FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

dbai

A Comprehensive Analysis of the cf2 Argumentation Semantics:

From Characterization to Implementation

Sarah Alice Gaggl

Institute of Informationsystems, Vienna University of Technology

Vienna — March 4, 2013

FACULTY OF !NFORMATICS

W|W|T|F

VIENNA SCIENCE
AND TECHNOLOGY FUND

Motivation

- Argumentation is one of the major fields in Artificial Intelligence (AI).
- Applications in diverse domains (legal reasoning, multi-agent systems, social networks, e-government, decision support).
- Concept of abstract Argumentation Frameworks (AFs) [Dung, 1995] is one of the most popular approaches.
- Arguments and a binary attack relation between them, denoting conflicts, are the only components.
- Numerous semantics to solve the inherent conflicts by selecting acceptable sets of argument.
- Admissible-based versus naive-based semantics.
- Development of competitive systems.

Motivation ctd.

cf2 Semantics

- is based on decomposition of the framework along its strongly connected components (SCCs) [Baroni et al., 2005];
- does not require to defend arguments against attacks;
- allows to treat cycles in a more sensitive way than other semantics;
- is not well studied, due to quite complicated definition.

cf2 Semantics

- is based on decomposition of the framework along its strongly connected components (SCCs) [Baroni et al., 2005];
- does not require to defend arguments against attacks;
- allows to treat cycles in a more sensitive way than other semantics;
- is not well studied, due to quite complicated definition.

Goals of the Thesis

- Answer-set programming encodings for *cf2*.
- Alternative characterization.
- Verification of behavior on concrete instances.
- Identification of possible redundancies.
- Complete complexity analysis.

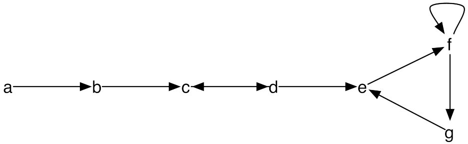
Outline

1. Background on abstract argumentation frameworks and semantics
2. Alternative characterization of $cf2$
3. Combining $cf2$ and stage semantics
4. Redundancies and strong equivalence
5. Computational complexity
6. Implementations
7. Conclusion

Abstract Argumentation Framework [Dung, 1995]

An abstract argumentation framework (AF) is a pair $F = (A, R)$, where A is a finite set of arguments and $R \subseteq A \times A$. Then $(a, b) \in R$ if a attacks b. Argument $a \in A$ is defended by $S \subseteq A$ (in F) iff, for each $b \in A$ with $(b, a) \in R$, S attacks b.

Example

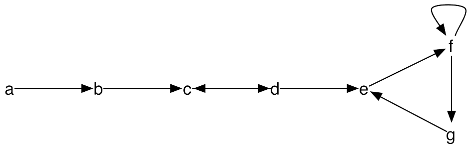


Semantics for AFs

Let $F = (A, R)$ and $S \subseteq A$, we say

- S is conflict-free in F, i.e. $S \in cf(F)$, if $\forall a, b \in S$: $(a, b) \notin R$;
- $S \in cf(F)$ is maximal conflict-free or naive (in F), i.e. $S \in naive(F)$, if $\forall T \in cf(F)$, $S \not\subset T$.

Example



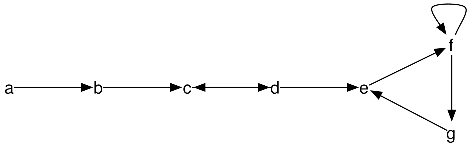
$$naive(F) = \{\{a, d, g\}, \{a, c, e\}, \{a, c, g\}\}.$$

Naive-based Semantics

Let $F = (A, R)$ and $S \subseteq A$. Let $S_R^+ = S \cup \{b \mid \exists a \in S$, s. t. $(a, b) \in R\}$ be the range of S. Then, a set $S \in cf(F)$ is

- a stable extension (of F), i.e. $S \in stable(F)$, if $S_R^+ = A$;
- stage in F, i.e. $S \in stage(F)$, if for each $T \in cf(F)$, $S_R^+ \not\subset T_R^+$.

Example



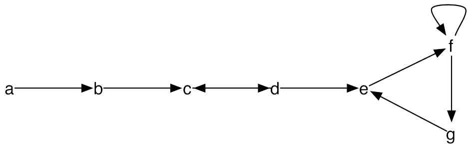
$$stable(F) = \emptyset, stage(F) = \{\{a, d, g\}, \{a, c, e\}, \{a, c, g\}\}.$$

Admissible-based Semantics

Then, $S \in cf(F)$ is

- admissible in F, i.e. $S \in adm(F)$, if each $a \in S$ is defended by S;
- a preferred extension (of F), i.e. $S \in pref(F)$, if $S \in adm(F)$ and for each $T \in adm(F)$, $S \not\subset T$.

Example



$$adm(F) = \{\emptyset, \{a\}, \{c\}, \{d\}, \{a,c\}, \{a,d\}\}, pref(F) = \{\{a,c\}, \{a,d\}\}.$$

cf2 Semantics

- One of the SCC-recursive semantics introduced in [Baroni et al., 2005].
- Naive-based semantics.
- Handles odd- and even-length cycles in a uniform way.
- Can accept arguments out of odd-length cycles.
- Can accept arguments attacked by self-attacking arguments.
- Satisfies most of the evaluation criteria proposed in [Baroni and Giacomin, 2007].

cf2 Semantics

- One of the SCC-recursive semantics introduced in [Baroni et al., 2005].
- Naive-based semantics.
- Handles odd- and even-length cycles in a uniform way.
- Can accept arguments out of odd-length cycles.
- Can accept arguments attacked by self-attacking arguments.
- Satisfies most of the evaluation criteria proposed in [Baroni and Giacomin, 2007].

Further Notations, let $F = (A, R)$

- $SCCs(F)$: set of strongly connected components of F,
- $C_F(a)$: the unique set $C \in SCCs(F)$, s.t. $a \in C$,
- $F|_S = ((A \cap S), R \cap (S \times S))$: sub-framework of F w.r.t. S,
- $F|_S - S' = F|_{S \setminus S'}$, $F - S = F|_{A \setminus S}$.

Definition ($D_F(S)$)

Let $F = (A, R)$ be an AF and $S \subseteq A$. An argument $b \in A$ is component-defeated by S (in F), if there exists an $a \in S$, such that $(a, b) \in R$ and $a \notin C_F(b)$. The set of arguments component-defeated by S in F is denoted by $D_F(S)$.

Definition ($D_F(S)$)

Let $F = (A, R)$ be an AF and $S \subseteq A$. An argument $b \in A$ is component-defeated by S (in F), if there exists an $a \in S$, such that $(a, b) \in R$ and $a \notin C_F(b)$. The set of arguments component-defeated by S in F is denoted by $D_F(S)$.

cf2 Extensions [Baroni et al., 2005]

Let $F = (A, R)$ be an argumentation framework and S a set of arguments. Then, S is a *cf2* extension of F, i.e. $S \in cf2(F)$, iff

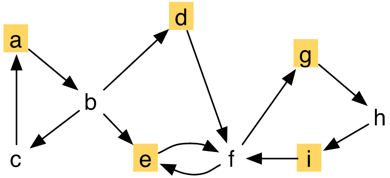
- $S \in naive(F)$, in case $|SCCs(F)| = 1$;
- otherwise, $\forall C \in SCCs(F)$, $(S \cap C) \in cf2(F|_C - D_F(S))$.

$S \in cf2(F)$ iff,

- $S \in naive(F)$, in case $|SCCs(F)| = 1$;
- otherwise, $\forall C \in SCCs(F)$, $(S \cap C) \in cf2(F|_C - D_F(S))$.

Example

$S = \{a, d, e, g, i\}$, $S \in cf2(F)$?



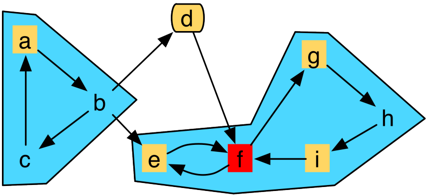
cf2 Semantics ctd.

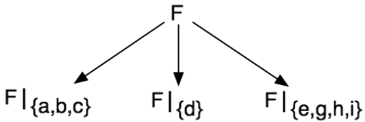
$S \in cf2(F)$ iff,

- $S \in naive(F)$, in case $|SCCs(F)| = 1$;
- otherwise, $\forall C \in SCCs(F)$, $(S \cap C) \in cf2(F|_C - D_F(S))$.

Example

$S = \{a, d, e, g, i\}$, $S \in cf2(F)$? $C_1 = \{a, b, c\}$, $C_2 = \{d\}$, $C_3 = \{e, f, g, h, i\}$ and $D_F(S) = \{f\}$.



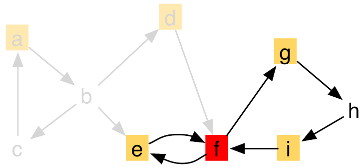


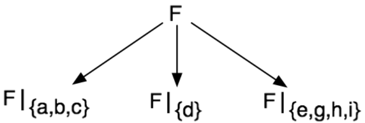
$S \in cf2(F)$ iff,

- $S \in naive(F)$, in case $|SCCs(F)| = 1$;
- otherwise, $\forall C \in SCCs(F)$, $(S \cap C) \in cf2(F|_C - D_F(S))$.

Example

$S = \{a, d, e, g, i\}$, $S \in cf2(F)$? $C_1 = \{a, b, c\}$, $C_2 = \{d\}$, $C_3 = \{e, f, g, h, i\}$ and $D_F(S) = \{f\}$.





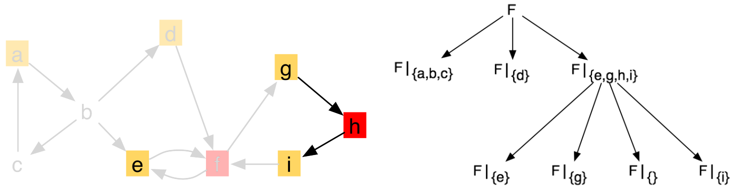
cf2 Semantics ctd.

$S \in cf2(F)$ iff,

- $S \in naive(F)$, in case $|SCCs(F)| = 1$;
- otherwise, $\forall C \in SCCs(F)$, $(S \cap C) \in cf2(F|_C - D_F(S))$.

Example

$S = \{a, d, e, g, i\}$, $S \in cf2(F)$? $C_4 = \{e\}$, $C_5 = \{g\}$, $C_6 = \{h\}$, $C_7 = \{i\}$
and $D_{F|_{\{e,g,h,i\}}}(\{e, g, i\}) = \{h\}$.



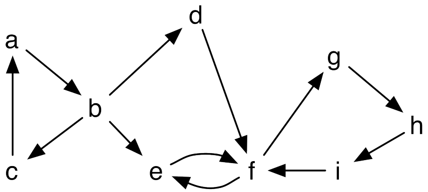
Alt. Characterization of *cf2*

- Original definition of *cf2* is rather cumbersome to be directly encoded in ASP due to the recursive computation of different sub-frameworks.
- In alternative characterization we shift the recursion to a certain set of arguments.
- This enables to directly
  - guess a set S;
  - check whether S is a naive extension of a certain instance of F.

Separation

An AF $F = (A, R)$ is called separated if for each $(a, b) \in R$, there exists a path from b to a. We define $[[F]] = \bigcup_{C \in SCCs(F)} F|_C$ and call $[[F]]$ the separation of F.

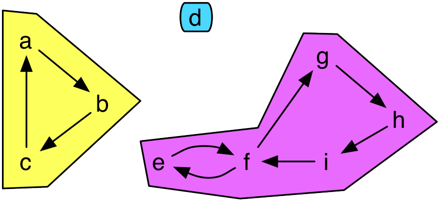
Example



Separation

An AF $F = (A, R)$ is called separated if for each $(a, b) \in R$, there exists a path from b to a. We define $[[F]] = \bigcup_{C \in SCCs(F)} F|_C$ and call $[[F]]$ the separation of F.

Example



Alt. Characterization of $cf2$ ctd.

Reachability

Let $F = (A, R)$ be an AF, B a set of arguments, and $a, b \in A$. We say that b is reachable in F from a modulo B, in symbols $a \Rightarrow_F^B b$, if there exists a path from a to b in $F|_B$.

dbai

Reachability

Let $F = (A, R)$ be an AF, B a set of arguments, and $a, b \in A$. We say that b is reachable in F from a modulo B, in symbols $a \Rightarrow_F^B b$, if there exists a path from a to b in $F|_B$.

Definition ($\Delta_{F,S}$)

For an AF $F = (A, R)$, $D \subseteq A$, and a set S of arguments,

$$\Delta_{F,S}(D) = \{a \in A \mid \exists b \in S : b \neq a, (b, a) \in R, a \not\Rightarrow_F^{A \setminus D} b\},$$

and $\Delta_{F,S}$ be the least fixed-point of $\Delta_{F,S}(\emptyset)$.

Alt. Characterization of *cf2* ctd.

cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

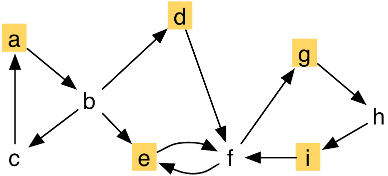
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

Example

$S = \{a, d, e, g, i\}$, $S \in naive(F)$.



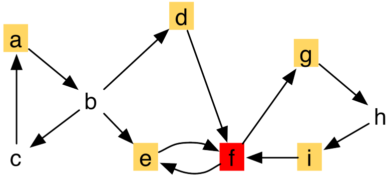
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

Example

$S = \{a, d, e, g, i\}$, $S \in naive(F)$, $\Delta_{F,S}(\emptyset) = \{f\}$.



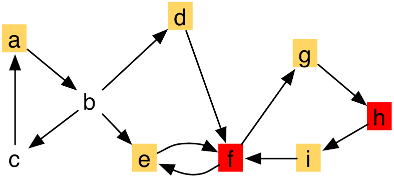
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

Example

$S = \{a, d, e, g, i\}$, $S \in naive(F)$, $\Delta_{F,S}(\{f\}) = \{f, h\}$.



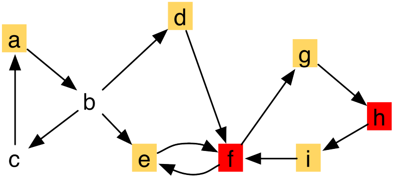
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

Example

$S = \{a, d, e, g, i\}$, $S \in naive(F)$, $\Delta_{F,S}(\{f, h\}) = \{f, h\}$.



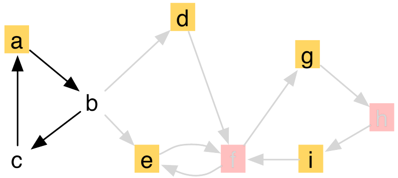
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

Example

$S = \{a, d, e, g, i\}$, $S \in naive(F)$, $\Delta_{F,S} = \{f, h\}$, $S \in naive([[F - \Delta_{F,S}]])$, thus $S \in cf2(F)$.



Alt. Characterization of *cf2* ctd.

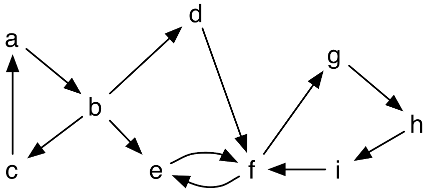
cf2 Extensions [Gaggl and Woltran, 2012]

Given an AF $F = (A, R)$.

$$cf2(F) = \{S \mid S \in naive(F) \cap naive([[F - \Delta_{F,S}]])\}.$$

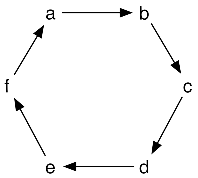
Example

$cf2(F) = \{\{a, d, e, g, i\}, \{c, d, e, g, i\}, \{b, f, h\}, \{b, g, i\}\}.$



- *cf2* produces questionable results on AFs with cycles of length ≥ 6.

Example

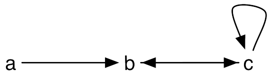


$$\textit{cf2}(F) = \textit{naive}(F) = \{\{a,d\},\{b,e\},\{c,f\},\{a,c,e\},\{b,d,f\}\};$$
$$\textit{stage}(F) = \{\{a,c,e\},\{b,d,f\}\}.$$

Shortcomings of *cf2* and Stage

- *cf2* produces questionable results on AFs with cycles of length ≥ 6.
- The grounded extension is not necessarily contained in every stage extension.
  - Stage semantics does not satisfy directionality.

Example



$$stage(F) = \{\{a\}, \{b\}\} \text{ but } cf2(F) = ground(F) = \{\{a\}\}.$$

Combining *cf2* and stage

We combine *cf2* and stage semantics [Dvorák and Gaggl, 2012a], by

- using the SCC-recursive schema of the *cf2* semantics and
- instantiate the base case with stage semantics.

We combine *cf2* and stage semantics [Dvorák and Gaggl, 2012a], by

- using the SCC-recursive schema of the *cf2* semantics and
- instantiate the base case with stage semantics.

stage2 Extensions

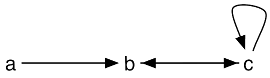
For any AF F,

$$stage2(F) = \{S \mid S \in naive(F) \cap stage([[F - \Delta_{F,S}]])\}.$$

$stage2$ Examples

For any AF F, $stage2(F) = \{S \mid S \in naive(F) \cap stage([[F - \Delta_{F,S}]])\}$.

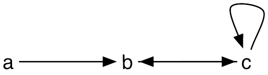
Example



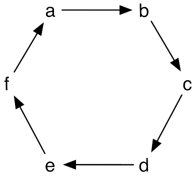
$$stage2(F) = cf2(F) = \{\{a\}\}, \text{ where } stage(F) = \{\{a\}, \{b\}\}.$$

For any AF F, $stage2(F) = \{S \mid S \in naive(F) \cap stage([[F - \Delta_{F,S}]])\}$.

Example

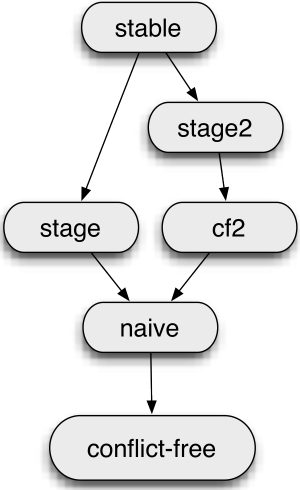


$stage2(F) = cf2(F) = \{\{a\}\}$, where $stage(F) = \{\{a\},\{b\}\}$.



$stage2(G) = stage(G) = \{\{a,c,e\},\{b,d,f\}\}$, but
$cf2(G) = naive(F) = \{\{a,d\},\{b,e\},\{c,f\},\{a,c,e\},\{b,d,f\}\}$.

Relations between Semantics



- Argumentation is a dynamic reasoning process.
- Which effect has additional information w.r.t. a semantics?
- Which information does not contribute to results no matter which changes are performed?
- Identification of kernels to remove redundant attacks [Oikarinen and Woltran, 2011].

Definition

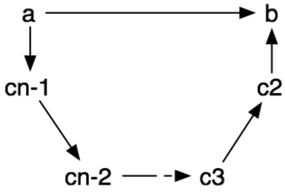
Two AFs F and G are strongly equivalent to each other w.r.t. a semantics σ, in symbols $F \equiv_s^\sigma G$, iff for each AF H, $\sigma(F \cup H) = \sigma(G \cup H)$.

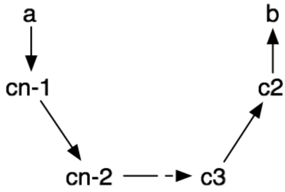
Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

Theorem

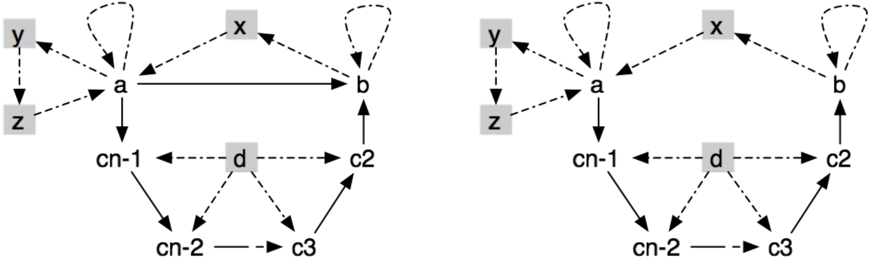
For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.





Theorem

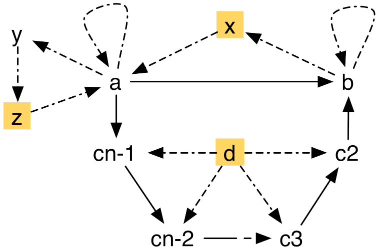
For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

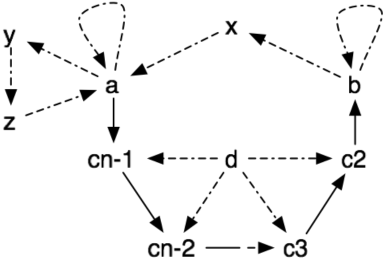


$$H = (A \cup \{d, x, y, z\}, \{(a,a), (b,b), (b,x), (x,a), (a,y), (y,z), (z,a), (d,c) \mid c \in A \setminus \{a,b\}\}).$$

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

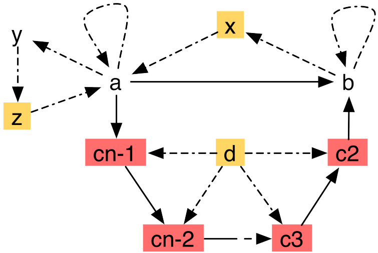


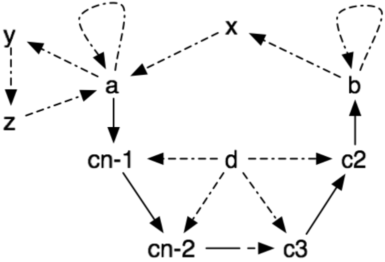


Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

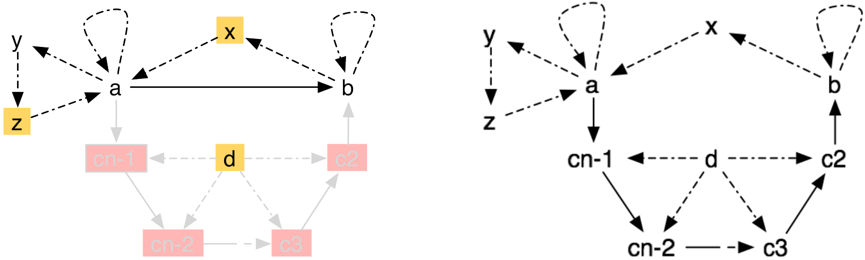




Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

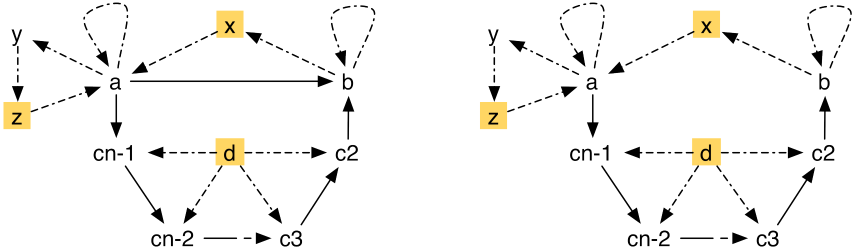
For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.



Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

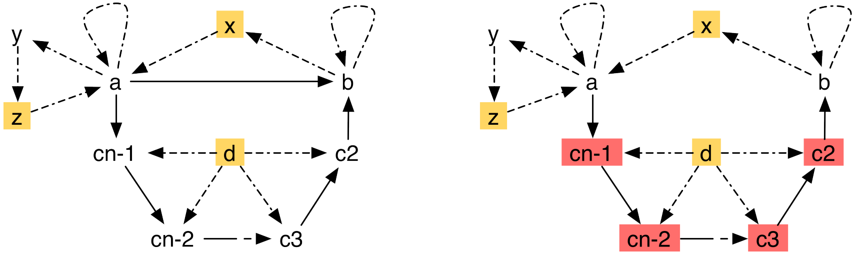
For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.



Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

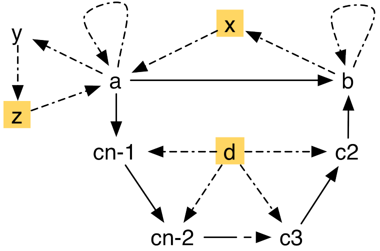


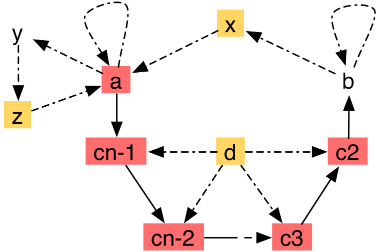
Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Strong Equivalence w.r.t. *cf2*

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

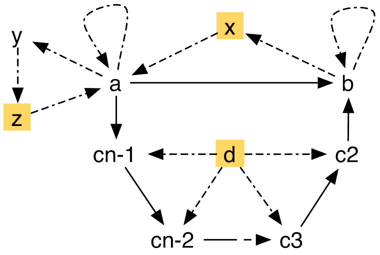


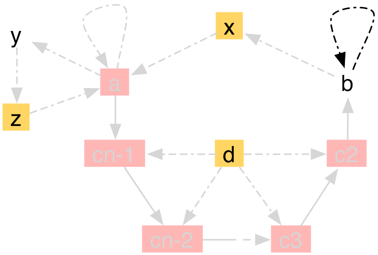


Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.





Let $E = \{d, x, z\}$, $E \in cf2(F \cup H)$ but $E \notin cf2(G \cup H)$.

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

- No matter which AFs $F \neq G$, one can always construct an H s.t. $cf2(F \cup H) \neq cf2(G \cup H)$;

SE w.r.t. *cf2* and *stage2*

Theorem

For any AFs F and G, $F \equiv_s^{cf2} G$ iff $F = G$.

- No matter which AFs $F \neq G$, one can always construct an H s.t. $cf2(F \cup H) \neq cf2(G \cup H)$;
- For *stage2* semantics also strong equivalence coincides with syntactic equivalence.

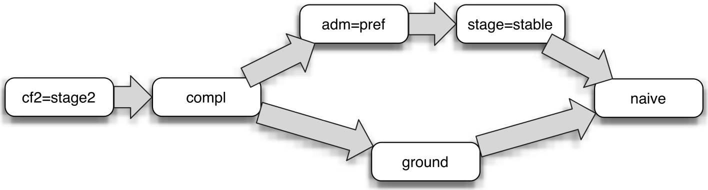
SE w.r.t. *cf2* and *stage2*

- No matter which AFs $F \neq G$, one can always construct an H s.t. $\mathit{cf2}(F \cup H) \neq \mathit{cf2}(G \cup H)$;
- For *stage2* semantics also strong equivalence coincides with syntactic equivalence.

Succinctness Property

An argumentation semantics σ satisfies the succinctness property or is maximal succinct iff no AF contains a redundant attack w.r.t. σ.

Comparing Semantics w.r.t. SE



Complexity Analysis

	Ver	*Cred*	*Skept*	$Exists^{\neg\emptyset}$
naive	in P	in P	in P	in P
stable	in P	NP-c	coNP-c	NP-c
cf2	in P	NP-c	coNP-c	in P
stage	coNP-c	Σ_2^P-c	Π_2^P-c	in P
stage2	coNP-c	Σ_2^P-c	Π_2^P-c	in P

Table: Computational complexity of naive-based semantics.

Reduction-based Approach

- Answer-set Programming (ASP) encodings for *cf2* and *stage2*.
- Saturation vs. `metasp` encodings for *stage2*.
- All encodings incorporated in the system ASPARTIX [Egly et al., 2010].

Direct Approach

- Labeling-based algorithms for *cf2* and *stage2*.

Web-Application

`http://rull.dbai.tuwien.ac.at:8080/ASPARTIX`

Conclusion

- Alternative characterization for *cf2* to avoid the recursive computation of sub-frameworks.
- *stage2* semantics overcomes problems of *cf2*.
- Strong equivalence w.r.t. *cf2* (resp. *stage2*) coincides with syntactic equivalence.
- Provided the missing complexity results for *cf2* (resp. *stage2*).
- Implementation in terms of ASP and labeling-based algorithms.

Future Work

- Further relations to other semantics like intertranslatability.
- Optimizations of ASP encodings.
- Development of appropriate instantiation methods for naive-based semantics.
- Other combinations of semantics in the alternative characterization, like $sem(F) = \{S \mid \sigma(F) \cap \tau([[F - \Delta_{F,S}]])\}$.

Baroni, P. and Giacomin, M. (2007).
On principle-based evaluation of extension-based argumentation semantics.
Artif. Intell., 171(10-15):675–700, 2007.

Baroni, P., Giacomin, M., and Guida, G. (2005).
Scc-recursiveness: A general schema for argumentation semantics.
Artif. Intell., 168(1-2):162–210.

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

Dvorák, W. and Gaggl, S. A. (2012)
Incorporating stage semantics in the scc-recursive schema for argumentation semantics.
In Proceedings of the 14th International Workshop on Non-Monotonic Reasoning (NMR 2012), 2012.

Dvorák, W. and Gaggl, S. A. (2012)
Computational aspects of cf2 and stage2 argumentation semantics.
In Bart Verheij, Stefan Szeider, and Stefan Woltran, editors, Proceedings of the 4th International Conference on Computational Models of Argument (COMMA 2012), volume 245 of Frontiers in Artificial Intelligence and Applications, pages 273–284. IOS Press, 2012.

Egly, U., Gaggl, S. A. and Woltran, S. (2010)
Answer-set programming encodings for argumentation frameworks.
Argument and Computation, 1(2):144–177, 2010.

Gaggl, S. A. and Woltran, S. (2012).
The cf2 argumentation semantics revisited.
Journal of Logic and Computation, 2012. doi: 10.1093/logcom/exs011.

Oikarinen, E. and Woltran, S. (2011)
Characterizing strong equivalence for argumentation frameworks.
Artif. Intell., 175(14–15):1985–2009, 2011.

Tractable Fragments

	$cf2$	$stage2$	$stable$	$stage$
$Cred_{\sigma}{}^{acycl}$	in P	in P	P-c	P-c
$Skept_{\sigma}{}^{acycl}$	in P	in P	P-c	P-c
$Cred_{\sigma}{}^{even-free}$	NP-c	coNP-h	P-c	Σ_2^P-c
$Skept_{\sigma}{}^{even-free}$	coNP-c	coNP-h	P-c	Π_2^P-c
$Cred_{\sigma}{}^{bipart}$	in P	in P	P-c	P-c
$Skept_{\sigma}{}^{bipart}$	in P	in P	P-c	P-c
$Cred_{\sigma}{}^{sym}$	in P	in P$/\Sigma_2^P*$	in P	in P$/\Sigma_2^P*$
$Skept_{\sigma}{}^{sym}$	in P	in P$/\Pi_2^P*$	in P	in P$/\Pi_2^P*$

Table: Complexity results for special AFs (* with self-attacking arguments).