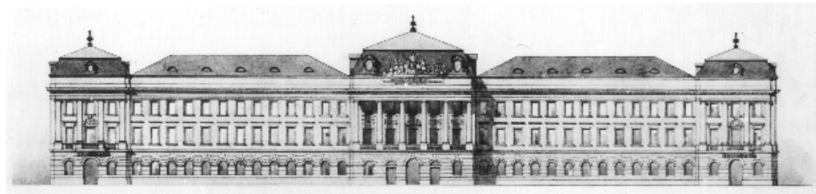
dbai

TECHNICAL REPORT



INSTITUT FÜR INFORMATIONSSYSTEME

ABTEILUNG DATENBANKEN UND ARTIFICIAL INTELLIGENCE

Technical Note: On the Complexity of the Uniqueness Problem in Abstract Argumentation

DBAI-TR-2017-108

Wolfgang Dvořák

Institut für Informationssysteme
Abteilung Datenbanken und
Artificial Intelligence
Technische Universität Wien
Favoritenstr. 9
A-1040 Vienna, Austria
Tel: +43-1-58801-18403
Fax: +43-1-58801-18493
sekret@dbai.tuwien.ac.at
www.dbai.tuwien.ac.at

DBAI TECHNICAL REPORT
2017

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

DBAI TECHNICAL REPORT
DBAI TECHNICAL REPORT DBAI-TR-2017-108, 2017

Technical Note: On the Complexity of the Uniqueness Problem in Abstract Argumentation

Wolfgang Dvořák [1]

Abstract. In this technical note we consider the *Uniqueness Problem* in abstract argumentation frameworks, i.e., the problem of deciding whether an AF yields a unique extensions w.r.t. a given semantics. In particular, we are interested in the computational complexity of the uniqueness problem. We survey results from the literature, that are often only implicitly, and complement them by our own results to provide an almost complete complexity landscape for a broad range of semantics, i.e., Dung's original semantics, cf2, resolution-based grounded, ideal, eager, semi-stable and stage semantics.

[1]TU Wien, Institute of Information Systems, Vienna, Austria. E-mail: dvorak@dbai.tuwien.ac.at

1 Preliminaries

In this section, we introduce argumentation frameworks [6] and recall the semantics we study (for an comprehensive introduction see [1]).

Definition 1. *An* argumentation framework (AF) *is a pair $F = (A, R)$ where $A \subseteq U$ is a finite 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 use A_F to refer to A and R_F to refer to R.*

For a set $S \subseteq A$ and an argument $a \in A$, we write $S \rightarrowtail_F a$ (resp. $a \rightarrowtail_F S$) in case there is an argument $b \in S$, such that $(b, a) \in R$ (resp. $(a, b) \in R$). In case no ambiguity arises, we may use $\rightarrowtail$ instead of $\rightarrowtail^R$.

Definition 2. *Given an AF F and $S \subseteq U$, we define $S^+_F = \{x \mid \exists y \in S : (y, x) \in F_R\}$, $S^-_F = \{x \mid \exists y \in S : (x, y) \in F_R\}$, and the* range of S in F *as $S^\oplus_F = S \cup S^+_F$.*

Definition 3. *Given an AF $F = (A, R)$, we say that an an argument $a \in A$ is* defended *(in F) by a set $S \subseteq A$ if $\{x\}^-_F \subseteq S^+_F$. The* characteristic function *$\mathcal{F}_F : 2^A \to 2^A$ of F is defined as $\mathcal{F}_F(S) = \{x \in A \mid x$ is defended by S in $F\}$.*

Semantics for argumentation frameworks are defined as functions σ which assign to each AF F a set $\sigma(F) \subseteq 2^{A(F)}$ of extensions. We consider for σ the functions $naive$, grd, stb, adm, com, $cf2$, $ideal$, $eager$, prf, sem and stg which stand for naive, grounded, stable, admissible, complete, and cf2, ideal, eager, preferred semi-stable and stage extensions, respectively.

Definition 4. *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 $S^\oplus_F = A$;*
- *$S \in adm(F)$, 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 $T \subset S$ such that $T \in com(F)$;*
- *$S \in prf(F)$, if $S \in adm(F)$ and there is no $T \supset S$ such that $T \in adm(F)$.*
- *$S \in ideal(F)$ if S is $\subseteq$-maximal among $\{S' \mid S' \in adm(F),\ S' \subseteq E$ for each $E \in prf(F)\}$.*
- *$S \in sem(F)$, if $S \in adm(F)$ and there is no $T \in adm(F)$ with $S^\oplus_R \subset T^\oplus_R$;*
- *$S \in eager(F)$ if S is $\subseteq$-maximal among $\{S' \mid S' \in adm(F),\ S' \subseteq E$ for each $E \in sem(F)\}$.*
- *$S \in stg(F)$, if there is no $T \in cf(F)$, with $S^\oplus_R \subset T^\oplus_R$.*

For the more evolved definitions of cf2 and resolution-based grounded semantics the interested reader is referred to [1, 2]. We wont exploit the definitions of the two semantics but obtain our results due to some meta-level observations. We recall that for each AF F, the grounded, ideal and eager semantics always yield a unique extension.

We will further assume that the reader is familiar with standard complexity classes and the polynomial hierarchy. For a short introduction into complexity theory in the context of abstract argumentation the interested reader is referred to [9, 8] or the corresponding chapter in the forthcoming handbook of formal argumentation.

2 Results

This technical note is about the computational complexity of deciding whether a given AF has a unique extensions w.r.t. given semantics. The problem can be formalized as follows.

- *Uniqueness of the solution* $Unique_\sigma$: Given AF $F = (A, R)$. Is there a unique set $S \in \sigma(F)$, i.e., is $\sigma(F) = \{S\}$?

Our results are summarized in Table 1 together with existing results. We start with some general observations and then discuss the complexity for the different semantics.

Table 1: Complexity of the Uniqueness problem in abstract argumentation. $\mathcal{C}$-c denotes completeness for class $\mathcal{C}$. For results marked with * the hardness holds under randomized reductions.

	cf	$naive$	grd	stb	adm	com	$resGr$	$cf2$
$Unique_\sigma$	in L	in L	trivial	DP-c*	coNP-c	coNP-c	in P	in P

	$ideal$	$eager$	prf	sem	stg
$Unique_\sigma$	trivial	trivial	coNP-c	in Θ_2^{P}/ DP-hard*	in Θ_2^{P}/ DP-hard*

2.1 General Observations

Let us first state the obvious, for unique status semantics the answer to the uniqueness problem is clearly true.

Observation 1. *For unique status semantics σ the answer to the $Unique_\sigma$ problem is trivially true.*

By the above observations we immediately get the results for grounded, ideal and eager semantics listed in Table 1.

Next we give a generic upper bound for semantics that always yield at least one extensions (this excludes stable semantics) based on the complexity of verifying an extension.

Proposition 1. *For semantics σ, with $|\sigma(F)| \geq 1$ for all AFs F we have that $Unique_\sigma$ is in* $\mathsf{coNP}^{\mathsf{V}}$ *where* V *is the complexity of* Ver_σ.

Proof. To check that $|\sigma(F)| > 1$ we can use the following NP^{V} procedure: It first non-deterministically guesses two sets and then verifies that they are different from each other and both are extensions (for the latter the V oracle is used). That is we have a $\mathsf{coNP}^{\mathsf{V}}$ procedure for testing that an AF has at most one extension, which for the considered semantics is equivalent $Unique_\sigma$. □

2.2 Conflict-free and Naive Semantics

First, cf semantics yield a unique extension iff all arguments in the AF are self-attacking, and $naive$ semantics yield a unique extensions if there is no conflict between non self-attacking arguments. Both criteria can be easily tested in L.

Theorem 1. *$Unique_{cf}$ and $Unique_{naive}$ are in* L.

2.3 Admissible and Complete Semantics

For adm, and com we can use the $\mathsf{coNP}^{\mathsf{V}}$ algorithm (cf. Proposition 1). In the following we complement this upper bound by matching lower bounds.

The lower bound for admissible semantics is by the corresponding hardness of deciding whether an AF has a non-empty admissible set.

Theorem 2. *$Unique_{adm}$ is* coNP*-complete.*

Proof. Notice that the empty set is always admissible and thus $Unique_{adm}$ is equivalent to the problem of deciding whether there is no non-empty admissible set which is well-known to be coNP-complete [5, 9]. □

The hardness result for complete semantics is by a reduction in [7] that was used for hardness results of ideal semantics.

Theorem 3. *$Unique_{com}$ is* coNP*-complete.*

Proof. The membership is immediately by Proposition 1 and by the fact that complete extension can be verified in polynomial-time.

The hardness is by the following simple modification of the so-called *standard reduction* [9]. Given a propositional formula φ in CNF given by a set of clauses C over the atoms Y, we define from φ as $G_\varphi = (A, R)$, where

$$
\begin{aligned}
A &= \{\varphi, \bar{\varphi}\} \cup C \cup Y \cup \bar{Y} \\
R &= \{(c, \varphi) \mid c \in C\} \cup \{(l, c) \mid l \in c, c \in C\} \cup \\
&\quad \{(x, \bar{x}), (\bar{x}, x) \mid x \in Y\} \cup \{(\varphi, \bar{\varphi}), (\bar{\varphi}, \varphi), (\bar{\varphi}, \bar{\varphi})\} \cup \\
&\quad \{(\bar{\varphi}, \bar{x}) \mid x \in Y\}
\end{aligned}
$$

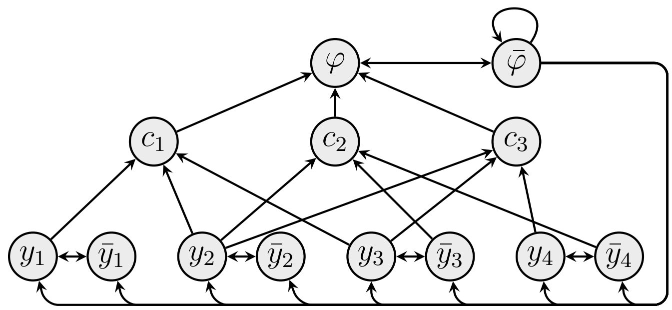


Figure 1: Illustration of the reduction G_φ, for the propositional formula φ with clauses $\{\{y_1, y_2, y_3\}, \{\bar{y}_2, \bar{y}_3, \bar{y}_4)\}, \{\bar{y}_1, \bar{y}_2, y_4\}\}$.

The construction is Illustrated in Figure 1. It is minor modification of the reduction used in [7, Theorem 1] to show the coNP-hardness of verifying an ideal extension. We have that the grounded extension is always $empty$ and following the arguments in [7, Theorem 1] we have that there is non-empty complete extension iff the formula is satisfiable. That is, G_φ has a unique complete extension iff there formula φ is unsatisfiable. coNP-hardness follows. □

2.4 Stable Semantics

When testing for the uniqueness of extensions stable semantics have a special behaviour, as it does not guarantees that there is at least one extension. That is, additionally to the procedure described in the proof of Proposition 1 we have to perform a check that there exists an extensions, which gives a DP-algorithm.

Theorem 4. *$Unique_{stb}$ is* DP*-complete (under randomized reductions).*

Proof. Membership in DP*:* We can test whether $Unique_{stb}$ holds by two independent tests:

1. We have to test whether there exists an extensions which is NP-complete [5, 9].
2. We perform the algorithm from proof of Proposition 1 to test whether there exist two or more extensions. As a stable extension can be verified in polynomial-time we get a coNP procedure.

Now as an instance is true iff it passes both tests we have a DP-algorithm.

DP-*hardness:* Here we consider the problem unique sat, i.e., the problem of deciding whether a given propositional formula has exactly one model. This problem is known to be DP-complete under randomized reductions [12] (see also [7, Section 3.6]). Now consider the following reduction (which is a slight adaptation of the standard reduction): Given a propositional formula φ in CNF given by a set of clauses C over the atoms Y, we define the translation from φ as $F_\varphi = (A, R)$,

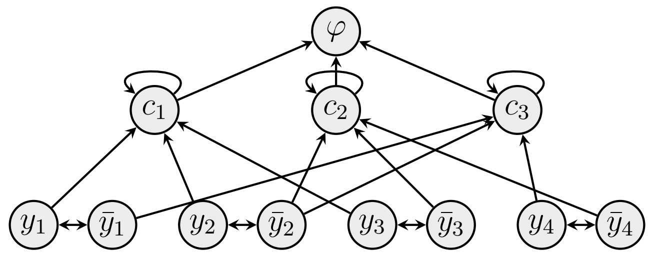


Figure 2: Illustration of the reduction F_φ, for the propositional formula φ with clauses $\{\{y_1, y_2, y_3\}, \{\bar{y}_2, \bar{y}_3, \bar{y}_4)\}, \{\bar{y}_1, \bar{y}_2, y_4\}\}$.

where

$$
\begin{aligned}
A &= \{\varphi\} \cup C \cup Y \cup \bar{Y} \\
R &= \{(c, \varphi), (c, c) \mid c \in C\} \cup \{(l, c) \mid l \in c, c \in C\} \cup \\
&\quad \{(x, \bar{x}), (\bar{x}, x) \mid x \in Y\}
\end{aligned}
$$

The construction is illustrated in Figure 2. Now it is easy to show that each model $M \subseteq Y$ corresponds to a stable extensions $M \cup \{\bar{x} \mid Y \setminus M\} \cup \{\varphi\}$ of the F_φ and vice versa. That is, F_φ has a unique stable extension iff φ has a unique model. $\square$

2.5 Preferred semantics

For prf semantics the $\mathsf{coNP}^{\mathsf{V}} = \Pi_2^{\mathsf{P}}$ algorithm can be improved by exploiting the observations that (1) there is always a conflict between any two preferred extensions of an AF and (2) two admissible sets that have a conflict cannot be contained in the same admissible set. That is, an AF has two admissible sets that are in conflict with each other iff it has two preferred extensions. Thus it suffices to guess two sets, and verify that both sets are admissible and there is a conflict between the two sets.

Theorem 5. *[10, Proposition 6] $Unique_{prf}$ is* coNP*-complete.*

2.6 Resolution-based grounded and cf2 semantics

In this section we exploit result on enumeration algorithms for abstract argumentation [11]. We say that an enumeration algorithms works with polynomial time when (a) the first extension is computed in polynomial time; (b) the computation of each further extensions only takes polynomial time since the last extension was reported; and (c) it only takes polynomial time to report that there is no further extension. Indeed we can use such an algorithm to decide the uniqueness of an extension. That is, we start the enumeration algorithm and if it terminates after the first extension we can answer yes and if it computes a second extension we can terminate and answer no. In any case the algorithm works in polynomial time.

Observation 2. *If the σ-extensions of an AF can be enumerated with polynomial delay then* $\mathit{Unique}_\sigma \in \mathsf{P}$.

We can now exploit the polynomial delay enumeration algorithms for $\mathit{cf2}$- and resGr by [11].

Theorem 6. $\mathit{Unique}_{\mathit{cf2}} \in \mathsf{P}$ *and* $\mathit{Unique}_{\mathit{resGr}} \in \mathsf{P}$.

Proof. By the above observation and the fact that $\mathit{cf2}$- and resGr- extensions can be enumerated with linear delay [11]. □

2.7 Semi-stable and stage Semantics

Again for stage and semi-stable semantics the straightforward $\mathsf{coNP}^{\mathsf{V}} = \Pi_2^{\mathsf{P}}$ algorithm is not optimal. However, the problem is harder than for preferred semantics. While we have to leave the exact complexity open we provide an algorithm improving over the the standard algorithm and give a lower bound that is worse than the complexity of preferred semantics.

The first step of our improved algorithm is to compute the maximal size of the range of any extension. This can be done with a Θ_2^{P}-algorithm. In the second step we then exploit that a witness for falsifying uniqueness is given by two admissible, conflict-free respectively, sets where one has maximal range size and the other has an incomparable range. With this kind of witnesses we can give a coNP algorithm for checking uniqueness, given the size of the largest range. Putting this two parts together we get a Θ_2^{P}-algorithm.

Theorem 7. $\mathit{Unique}_{\mathit{sem}}$ *and* $\mathit{Unique}_{\mathit{stg}}$ *are in* Θ_2^{P}.

Proof. Consider the following algorithm:

- We start with a binary search in the interval $[1, n]$ to determine the size of the largest range of any admissible, conflict-free resp., set. In each step of the binary search we ask the NP oracle whether there is a admissible, conflict-free resp., set that has a range of size larger some value i. This binary search can be clearly done in $\log(n)$ steps and thus also only $\log(n)$ many oracle calls are required

- Given the size $s_{\max}$ of the largest range we use a NP-oracle to disprove Unique_σ. Within this oracle we guess two sets A, B and verify that both are admissible, conflict-free resp., sets and the range of A has size $s_{\max}$ and $B^+ \not\subseteq A^+$.

The correctness of the algorithm is by the following facts: (1) A set A satisfying the above conditions is a semi-stable, stage resp., extension. (2) A set B satisfying the above conditions is either a semi-stable, stage resp., extension or there is a semi-stable, stage resp., extension C with $B^+ \subset C^+$. However, as $B^+ \not\subseteq A^+$ we have $A \neq C$. Thus the existence of such A, B guarantee there existence of at least two extensions. Vice versa, if there are several extensions there is one, lets call it A, with size $s_{\max}$ and all the others, lets call them B, have incomparable range, i.e,$B^+ \not\subseteq A^+$. That is, an AF has two or more extension iff sets A, B satisfying the conditions of the algorithm exists. □

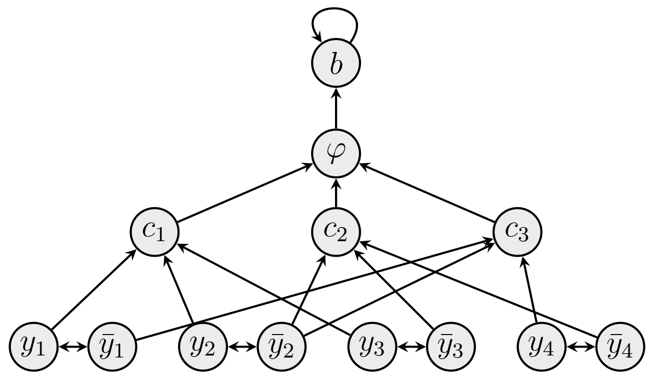


Figure 3: Illustration of the reduction H_φ, for the propositional formula φ with clauses $\{\{y_1, y_2, y_3\}, \{\bar{y}_2, \bar{y}_3, \bar{y}_4)\}, \{\bar{y}_1, \bar{y}_2, y_4\}\}$.

Notice that the first step of the algorithm can be easily shown to be complete for the optimisation class corresponding to Θ_2^{P}. However it is not clear whether this step is really necessary.

Next we giving the DP that holds under randomized reductions. That is we start from the problem unique sat hat is DP-complete under randomized reductions [12] but provide a standard reduction.

Theorem 8. *$Unique_{sem}$ and $Unique_{stg}$ are* DP*-hard (under randomized reductions).*

Proof. DP*-hardness:* Here we consider the problem unique sat, of deciding whether a given propositional formula has exactly one model. This problem is known to be DP-complete under randomized reductions [12] (see also [7, Section 3.6]). Now consider the following reduction (which is a slight adaptation of the standard reduction): Given a propositional formula φ in CNF given by a set of clauses C over the atoms Y, we define the translation from φ as $H_\varphi = (A, R)$, where

$$\begin{aligned}
A &= \{\varphi, b\} \cup C \cup Y \cup \bar{Y} \\
R &= \{(c, \varphi), (c, c) \mid c \in C\} \cup \{(l, c) \mid l \in c, c \in C\} \cup \\
&\quad \{(x, \bar{x}), (\bar{x}, x) \mid x \in Y\} \cup \{(\varphi, b), (b, b)\}
\end{aligned}$$

The construction is illustrated in Figure 3. Now it is easy to show that each model $M \subseteq Y$ corresponds to a stable extensions $M \cup \{\bar{x} \mid Y \setminus M\} \cup \{\varphi\}$ of the H_φ and vice versa. Recall that if a stable extension exists then stable, semi-stable, and stage semantics coincide. Thus, if the unique sat instance has at least one model we have valid reduction from unique sat to $Unique_{sem}$ and $Unique_{stg}$. We next consider the case where the formula has no model: If there is no model there is no stable extension and each assignment $A \subseteq Y$ gives rise to a semi-stable, stage resp., extension $M \cup \{\bar{x} \mid Y \setminus M\} \cup \{c \in C \mid A$ does not satisfy $c\}$. Notice that, each of these extensions has the range $Y \cup \bar{Y} \cup C \cup \{\varphi\}$, i.e., all arguments but b are either in the extension or attacked by the extension. Each admissible set, conflict-free set resp., with larger range would be stable, which is in contradiction to our assumption that φ has no model. Hence, in the case where the formula has

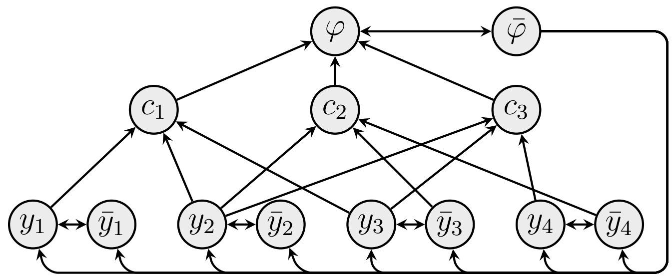


Figure 4: Illustration of the reduction G'_φ, for the propositional formula φ with clauses $\{\{y_1, y_2, y_3\}, \{\bar{y}_2, \bar{y}_3, \bar{y}_4)\}, \{\bar{y}_1, \bar{y}_2, y_4\}\}$.

no model we have several semi-stable extensions and thus the reduction is valid. That is, H_φ has a unique semi-stable, stage resp., extension iff φ has a unique model. $\square$

A standard technique [4] to show DP or Θ_2^{P} hardness is to show NP and coNP hardness and then show that the problem satisfies the AND_2 property and for Θ_2^{P} hardness additionally that the problem satisfies the OR_ω property. This technique was earlier used in the context of abstract argumentation [7, 3] (see, in particular the discussion in [7, Section 3.6]). Next we provide some results that could be first step to also establish such results for $Unique_{stb}$, $Unique_{sem}$ and $Unique_{stg}$.

Lemma 1. *$Unique_{sem}$, $Unique_{stg}$ and $Unique_{stb}$ have the AND_2 property.*

Proof. Given two instances, i.e., two AFs F,G, of our problem we can easily merge them to one instance that is true iff both of the original instances where true as follows. In a first step rename that arguments of the two AFs such that the AFs F, G are disjoint. Then create the new AF H as the (disjoint) union of the two frameworks. As unconnected parts of the framework are evaluated independently the extensions of $\sigma(H)$ are given by $\sigma(H) = \{E_1 \cup E_2 \mid E_1 \in \sigma(F), E_2 \in \sigma(G)\}$. That is, H has a unique extension iff both F and G have a unique extension. $\square$

Proposition 2. *$Unique_{sem}$, $Unique_{stg}$, and $Unique_{stb}$ are* coNP*-hard.*

Proof. This is by a reduction from the UNSAT problem. Given a propositional formula φ in CNF given by a set of clauses C over the atoms Y, we define from φ as $G'_\varphi = (A, R)$, where

$$\begin{aligned} A =& \{\varphi, \bar{\varphi}\} \cup C \cup Y \cup \bar{Y} \\ R =& \{(c, \varphi) \mid c \in C\} \cup \{(l, c) \mid l \in c, c \in C\} \cup \\ & \{(x, \bar{x}), (\bar{x}, x) \mid x \in Y\} \cup \{(\varphi, \bar{\varphi}), (\bar{\varphi}, \varphi)\} \cup \\ & \{(\bar{\varphi}, \bar{x}) \mid x \in Y\} \end{aligned}$$

The construction is illustrated in Figure 4. First notice that $C \cup \{\bar{\varphi}\}$ is always a stable extensions and recall that if a stable extension exists then stable, semi-stable, and stage semantics coincide. Thus, we can restrict ourselves to stable semantics. Now it is easy to show that each model $M \subseteq Y$ corresponds to a stable extensions $M \cup \{\bar{x} \mid Y \setminus M\} \cup \{\varphi\}$ of the H_φ and each stable extension, except the former mentioned, corresponds to a model of φ. vice versa. That is, G'_φ has a unique stable extension iff φ is unsatisfiable. $\square$

References

[1] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *Knowledge Eng. Review*, 26(4):365–410, 2011.

[2] Pietro Baroni, Paul E. Dunne, and Massimiliano Giacomin. On the resolution-based family of abstract argumentation semantics and its grounded instance. *Artif. Intell.*, 175(3-4):791–813, 2011.

[3] Martin Caminada, Walter A. Carnielli, and Paul E. Dunne. Semi-stable semantics. *J. Log. Comput.*, 22:1207–1254, 2012.

[4] Richard Chang and Jim Kadin. On computing boolean connectives of characteristic functions. *Mathematical Systems Theory*, 28(3):173–198, 1995.

[5] Yannis Dimopoulos and Alberto Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comput. Sci.*, 170(1-2):209–244, 1996.

[6] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[7] Paul E. Dunne. The computational complexity of ideal semantics. *Artif. Intell.*, 173(18):1559–1591, 2009.

[8] Paul E. Dunne, Wolfgang Dvořák, and Stefan Woltran. Parametric properties of ideal semantics. *Artificial Intelligence*, 202(0):1 – 28, 2013.

[9] Paul E. Dunne and Michael Wooldridge. Complexity of abstract argumentation. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 85–104. Springer US, 2009.

[10] Wolfgang Dvořák, Matti Järvisalo, Johannes Peter Wallner, and Stefan Woltran. Complexity-sensitive decision procedures for abstract argumentation. *Artificial Intelligence*, 206(0):53 – 78, 2014.

[11] Markus Kröll, Reinhard Pichler, and Stefan Woltran. On the complexity of enumerating the extensions of abstract argumentation frameworks. In *IJCAI 2017 (to appear)*, 2017.

[12] Leslie G. Valiant and Vijay V. Vazirani. NP is as easy as detecting unique solutions. *Theor. Comput. Sci.*, 47(3):85–93, 1986.