedge \sigma_R(Y))$. We would like to note that these encodings do not use quantification over edge sets whenever σ is free of such a quantification. □

MSO is also a gentle tool for studying the relation between different semantics, as illustrated by Theorem 3.

Theorem 3. *For any argumentation semantics* σ, σ' *expressible in MSO, the following tasks are fixed-parameter tractable w.r.t. the treewidth of the given AF.*

- *Deciding whether* $\sigma(F) = \sigma'(F)$ *(Coincidence).*
- *Deciding whether arguments skeptically accepted w.r.t.* σ *are also skeptically accepted w.r.t.* σ' *(Skepticism 1).*
- *Deciding whether arguments credulously accepted w.r.t.* σ *are also credulously accepted w.r.t.* σ' *(Skepticism 2).*
- *Deciding whether* $\sigma(F) \subseteq \sigma'(F)$ *(Skepticism 3).*

If σ *is expressible in MSO*$_1$ *the above tasks are also fixed-parameter tractable w.r.t. the clique-width of the AF.*

Proof. The result follows by Theorem 1 and the following MSO-encodings: Coincidence: $\phi^{\sigma}_{\mathsf{Coin}}(x) = \forall X\ (\sigma_R(X) \leftrightarrow \sigma'_R(X))$; Skepticism 1: $\phi^{\sigma}_{\mathsf{sk1}}(x) = \forall x(\phi^{\sigma}_{\mathsf{Skept}}(x) \rightarrow \phi^{\sigma'}_{\mathsf{Skept}}(x))$; Skepticism 2: $\phi^{\sigma}_{\mathsf{sk1}}(x) = \forall x(\phi^{\sigma}_{\mathsf{Cred}}(x) \rightarrow \phi^{\sigma'}_{\mathsf{Cred}}(x))$; Skepticism 3: $\phi^{\sigma}_{\mathsf{sk1}}(x) = \forall X(\sigma_{A,R}(X) \rightarrow \sigma'_{A,R}(X))$. □

One prominent instantiation of the first problem mentioned in Theorem 3 is deciding whether an AF is coherent, i.e., whether stable and preferred extensions coincide.

Most of the characterizations we have provided so far are actually in MSO_1 and by the above results we obtain fixed-parameter tractability for treewidth and clique-width. The notable exception is the schema (1) we provided for the resolution-based semantics. There is no straight forward way to reduce this MSO_2 formula into MSO_1 (and thus providing complexity results in terms of clique-width) and in general it is unclear whether this is possible at all. Surprisingly, in the case of resolution-based grounded semantics one can get rid off the explicit quantification over sets of attacks as we show next.

5 An MSO_1-characterization for *grd**

We provide a novel characterisation of resolution-based grounded semantics that avoids the quantification over sets of attacks in schema, as in (1), and thus yields an MSO_1-encoding. To this end we first restrict the class of resolutions we have to consider when showing that a given set is a complete extension of some resolved AF.

Lemma 1. *For each AF $F = (A, R)$ and $E \in grd^*(F)$, there exists a resolution β with $\{ (b, a) \mid a \in E, b \notin E, \{(a, b), (b, a)\} \subseteq R \} \subseteq \beta$ such that $E \in com(A, R \setminus \beta)$.*

Proof. As $E \in grd^*(F)$ we have that there exists a resolution β' such that $E \in grd(A, R \setminus \beta')$. Now let us define β as $\{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \} \cup (\beta' \cap (A \setminus E \times A \setminus E))$. Clearly E is conflict-free in $(A, R \setminus \beta)$. Next we show that (i) $E^{\oplus}_{R\setminus\beta'} = E^{\oplus}_{R\setminus\beta}$ and (ii) $E^{\ominus}_{R\setminus\beta'} \supseteq E^{\ominus}_{R\setminus\beta}$.

For (i), let us first consider $b \in E^{\oplus}_{R\setminus\beta'}$. Then there exists $(a, b) \in R \setminus \beta'$ with $a \in E$ and by construction also $(a, b) \in R \setminus \beta$ and thus $b \in E^{\oplus}_{R\setminus\beta}$. Now let us consider $b \in E^{\oplus}_{R\setminus\beta}$. Then there exists $(a, b) \in R \setminus \beta$ with $a \in E$ and by construction either $(a, b) \in R \setminus \beta'$ or $(b, a) \in R \setminus \beta'$. In the first case clearly $b \in E^{\oplus}_{R\setminus\beta'}$. In the latter case b attacks E and as E is admissible in $(A, R \setminus \beta')$ there exists $c \in E$ such that $(c, b) \in R \setminus \beta'$, hence $b \in E^{\oplus}_{R\setminus\beta'}$. For (ii) consider $b \in E^{\ominus}_{R\setminus\beta}$, i.e., exists $a \in E$ such that $(b, a) \in R \setminus \beta$. By the construction of β we have that $(a, b) \notin R$ and therefore $(b, a) \in R \setminus \beta'$. Hence also $b \in E^{\ominus}_{R\setminus\beta'}$.

As $E \in adm(A, R\setminus\beta')$ we have that $E^{\ominus}_{R\setminus\beta'} \subseteq E^{\oplus}_{R\setminus\beta'}$ and by the above observations then also $E^{\ominus}_{R\setminus\beta} \subseteq E^{\oplus}_{R\setminus\beta}$. Thus E is an admissible set. Finally let us consider an argument $a \in A \setminus E^{\oplus}_{R\setminus\beta}$. In the construction of β the incident attacks of a are not effected and hence $\{a\}^{\ominus}_{R\setminus\beta'} = \{a\}^{\ominus}_{R\setminus\beta}$. That is E defends a in $(A, R \setminus \beta)$ iff E defends a in $(A, R \setminus \beta')$. Now as $E \in com(A, R \setminus \beta')$ we have that a is not defended and hence $E \in com(A, R \setminus \beta)$. □

With this result at hand, we can give an alternative characterization for grd^*.

Lemma 2. *For each AF $F = (A, R)$ and $E \subseteq A$, $E \in grd^*(F)$ if and only if the following conditions hold:*

1. *there exists a resolution β with $\{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \} \subseteq \beta$ and $E \in com(A, R \setminus \beta)$*
2. *E is $\subseteq$-minimal w.r.t. (1).*

Proof. Let us first recall that, by definition, the grounded extension is the $\subseteq$-minimal complete extension and hence $grd^* = com^*$.

$\Rightarrow$: Let $E \in grd^*(F)$. Then by Lemma 1, E fulfills condition (1). Further we have that each set E satisfying (1) is a complete extension of a resolved AF. As by definition E is $\subseteq$-minimal in the set of all complete extensions of all resolved AFs it is also minimal for those satisfying (1).

$\Leftarrow$: As E satisfies (1) it is a complete extension of a resolved AF. Now towards a contradiction let us assume it is not a resolution-based grounded extension. Then there exists $G \in grd^*(F)$ with $G \subset E$. But by Lemma 1 G fulfills condition (1) and thus $G \subset E$ contradicts (2). $\square$

In the next step we look for an easier characterization of condition (1).

Lemma 3. *For each AF $F = (A, R)$ and $E \subseteq A$ the following statements are equivalent:*

1. *There exists a resolution β with $\{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \} \subseteq \beta$ and $E \in com(A, R \setminus \beta)$.*
2. *$E \in com(A, R \setminus \{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \})$ and $grd^*(A \setminus E^+_R, R \cap ((A \setminus E^+_R) \times (A \setminus E^+_R))) = \{\emptyset\}$.*

Proof. In the following we will use the shorthands $R^* = R \setminus \{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \}$ and $(A', R') = (A \setminus E^+_R, R \cap ((A \setminus E^+_R) \times (A \setminus E^+_R)))$.

(1) $\Rightarrow$ (2): Consider a resolution β such that $E \in com(A, R \setminus \beta)$. We first show that then also $E \in com(A, R^*)$. By construction we have that for arbitrary $b \in A$ that (a) $E \rightarrowtail^R b$ iff $E \rightarrowtail^{R \setminus \beta} b$ iff $E \rightarrowtail^{R^*} b$, and (b) $b \rightarrowtail^{R \setminus \beta} E$ iff $b \rightarrowtail^{R^*} E$. Hence we have that (i) $E \in adm(A, R \setminus \beta)$ iff $E \in adm(A, R^*)$ and (ii) $E^+_R = E^+_{R \setminus \beta} = E^+_{R^*}$. By definition of complete semantics, $E \in com(A, R \setminus \beta)$ is equivalent to for each argument $b \in A \setminus E$ there exists an argument $c \in A$ such that $c \rightarrowtail^{R \setminus \beta} b$ and $E \not\rightarrowtail^{R \setminus \beta} c$. As $R^* \supseteq R \setminus \beta$ we obtain that $(c, b) \in R \setminus \beta$ implies $(c, b) \in R^*$. Using (a) we obtain that $E \in com(A, R \setminus \beta)$ implies for each argument $b \in A \setminus E$ existence of an argument $c \in A$ such that $(c, b) \in R^*$ and $E \not\rightarrowtail^{R^*} c$, i.e., $E \in com(A, R^*)$.

Now addressing $grd^*(A', R') = \{\emptyset\}$ we again use the assumption $E \in com(A, R \setminus \beta)$, i.e., each argument which is defended by E is already contained in E, we have that $grd(A \setminus E^+_{R \setminus \beta}, R \setminus \beta \cap ((A \setminus E^+_R) \times (A \setminus E^+_R))) = grd(A', R' \setminus \beta) = \{\emptyset\}$. Note that $\beta' = \beta \cap R'$ is a resolution of (A', R') and that $grd(A', R' \setminus \beta) = grd(A', R' \setminus \beta') = \{\emptyset\}$. We can conclude that $grd^*(A', R') = \{\emptyset\}$.

(1) $\Leftarrow$ (2): Consider $\beta' \in \gamma(F)$ s.t. $grd(A', R' \setminus \beta') = \{\emptyset\}$; such a β' exists since $grd^*(A', R') = \{\emptyset\}$. Now consider the resolution $\beta = \{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \} \cup \beta'$. Again, by construction of β we have that for arbitrary $b \in A$: (a) $E \rightarrowtail^R b$ iff $E \rightarrowtail^{R \setminus \beta} b$ iff $E \rightarrowtail^{R^*} b$, and (b) $b \rightarrowtail^{R \setminus \beta} E$ iff

$b \rightarrowtail^{R^*} E$. Hence we obtain that $E \in adm(A, R \setminus \beta)$. Using $R = E^+_{R\setminus\beta} = E^+_{R^*}$ we have $grd(A \setminus E^+_{R\setminus\beta}, (R \setminus \beta) \cap ((A \setminus E^+_R) \times (A \setminus E^+_R))) = grd(A', R' \setminus \beta') = \{\emptyset\}$. Thus, $E \in com(A, R \setminus \beta)$. □

Finally we exploit a result from [3].

Proposition 2 ([3]). *For every AF $F = (A, R)$, $grd^*(F) = \{\emptyset\}$ iff for each minimal SCC S of F at least one one of the following conditions holds: (i) S contains a self-attacking argument; (ii) S contains a non-symmetric attack; and (iii) S contains an undirected cycle*

Based on the above observations we obtain the following characterization of resolution-based grounded semantics.

Theorem 4. *For each AF $F = (A, R)$, the grd^*-extensions are the $\subseteq$-minimal sets $E \subseteq A$ such that:*

1. *$E \in com(A, R')$ with $R' = R \setminus \{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \})$.*
2. *Each minimal SCC S of $\hat{F} = (A \setminus E^+_R, R \cap A \setminus E^+_R \times A \setminus E^+_R)$ satisfies one of the following conditions: S contains a self-attacking argument; S contains a non-symmetric attack; or S contains an undirected cycle*

Proof. By Lemma 3, condition (1) in Lemma 2 is equivalent to $E \in com(A, R \setminus \{ (b, a) \mid a \in E, \{(a, b), (b, a)\} \subseteq R \})$ and $grd^*(A \setminus E^+_R, R \cap ((A \setminus E^+_R) \times (A \setminus E^+_R))) = \{\emptyset\}$. The former being condition (1) of the theorem. The latter, due to Proposition 2, is equivalent to condition (2) of the theorem. □

Having Theorem 4 at hand we can build an MSO_1-encoding as follows. First we encode the attack relation R' as $R'_E(x, y) = (x, y) \in R \wedge \neg(x \in E \wedge y \notin E \wedge (x, y) \in R \wedge (y, x) \in R)$. Then the AF $\hat{F} = (\hat{A}, \hat{R})$ is given by:

$$\hat{A}_{A,R,E}(x) = x \in A \wedge x \notin E \wedge \neg\exists y \in E : R'_E(y, x)$$
$$\hat{R}_{E,R}(x, y) = (x, y) \in R \wedge A^*_{A,R,E}(x) \wedge A^*_{A,R,E}(y)$$

Based on reachability we can easily specify whether arguments are strongly connected $SC_R(x, y) = reach_R(x, y) \wedge reach_R(y, x)$, and a predicate that captures all arguments in minimal SCCs $minSCC_{A,R}(x) = A(x) \wedge \neg\exists y\ (A(y) \wedge reach_R(y, x) \wedge \neg reach_R(x, y))$. It remains to encode the check for each SCC.

$$\begin{aligned}
C1_R(x) &= \exists y(SC_R(x, y) \wedge (y, y) \in R)\\
C2_R(x) &= \exists y, z(SC_R(x, y) \wedge SC_R(x, z) \wedge (y, z) \in R \wedge (z, y) \notin R)\\
C3_R(x) &= \exists X(\exists y \in X \wedge \forall y \in X[SC_R(x, y) \wedge\\
&\qquad \exists u, v \in X : u \neq v \wedge (u, y) \in R \wedge (y, v) \in R])\\
C_R(x) &= C1_R(x) \vee C2_R(x) \vee C3_R(x)
\end{aligned}$$

Finally using Theorem 4 we obtain an MSO_1-encoding for resolution-based grounded semantics:

$$grd^*{}_{A,R}(X) = cand_{A,R}(X) \wedge \neg\exists Y(cand_{A,R}(Y) \wedge Y \subset X)$$

where $cand_{A,R}(X)$ stands for

$$com_{A,R'_X}(X) \wedge \forall x(minSCC_{\hat{A}_{A,R,E},\hat{R}_{E,R}}(x) \rightarrow C_{\hat{R}_{E,R}}(x)).$$

6 Conclusion

In this paper we have shown that Monadic Second Order Logic (MSO) provides a suitable unifying framework for abstract argumentation. We encoded the most popular semantics within MSO and gave building blocks illustrating that MSO can naturally capture several concepts that are used for specifying semantics. This shows that MSO can be used as rapid prototyping tool for the development of new semantics.

Moreover, we gave a new characterisation of resolution-based grounded semantics that admits an MSO_1-encoding. This shows that reasoning in this semantics is tractable for frameworks of bounded clique-width. In fact, the collection of encodings we provided here shows that acceptance as well as other reasoning tasks are fixed-parameter tractable for several semantics w.r.t. the clique-width (hence also for treewidth).

For future work we suggest to study whether also other instantiations of the resolution-based semantics can be expressed in MSO_1 (recall that we provided already a schema for MSO_2-encodings). Moreover, it might be interesting to compare the performance of MSO tools with dedicated argumentation systems. Finally, we want to advocate the use of MSO for automated theorem discovery [31]. In fact, our encodings allow us to express meta-statements like "does it hold for AFs F that each σ-extension is also a σ'-extension." Although we have to face undecidability for such formulas, there is the possibility that MSO-theorem provers come up with a counter-model. Thus, in a somewhat similar way as Weydert [33], who used a First Order Logic encoding of complete semantics to show certain properties for semi-stable semantics of infinite AFs, MSO can possibly be used to support the argumentation researcher in obtaining new insights concerning the wide range of different argumentation semantics.

References

1. Amgoud, L., Devred, C.: Argumentation frameworks as constraint satisfaction problems. In: Benferhat, S., Grant, J. (eds.) Proc. SUM. LNCS, vol. 6929, pp. 110–122. Springer (2011)
2. Arnborg, S., Lagergren, J., Seese, D.: Easy problems for tree-decomposable graphs. J. Algorithms 12(2), 308–340 (1991)
3. Baroni, P., Dunne, P.E., Giacomin, M.: On the resolution-based family of abstract argumentation semantics and its grounded instance. Artif. Intell. 175(3-4), 791–813 (2011)
4. Baroni, P., Giacomin, M.: Semantics of abstract argument systems. In: Rahwan, I., Simari, G. (eds.) Argumentation in Artificial Intelligence, pp. 25–44. Springer (2009)
5. Baroni, P., Giacomin, M., Guida, G.: SCC-recursiveness: A general schema for argumentation semantics. Artif. Intell. 168(1-2), 162–210 (2005)
6. Bench-Capon, T.J.M., Dunne, P.E.: Argumentation in artificial intelligence. Artif. Intell. 171(10-15), 619–641 (2007)
7. Besnard, P., Doutre, S.: Checking the acceptability of a set of arguments. In: Delgrande, J.P., Schaub, T. (eds.) Proc. NMR. pp. 59–64 (2004)
8. Bistarelli, S., Campli, P., Santini, F.: Finding partitions of arguments with Dung's properties via SCSPs. In: Chu, W.C., Wong, W.E., Palakal, M.J., Hung, C.C. (eds.) Proc. SAC. pp. 913–919. ACM (2011)
9. Bodlaender, H.L.: A tourist guide through treewidth. Acta Cybernetica 11, 1–21 (1993)
10. Bodlaender, H.L.: A linear-time algorithm for finding tree-decompositions of small treewidth. SIAM J. Comput. 25(6), 1305–1317 (1996)

11. Caminada, M., Gabbay, D.M.: A logical account of formal argumentation. Studia Logica 93(2), 109–145 (2009)
12. Courcelle, B., Engelfriet, J.: Graph structure and monadic second-order logic, a language theoretic approach, vol. 2012. Cambridge University Press (2011)
13. Courcelle, B., Makowsky, J.A., Rotics, U.: Linear time solvable optimization problems on graphs of bounded clique-width. Theory Comput. Syst. 33(2), 125–150 (2000)
14. Courcelle, B., Makowsky, J.A., Rotics, U.: On the fixed parameter complexity of graph enumeration problems definable in monadic second-order logic. Discr. Appl. Math. 108(1-2), 23–52 (2001)
15. Courcelle, B.: Recognizability and second-order definability for sets of finite graphs. Tech. Rep. I-8634, Université de Bordeaux (1987)
16. Courcelle, B.: Graph rewriting: an algebraic and logic approach. In: Handbook of theoretical computer science, Vol. B, pp. 193–242. Elsevier, Amsterdam (1990)
17. Courcelle, B., Engelfriet, J., Rozenberg, G.: Context-free handle-rewriting hypergraph grammars. In: Ehrig, H., Kreowski, H.J., Rozenberg, G. (eds.) In: Ehrig, H., Kreowski, H.J., Rozenberg, G. (eds.) Proc. Graph Grammars. LNCS, vol. 532, pp. 253–268 (1991)
18. Dung, P.M.: 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)
19. Dunne, P.E.: Computational properties of argument systems satisfying graph-theoretic constraints. Artif. Intell. 171(10-15), 701–729 (2007)
20. Dvořák, W., Szeider, S., Woltran, S.: Reasoning in argumentation frameworks of bounded clique-width. In: Baroni, P., Cerutti, F., Giacomin, M., Simari, G.R. (eds.) Proc. COMMA. FAIA, vol. 216, pp. 219–230. IOS Press (2010)
21. Dvořák, W., Pichler, R., Woltran, S.: Towards fixed-parameter tractable algorithms for abstract argumentation. Artif. Intell. 186, 1–37 (2012)
22. Dvořák, W., Woltran, S.: Complexity of semi-stable and stage semantics in argumentation frameworks. Inf. Process. Lett. 110(11), 425–430 (2010)
23. Egly, U., Gaggl, S.A., Woltran, S.: Answer-set programming encodings for argumentation frameworks. Argument and Computation 1(2), 147–177 (2010)
24. Egly, U., Woltran, S.: Reasoning in argumentation frameworks using quantified boolean formulas. In: Dunne, P.E., Bench-Capon, T.J.M. (eds.) Proc. COMMA. FAIA, vol. 144, pp. 133–144. IOS Press (2006)
25. Gaggl, S.A., Woltran, S.: cf2 semantics revisited. In: Baroni, P., Cerutti, F., Giacomin, M., Simari, G.R. (eds.) Proc. COMMA. FAIA, vol. 216, pp. 243–254. IOS Press (2010)
26. Gottlob, G., Pichler, R., Wei, F.: Bounded treewidth as a key to tractability of knowledge representation and reasoning. Artif. Intell. 174(1), 105–132 (2010)
27. Gottlob, G., Szeider, S.: Fixed-parameter algorithms for artificial intelligence, constraint satisfaction, and database problems. The Computer Journal 51(3), 303–325 (2006)
28. Kneis, J., Langer, A., Rossmanith, P.: Courcelle's theorem - a game-theoretic approach. Discrete Optimization 8(4), 568–594 (2011)
29. Langer, A., Reidl, F., Rossmanith, P., Sikdar, S.: Evaluation of an MSO-solver. In: Proc. of ALENEX 2012, 55–63, SIAM (2012)
30. Oum, S., Seymour, P.D.: Approximating clique-width and branch-width. J. Comb. Theory, Ser. B 96(4), 514–528 (2006)
31. Tang, P., Lin, F.: Discovering theorems in game theory: Two-person games with unique pure nash equilibrium payoffs. Artif. Intell. 175(14-15), 2010–2020 (2011)
32. Toni, F., Sergot, M.: Argumentation and answer set programming. In: Balduccini, M., Son, T. (eds.) Gelfond Festschrift, LNAI, vol. 6565, pp. 164–180. Springer (2011)
33. Weydert, E.: Semi-stable extensions for infinite frameworks. In: Proc. BNAIC, pp. 336–343 (2011)