Rejection in Abstract Argumentation: Harder Than Acceptance?

Johannes K. Fichte[1] Markus Hecher[2] Arne Meier[3] Yasir Mahmood [4]

[1] Linköping University, [2] Massachusetts Institute of Technology,
[3] Leibniz Universität Hannover, [4] Universität Paderborn

Workshop on Algorithms, Complexity and Logic (“Theorietag”), Univ. Jena, March 3rd, 2025

Presented work published at ECAI’24

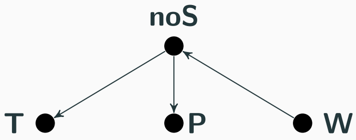
Abstract Argumentation (Dung 1995)

Image generated by DALL-E 3, text afterwards added

Formal Framework to

- resolve conflicts between arguments
- model 'the last un-attacked argument wins'
- abstract away details of how arguments are constructed

Example 1 (Going to the conference or not?)



noS ... noSubmission
T ... Travel
P ... Presentation
W ... Written up Paper

"Stable" extension is $E = \{W, T, P\}$.

Definition

Argumentation framework (AF) is a digraph $F = (A, R)$ where

- A is a set of arguments, and
- $R \subseteq A \times A$ is (directed) attack relation.

Semantics

A σ-Extension is a set $S \subseteq A$ of arguments such that:

semantics σ	properties
conflict-free	$(S \times S) \cap R = \emptyset$
stage	conflict-free and $\neg\exists$ conflict-free S' with $S_R^+ \subsetneq (S')_R^+$
admissible	conflict-free and $\forall s \in S$ s.t. $s' R s \, \exists s'' \in S$ s.t. $s'' R s'$
stable	admissible and every $s \notin S$ is attacked by some $s' \in S$
semi-stable	admissible and $\neg\exists$ admissible S' with $S_R^+ \subsetneq (S')_R^+$
complete	admissible and $\mathrm{def}(S) = S$

$S_R^+ := S \cup \{ a \mid (b, a) \in R, b \in S \}$, Set of all extensions: $\mathrm{Ext}_\sigma(F)$
$\mathrm{def}(S)$: set of all arguments defended by S

Known inclusions: $\mathrm{stable}(F) \subseteq \mathrm{semi\text{-}stable}(F) \subseteq \mathrm{compl}(F) \subseteq \mathrm{admissible}(F) \subseteq \mathrm{conflict\text{-}free}(F)$ and
$\mathrm{stable}(F) \subseteq \mathrm{stage}(F) \subseteq \mathrm{conflict\text{-}free}(F)$.

Problems of Interest

Problem: Consistency $\mathcal{S}$, short cons_σ
Input: AF $F = (A, R)$
Question: $\mathrm{Ext}_\sigma(F) \neq \emptyset$

Problem: Credulous $\mathcal{S}$-Reasoning, short cred_σ
Input: AF $F = (A, R)$, argument $s \in A$
Question: Is s contained in some σ-extension?

⋆ See (Dvorak & Dunne'17) for an overview on decision complexity results.
and (Fichte, Hecher, M'24) for an overview on counting complexity results.

The past: Extensions of AFs

AFs lack certain acceptance features. (Brewka & Woltran'10)

The past: Extensions of AFs

AFs lack certain acceptance features. (Brewka & Woltran'10)

ADFs: AF + acceptance conditions for arguments. (Brewka & Woltran'10)
$\leadsto$ Results in a certain locality.

The past: Extensions of AFs

AFs lack certain acceptance features. (Brewka & Woltran'10)

ADFs: AF + acceptance conditions for arguments. (Brewka & Woltran'10)
$\rightsquigarrow$ Results in a certain locality.

CAFs: AF + a propositional constraint over arguments. (Marquis et al.'06)
$\diamond$ No constraints for individual arguments.
$\diamond$ Variables of the constraint are arguments.

$\Rightarrow$ Acceptance and its complexity well studied.

The past: Extensions of AFs

AFs lack certain acceptance features. (Brewka & Woltran'10)

ADFs: AF + acceptance conditions for arguments. (Brewka & Woltran'10)
$\rightsquigarrow$ Results in a certain locality.

CAFs: AF + a propositional constraint over arguments. (Marquis et al.'06)
$\diamond$ No constraints for individual arguments.
$\diamond$ Variables of the constraint are arguments.

$\Rightarrow$ Acceptance and its complexity well studied.

What happens if we take rejection instead acceptance?

Contribution

TL;DR

Contributions

- Add rejection conditions (RC) to abstract argumentation frameworks
- Study (its influence on) its computational complexity

RCF

Rejection in Abstract Argumentation

Rejection Condition Frameworks

Idea

Model rejection at arguments.

- Enhance each argument in AFs by constraint (formula/logic program) called rejection condition (RC) collected in set C
- If argument in extension, then rejection condition needs to be invalidated

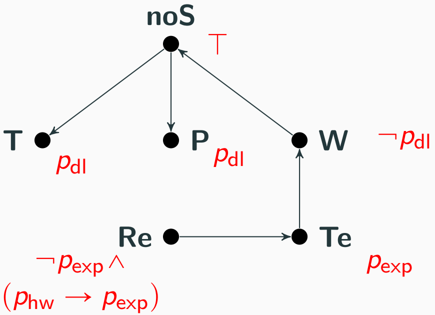
Definition (Rejection Condition)

Let $F = (A, R, C)$ be a RCF and σ be a semantics. $E \subseteq A$ is a σ-extension (of F) if

- $E \in \mathrm{Ext}_\sigma(A, R)$ (classical semantics hold)
- $E \cup \bigcup_{e \in E} C(e) \cup \bigcup_{a \in A \setminus E} \{\bot \leftarrow a\}$ is inconsistent (RCs invalidated)

Rejection Condition Frameworks (cont.)

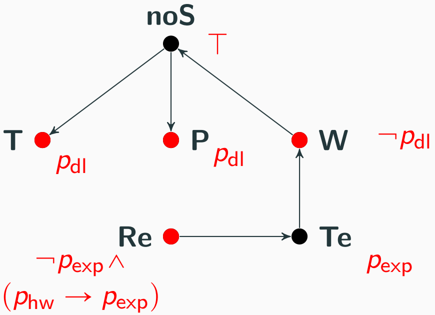
Example 2 (Bringing deadlines and teaching into the picture)



noS ... noSubmission
T ... Travel
Te ... Teaching
P ... Present
Re ... Research
W ... Written up Paper

Conditions (RC): p_{dl} ... deadline, p_{hw} ... hard working, p_{exp} ... experiments

Example 2 (Bringing deadlines and teaching into the picture)



noS ... noSubmission
T ... Travel
Te ... Teaching
P ... Present
Re ... Research
W ... Written up Paper

Conditions (RC): p_{dl} ... deadline, p_{hw} ... hard working, p_{exp} ... experiments

Stable extension: $\{\mathrm{W}, \mathrm{T}, \mathrm{P}, \mathrm{Re}\}$

Rejection conditions: $C(E) = \neg p_{dl} \wedge p_{dl} \wedge \neg p_{exp} \wedge (p_{hw} \rightarrow p_{exp}) \equiv \bot$ ✓

Our Results

RC	cons_{σ_0}	cons_{σ_1}	cons_{σ_2}	$\text{cred}_{\sigma_0,\sigma_1}$	cred_{σ_2}	
–	trivial	NP	trivial	NP	Σ_2^P	⋆
PL[**Simple**]	NP	NP	NP	NP	Σ_2^P	
PL[**Prop.**]	Σ_2^P	Σ_2^P	Σ_2^P	Σ_2^P	Σ_3^P	
ASP[**Tight**]	Σ_2^P	Σ_2^P	Σ_2^P	Σ_2^P	Σ_3^P	
ASP[**Disj**]	Σ_3^P	Σ_3^P	Σ_3^P	Σ_3^P	Σ_4^P	

$\sigma_0 \in \{$conflict-free, admissible, complete$\}$

$\sigma_1 \in \{$stabe$\}$

$\sigma_2 \in \{$semi-stable, stage$\}$

PL[Simple]: CNFs with variable set A

PL[Prop.]: CNFs

ASP[Tight]: cycle-free ASP graph

ASP[Disj]: Disjunctive ASPs

⋆ Overview in (Dvorak & Dunne'17)

Our Results (parameterized by treewidth) give tight bounds under ETH

RC	$\text{cons}_{\sigma_0,\sigma_1,\sigma_2}$	$\text{cred}_{\sigma_0,\sigma_1}$	cred_{σ_2}
PL[**Simple**]	$\exp(1, \Theta(tw(\mathcal{G}_F)))$	$\exp(1, \Theta(tw(\mathcal{G}_F)))$	$\exp(1, \Theta(tw(\mathcal{G}_F)))$
PL[**Prop.**]	$\exp(2, \Theta(tw(\mathcal{G}_F)))$	$\exp(2, \Theta(tw(\mathcal{G}_F)))$	$\exp(2, \Theta(tw(\mathcal{G}_F)))$
ASP[**Tight**]	$\exp(2, \Theta(tw(\mathcal{G}_F)))$	$\exp(2, \Theta(tw(\mathcal{G}_F)))$	$\exp(2, \Theta(tw(\mathcal{G}_F)))$
ASP[**Disj**]	$\exp(3, \Theta(tw(\mathcal{G}_F)))$	$\exp(3, \Theta(tw(\mathcal{G}_F)))$	$\exp(3, \Theta(tw(\mathcal{G}_F)))$

$\sigma_0 \in \{\text{conflict-free, admissible, complete}\}$

$\sigma_1 \in \{\text{stabe}\}$

$\sigma_2 \in \{\text{semi-stable, stage}\}$

PL[Simple]: CNFs with variable set A

PL[Prop.]: CNFs

ASP[Tight]: cycle-free ASP graph

ASP[Disj]: Disjunctive ASPs

$\exp(i, k) := 2^{2^{\cdot^{\cdot^{\cdot^{k}}}}}$, tower of exponentials of height i;

$\mathcal{G}_F$: graph of AF where directed edges are replaced by undirected ones.

Closing

Summary and Future Works

Conclusion

- Defined rejection instead of acceptance
- Insights into differences and interactions between rejection and acceptance
- Natural problems for 3rd and 4th level of PH (so rejection is presumably harder)
- Tight runtime bounds for treewidth as parameterization under ETH

Summary and Future Works

Conclusion

- Defined rejection instead of acceptance
- Insights into differences and interactions between rejection and acceptance
- Natural problems for 3rd and 4th level of PH (so rejection is presumably harder)
- Tight runtime bounds for treewidth as parameterization under ETH

Future Works

- Higher-order, e.g., non-ground programs of fixed arity, FO
- Implementation / Empirical evaluation
- Enhance ASP with Argumentation (vice versa perspective)
- Comparison with recent work (Heine, Ulbricht, KR'24)

Summary and Future Works

Conclusion

- Defined rejection instead of acceptance
- Insights into differences and interactions between rejection and acceptance
- Natural problems for 3rd and 4th level of PH (so rejection is presumably harder)
- Tight runtime bounds for treewidth as parameterization under ETH

Future Works

- Higher-order, e.g., non-ground programs of fixed arity, FO
- Implementation / Empirical evaluation
- Enhance ASP with Argumentation (vice versa perspective)
- Comparison with recent work (Heine, Ulbricht, KR'24)

Thank you. Questions?

Research supported by DFG, ELLIIT, FWF, GFF, MKW NRW, ÖAW, WWTF.

References

- Phan Minh Dung, 1995: On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games.
- Marquis et al., 2006: Constrained Argumentation Frameworks. KR, 2006
- Brewka & Woltran, 2010: Abstract Dialectical Frameworks. KR, 2010
- Dvorak & Dunne, 2017: Computational Problems in Formal Argumentation and their Complexity.
- Fichte et al., JAIR 2024: Counting Complexity for Reasoning in Abstract Argumentation

ASP: Syntax + Semantics

Definition

U a universe of prop. atoms.

A literal: an atom $a \in U$ or $\neg a$.

A rule r: $a_1 \vee \ldots \vee a_l \leftarrow b_1, \ldots, b_n, \sim c_1, \ldots, \sim c_m$
where $a_1, \ldots, a_l, b_1, \ldots, b_n, c_1, \ldots, c_m \in U$ and $l, n, m \in \mathbb{N}$

Or, $H(r) \leftarrow B^+(r), B^-(r)$.

A program $\mathcal{P}$: a set of rules.

A set $M \subseteq U$ satisfies a rule r if $(H(r) \cup B^-(r)) \cap M \neq \emptyset$ or $B^+(r) \backslash M \neq \emptyset$.

M is a model of $\mathcal{P}$ if it satisfies every $r \in \mathcal{P}$.