

Current Trends in Argumentation Dynamics

Jean-Guy Mailly

EASSS 2024 – 21/08/2024

Université Toulouse Capitole, IRIT

Who am I? What do I do?



Me, at KR'23 in Rhodes



Toulouse

- 2012-15: PhD in Computer Science
- 2015-16: Postdoc at TU Wien
- 2016-24: Associate Professor at Université Paris Cité
- 2024-...: Junior Professor at Université Toulouse Capitole

- Knowledge representation and reasoning
 - Somewhere between formal logic and computer engineering
- More precisely, argumentation
 - Connections to AAMAS

Feel free to...

- Ask questions during the talk
- Ask questions after the talk (jean-guy.mailly@irit.fr)
 - I can share the slides if you want
- Stay in touch :)
 - <https://www.linkedin.com/in/jeanguymailly/>

Table of contents

1. Background: Abstract Argumentation
 - Dung's Framework
 - SAT-based Computation for Argumentation
 - What is Argumentation Dynamics?
2. Dynamic Computation
 - The Subgraph-based Approach
 - The Iterative SAT-based Approach
3. Extension Enforcement
 - How to modify arguments acceptability?
 - Argumentation under Uncertainty
 - Argumentation Dynamics under Uncertainty
4. Application to a Multi-Agent Scenario: Automated Negotiation
5. Conclusion

Outline

Background: Abstract Argumentation

Dung's Framework

SAT-based Computation for Argumentation

What is Argumentation Dynamics?

Dynamic Computation

The Subgraph-based Approach

The Iterative SAT-based Approach

Extension Enforcement

How to modify arguments acceptability?

Argumentation under Uncertainty

Argumentation Dynamics under Uncertainty

Application to a Multi-Agent Scenario: Automated Negotiation

Conclusion

What is Argumentation?

- Argumentation is an important part of human reasoning
 - Justifying one's own beliefs/decisions
 - Convincing someone else to believe something/do something
 - Analysing conflicting pieces of information
- Formal argumentation in AI studies:
 - Modeling of arguments and their relationships
 - Acceptability of arguments
 - Protocols for several agents using arguments in dialogues
- Two families of formal models
 - Structured/logic-based frameworks
 - Abstract frameworks

What is Argumentation?

- Argumentation is an important part of human reasoning
 - Justifying one's own beliefs/decisions
 - Convincing someone else to believe something/do something
 - Analysing conflicting pieces of information
- Formal argumentation in AI studies:
 - Modeling of arguments and their relationships
 - Acceptability of arguments
 - Protocols for several agents using arguments in dialogues
- Two families of formal models
 - Structured/logic-based frameworks
 - Abstract frameworks

What is Argumentation?

- Argumentation is an important part of human reasoning
 - Justifying one's own beliefs/decisions
 - Convincing someone else to believe something/do something
 - Analysing conflicting pieces of information
- Formal argumentation in AI studies:
 - Modeling of arguments and their relationships
 - Acceptability of arguments
 - Protocols for several agents using arguments in dialogues
- Two families of formal models
 - Structured/logic-based frameworks
 - Abstract frameworks

Abstract Argumentation

Dung's Abstract Argumentation Framework

Argumentation Framework (**AF** for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments

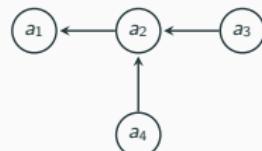
Abstract Argumentation

Dung's Abstract Argumentation Framework

Argumentation Framework (**AF** for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments
- Example:
 - a_1 : (John) "I'm hungry, let's go to this restaurant."
 - a_2 : (Yoko) "I've seen on Tripadvisor that the food is bad, let's go somewhere else."
 - a_3 : (John) "These grades are old, and there's a new chef, so it should be better now."
 - a_4 : (John) "Moreover, the other restaurants in the streets are closed."

$$F = \langle A, R \rangle \text{ with}$$
$$A = \{a_1, a_2, a_3, a_4\},$$
$$R = \{(a_2, a_1), (a_3, a_2), (a_4, a_2)\}$$



Abstract Argumentation

Dung's Abstract Argumentation Framework

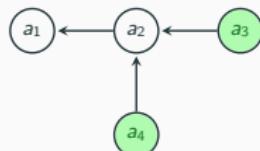
Argumentation Framework (**AF** for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments
- Example:
 - a_1 : (John) "I'm hungry, let's go to this restaurant."
 - a_2 : (Yoko) "I've seen on Tripadvisor that the food is bad, let's go somewhere else."
 - a_3 : (John) "These grades are old, and there's a new chef, so it should be better now."
 - a_4 : (John) "Moreover, the other restaurants in the streets are closed."

$$F = \langle A, R \rangle \text{ with}$$

$$A = \{a_1, a_2, a_3, a_4\},$$

$$R = \{(a_2, a_1), (a_3, a_2), (a_4, a_2)\}$$



P. M. 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)

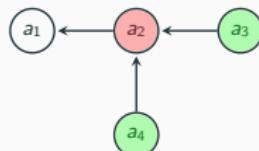
Abstract Argumentation

Dung's Abstract Argumentation Framework

Argumentation Framework (AF for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments
- Example:
 - a_1 : (John) "I'm hungry, let's go to this restaurant."
 - a_2 : (Yoko) "I've seen on Tripadvisor that the food is bad, let's go somewhere else."
 - a_3 : (John) "These grades are old, and there's a new chef, so it should be better now."
 - a_4 : (John) "Moreover, the other restaurants in the streets are closed."

$$F = \langle A, R \rangle \text{ with}$$
$$A = \{a_1, a_2, a_3, a_4\},$$
$$R = \{(a_2, a_1), (a_3, a_2), (a_4, a_2)\}$$



P. M. 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)

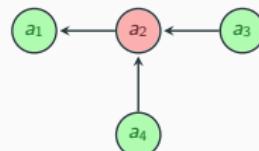
Abstract Argumentation

Dung's Abstract Argumentation Framework

Argumentation Framework (**AF** for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments
- Example:
 - a_1 : (John) "I'm hungry, let's go to this restaurant."
 - a_2 : (Yoko) "I've seen on Tripadvisor that the food is bad, let's go somewhere else."
 - a_3 : (John) "These grades are old, and there's a new chef, so it should be better now."
 - a_4 : (John) "Moreover, the other restaurants in the streets are closed."

$$F = \langle A, R \rangle \text{ with}$$
$$A = \{a_1, a_2, a_3, a_4\},$$
$$R = \{(a_2, a_1), (a_3, a_2), (a_4, a_2)\}$$



P. M. 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)

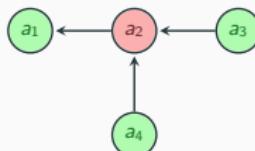
Abstract Argumentation

Dung's Abstract Argumentation Framework

Argumentation Framework (AF for short): $F = \langle A, R \rangle$ where

- A is a set of arguments
- $R \subseteq A \times A$ represents attacks between arguments
- Example:
 - a_1 : (John) "I'm hungry, let's go to this restaurant."
 - a_2 : (Yoko) "I've seen on Tripadvisor that the food is bad, let's go somewhere else."
 - a_3 : (John) "These grades are old, and there's a new chef, so it should be better now."
 - a_4 : (John) "Moreover, the other restaurants in the streets are closed."
- It doesn't work if there are cycles → various semantics to remedy this issue

$$F = \langle A, R \rangle \text{ with}$$
$$A = \{a_1, a_2, a_3, a_4\},$$
$$R = \{(a_2, a_1), (a_3, a_2), (a_4, a_2)\}$$



P. M. 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)

Extension

Given $F = \langle A, R \rangle$, an extension is a set of jointly acceptable arguments

Extension

Given $F = \langle A, R \rangle$, an extension is a set of jointly acceptable arguments

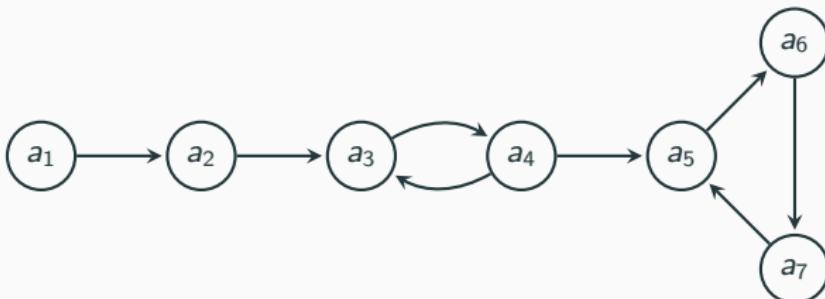
Extension-based Semantics

Given $F = \langle A, R \rangle$, $S \subseteq A$ is

- *conflict-free (cf)* if there is no $a, b \in S$ s.t. $(a, b) \in R$
- *admissible (ad)* if $S \in \text{cf}(F)$ and S defends all its elements
- *stable (st)* if $S \in \text{cf}(F)$ and S attacks each argument in $A \setminus S$
- *complete (co)* if $S \in \text{ad}(F)$ and S doesn't defend any argument in $A \setminus S$
- *preferred (pr)* if S is \subseteq -maximal in $\text{ad}(F)$
- *grounded (gr)* if S is \subseteq -minimal in $\text{co}(F)$

P. M. 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)

Example: Semantics Comparison



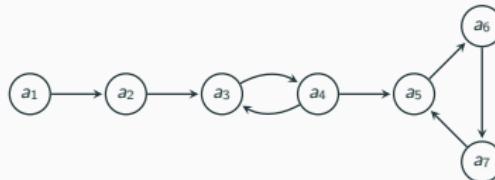
Semantics σ	σ -extensions	$cred_\sigma$	$skep_\sigma$
grounded	$\{\{a_1\}\}$	$\{a_1\}$	$\{a_1\}$
stable	$\{\{a_1, a_4, a_6\}\}$	$\{a_1, a_4, a_6\}$	$\{a_1, a_4, a_6\}$
preferred	$\{\{a_1, a_4, a_6\}, \{a_1, a_3\}\}$	$\{a_1, a_3, a_4, a_6\}$	$\{a_1\}$
complete	$\{\{a_1, a_4, a_6\}, \{a_1, a_3\}, \{a_1\}\}$	$\{a_1, a_3, a_4, a_6\}$	$\{a_1\}$

- $cred_\sigma(F) = \cup_{S \in \sigma(F)} S$: credulously accepted arguments
- $skep_\sigma(F) = \cap_{S \in \sigma(F)} S$: skeptically accepted arguments

- 3-valued representation of extensions
- Given $F = \langle A, R \rangle$ an AF, and S a σ -extension,
 - $L_S(a) = IN$ iff $a \in S$,
 - $L_S(a) = OUT$ iff $\exists b \in S$ s.t. $(b, a) \in R$,
 - $L_S(a) = UNDEC$ otherwise

Labellings

- 3-valued representation of extensions
- Given $F = \langle A, R \rangle$ an AF, and S a σ -extension,
 - $L_S(a) = \text{IN}$ iff $a \in S$,
 - $L_S(a) = \text{OUT}$ iff $\exists b \in S$ s.t. $(b, a) \in R$,
 - $L_S(a) = \text{UNDEC}$ otherwise



Semantics σ	σ -labellings
grounded	$\{\{a_1, a_2, a_3, a_3, a_5, a_6, a_7\}\}$
stable	$\{\{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}\}$
preferred	$\{\{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}, \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}\}$
complete	$\{\{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}, \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}, \{a_1, a_2, a_3, a_3, a_5, a_6, a_7\}\}$

<https://pyarg.npai.science.uu.nl>

The screenshot shows the PyArg web interface. At the top, there is a navigation bar with tabs: "Generate", "Visualise", "Learn", "Algorithms", "Applications", and "Colorblind mode". Below the navigation bar, the main content area has a title "Welcome to PyArg!". A sub-section below it says "This is a Python package and web interface for solving various computational argumentation problems." It also includes a contact link "Daphne Odekerken (D.Odekerken@UU.nl)". On the left side, there is a sidebar with sections for "Contributors" (listing names like Matti Berthold, AnneMarie Borg, Jonas Klein, Bertram Ludäscher, Daphne Odekerken, and Yilin Xia) and "Abstract", "ASPiC+", "ABA", and "IAF".

Visualisation of abstract argumentation frameworks

Abstract Argumentation Framework

Generate random Open existing AF

Arguments

- A
- B
- C

Attacks

- (A,B)
- (B,C)
- (C,B)
- (B,A)

Filename edited_af . JSON Download

Evaluation

...

Default visualisation Layered visualisation

```
graph LR; A((A)) --> B((B)); B --> C((C)); C --> A;
```

Visualisation of abstract argumentation frameworks

Abstract Argumentation Framework

Evaluation

Semantics Complete

Evaluation strategy Credulous

The extension(s): (B) (A, C)

The accepted argument(s): A B C

Click on the extension/argument buttons to display the corresponding argument(s) in the graph.

Explanation

Default visualisation Layered visualisation

```
graph LR; A((A)) --> A; A --> B((B)); A <--> B; B --> C((C)); B <--> C;
```

PyArg: Evaluation (2)

Visualisation of abstract argumentation frameworks

Abstract Argumentation Framework ▾

Evaluation ▾

Semantics Complete ▾
Evaluation strategy Credulous ▾

The extension(s): { } {B} {A, C}

The accepted argument(s): A B C

Click on the extension/argument buttons to display the corresponding argument(s) in the graph.

Explanation ▾

Default visualisation

Layered visualisation

```
graph LR; A((A)) <-->|2x| B((B)); B <-->|1x| C((C))
```

PyArg: Evaluation (3)

Visualisation of abstract argumentation frameworks

Abstract Argumentation Framework	
Evaluation	
Semantics	Complete
Evaluation strategy	Credulous
The extension(s):	
{ } {B} {A, C}	
The accepted argument(s):	
A B C	
Click on the extension/argument buttons to display the corresponding argument(s) in the graph.	
Explanation	

Default visualisation Layered visualisation



Computational Complexity

- Reasoning with AFs is generally hard

Problem	Grounded	Stable	Preferred	Complete
$\sigma\text{-Exist}$	Trivial	NP-c	Trivial	Trivial
$\sigma\text{-Exist}^{NT}$	L	NP-c	NP-c	NP-c
$\sigma\text{-Verif}$	P-c	L	coNP-c	L
$\sigma\text{-Cred}$	P-c	NP-c	NP-c	NP-c
$\sigma\text{-Skep}$	P-c	coNP-c	Π_2^P -c	P-c

- $\sigma\text{-Exist}$: Given F , is $\sigma(F) \neq \emptyset$?
- $\sigma\text{-Exist}^{NT}$: Given F , is $\sigma(F) \neq \emptyset$ s.t. F has at least one non-empty extension?
- $\sigma\text{-Verif}$: Given F and S , is $S \in \sigma(F)$?
- $\sigma\text{-Cred}$: Given F and a , is $a \in \text{cred}_\sigma(F)$?
- $\sigma\text{-Skep}$: Given F and a , is $a \in \text{skep}_\sigma(F)$?

W. Dvorák, P. E.Dunne: Computational Problems in Formal Argumentation and their Complexity. Handbook of Formal Argumentation: 631–688 (2018)

Propositional Encoding of Semantics

Approach proposed in (Besnard and Doutre 04). Intuition:

- Encoding arguments' acceptance in Boolean variables
- Define a formula such that each model corresponds to an extension

Propositional Encoding of Semantics

Approach proposed in (Besnard and Doutre 04). Intuition:

- Encoding arguments' acceptance in Boolean variables
- Define a formula such that each model corresponds to an extension

Logical Encoding of Stable Semantics

For $F = \langle A, R \rangle$, $S \subseteq A$ is a stable extension of F iff S is a model of

$$\phi_{st}(F) = \bigwedge_{a \in A} (a \leftrightarrow \bigwedge_{(b,a) \in R} \neg b)$$

Similar encodings exist for conflict-freeness, admissibility and the complete semantics.

P. Besnard, S. Doutre: Checking the acceptability of a set of arguments. NMR 2004: 59-64

- For $\sigma \in \{cf, ad, co, st\}$, $\text{mod}(\phi_\sigma(F)) = \sigma(F)$
 - Compute one/each extension = compute one/each model
 - Decide the credulous acceptability of a = check if $\phi_\sigma(F) \wedge a$ is SAT
 - Decide the skeptical acceptability of a = check if $\phi_\sigma(F) \wedge \neg a$ is UNSAT
- For $\sigma = gr$, computation is polynomial (can be done with unit propagation over $\phi_{co}(F)$)
- For $\sigma = pr$, need to use other techniques, e.g. CEGAR or MSS extraction

- Since 2015, the International Competition on Computational Models of Argumentation (ICCMA) evaluates the best solvers for argumentation problems
 - <https://argumentationcompetition.org>
 - Many available solvers
- μ -toskia: in C++
 - Winner of ICCMA 2019
 - <https://bitbucket.org/andreasniskanen/mu-toskia/src/master/>
- Crustabri: in Rust
 - Winner of several tracks at ICCMA 2023 (9 sub-tracks over 13 in the main track, and 3 sub-tracks over 3 in the dynamic track)
 - <https://www.cril.univ-artois.fr/software/crustabri/>
- pygarg : not an ICCMA solver, open-source Python implementation of the SAT-based algorithms from Crustabri
 - pip install pygarg
 - <https://github.com/jgmailly/pygarg>

Andreas Niskanen, Matti Järvisalo: μ -toskia: An Efficient Abstract Argumentation Reasoner. KR 2020: 800-804

J.-M. Lagniez, E. Lonca, J.-G. Mailly: A SAT-based Approach for Argumentation Dynamics. AAMAS 2024

J.-G. Mailly: pygarg: A Python engine for argumentation. Argument. Comput. 2024

If you like challenges, ICCMA 2025 will be announced soon (Sep. 17th, during SAFA 2024), and participation is open to everyone :)

<http://safa2024.argumentationcompetition.org>

<https://argumentationcompetition.org/2025/index.html>

Two Aspects of Argumentation Dynamics

- Dynamic re-computation when an update occurs: when some arguments/attacks are added/deleted, can we re-compute the acceptability of arguments without running (naively) algorithms from scratch?

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ \text{Acceptability of } a \in A \\ \text{New argument } x \\ \text{and attacks } R' \text{ from } x \end{array} \right\} \implies \text{Acceptability of } a \text{ in } F' = \langle A \cup \{x\}, R \cup R' \rangle$$

Two Aspects of Argumentation Dynamics

- Dynamic re-computation when an update occurs: when some arguments/attacks are added/deleted, can we re-compute the acceptability of arguments without running (naively) algorithms from scratch?

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ \text{Acceptability of } a \in A \\ \text{New argument } x \\ \text{and attacks } R' \text{ from } x \end{array} \right\} \implies \text{Acceptability of } a \text{ in } F' = \langle A \cup \{x\}, R \cup R' \rangle$$

- Belief change/strategic aspect: How to change (minimally?) an AF in order to satisfy some property (e.g. regarding arguments acceptability)

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ \text{Constraint} \end{array} \right\} \implies F' = \langle A', R' \rangle \text{ which satisfies the constraint}$$

Outline

Background: Abstract Argumentation

Dung's Framework

SAT-based Computation for Argumentation

What is Argumentation Dynamics?

Dynamic Computation

The Subgraph-based Approach

The Iterative SAT-based Approach

Extension Enforcement

How to modify arguments acceptability?

Argumentation under Uncertainty

Argumentation Dynamics under Uncertainty

Application to a Multi-Agent Scenario: Automated Negotiation

Conclusion

Two main approaches

- When an AF is updated (addition/deletion of arguments/attacks), can we re-compute the acceptability of arguments without re-starting from scratch?
- Two approaches in the literature:
 - Identify the part of the graph which is impacted by the update, and re-compute only for this (smaller) part
 - Use incremental SAT solving to keep track of what the SAT solver has learnt in previous computation steps

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ \text{Acceptability of } a \in A \\ \text{New argument } x \\ \text{and attacks } R' \text{ from } x \end{array} \right\} \implies \text{Acceptability of } a \text{ in } F' = \langle A \cup \{x\}, R \cup R' \rangle$$

The Subgraph-based Approach: Algorithm

Input:

- AF $F_0 = \langle A_0, R_0 \rangle$
- Attack update $u = \star(a, b)$, $\star \in \{+, -\}$
- Semantics σ
- Extension $E_0 \in \sigma(F_0)$

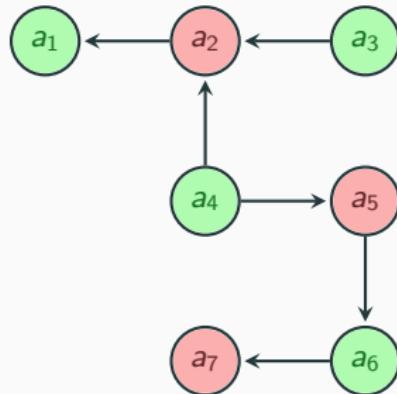
Functions used:

- $I(u, A_0, E_0)$ returns the set of arguments influenced by the update
- $R(U, A_0, E_0)$ returns the reduced AF
- $solve_\sigma(F)$ returns a σ -extension of an AF F if it exists, \perp otherwise

```
 $S = I(u, A_0, E_0)$ 
if  $S = \emptyset$  then
    return  $E_0$ 
else
     $F_1 = R(u, A_0, E_0)$ 
     $E_1 = solve_\sigma(F_1)$ 
    if  $E_1 \neq \perp$  then
        return  $(E_0 \setminus S) \cup E_1$ 
    else
        return  $solve_\sigma(u(A_0))$ 
    end if
end if
```

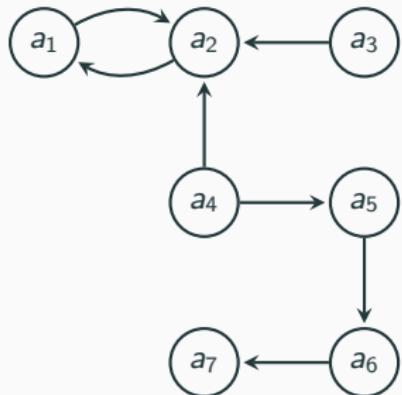
Gianvincenzo Alfano, Sergio Greco, Francesco Parisi: Efficient Computation of Extensions for Dynamic Abstract Argumentation Frameworks: An Incremental Approach. IJCAI 2017: 49-55

The Subgraph-based Approach: Example



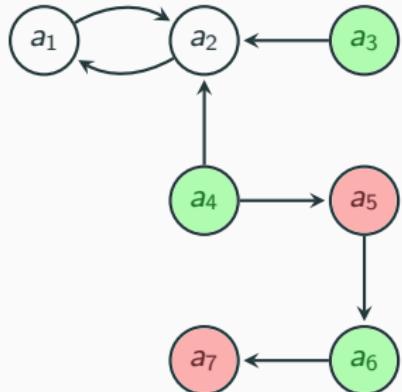
- $F_0 = \langle A, R \rangle$, $\sigma = st$,
 $E_0 = \{a_1, a_3, a_4, a_6\}$

The Subgraph-based Approach: Example



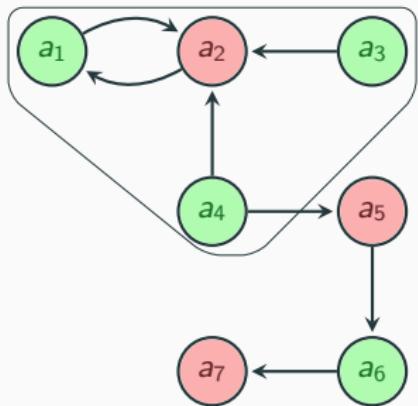
- $F_0 = \langle A, R \rangle$, $\sigma = st$,
 $E_0 = \{a_1, a_3, a_4, a_6\}$
- $u = +(a_1, a_2)$

The Subgraph-based Approach: Example



- $F_0 = \langle A, R \rangle, \sigma = st,$
 $E_0 = \{a_1, a_3, a_4, a_6\}$
- $u = +(a_1, a_2)$
- Influenced set
 $I(u, A_0, E_0) = \{a_1, a_2\}$

The Subgraph-based Approach: Example



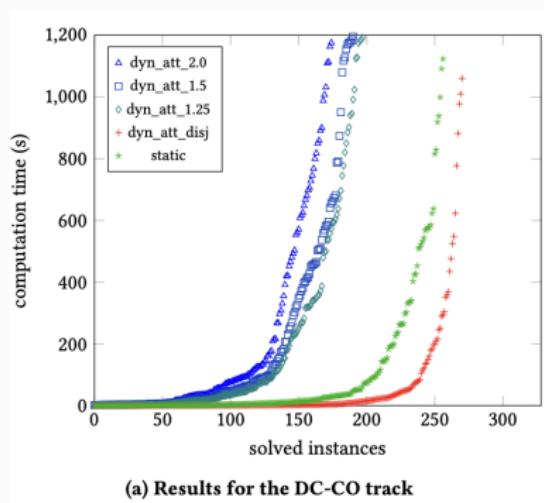
- $F_0 = \langle A, R \rangle$, $\sigma = st$,
 $E_0 = \{a_1, a_3, a_4, a_6\}$
- $u = +(a_1, a_2)$
- Influenced set
 $I(u, A_0, E_0) = \{a_1, a_2\}$
- Re-computation is localized to a smaller graph $R(U, A_0, E_0)$

The Subgraph-based Approach: Other Work

- In structured argumentation: Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, Gerardo Ignacio Simari, Guillermo Ricardo Simari: An Incremental Approach to Structured Argumentation over Dynamic Knowledge Bases. KR 2018: 78-87
- In bipolar AFs: Gianvincenzo Alfano, Sergio Greco, Francesco Parisi: A meta-argumentation approach for the efficient computation of stable and preferred extensions in dynamic bipolar argumentation frameworks. Intelligenza Artificiale 12(2): 193-211 (2018)
- For arguments skeptical acceptability: Gianvincenzo Alfano, Sergio Greco, Francesco Parisi: An Efficient Algorithm for Skeptical Preferred Acceptance in Dynamic Argumentation Frameworks. IJCAI 2019: 18-24
- In higher-order bipolar AFs: Gianvincenzo Alfano, Andrea Cohen, Sebastian Gottifredi, Sergio Greco, Francesco Parisi, Guillermo Ricardo Simari: Credulous acceptance in high-order argumentation frameworks with necessities: An incremental approach. Artif. Intell. 333: 104159 (2024)

Incremental SAT-based Computation

- Incremental SAT-based approach implemented in Crustabri, won all sub-tracks of the dynamic track at ICCMA 2023
- Idea: use the assumption mechanism of SAT solvers over variables representing the disjunction of attackers. Allows to easily update the set of attacks and the set of arguments



- Red: Crustabri approach
- Green: re-computation from scratch
- Blue: various versions of μ -toksia approach

- Future work: combine our approach with “subgraph-based” approach

Outline

Background: Abstract Argumentation

Dung's Framework

SAT-based Computation for Argumentation

What is Argumentation Dynamics?

Dynamic Computation

The Subgraph-based Approach

The Iterative SAT-based Approach

Extension Enforcement

How to modify arguments acceptability?

Argumentation under Uncertainty

Argumentation Dynamics under Uncertainty

Application to a Multi-Agent Scenario: Automated Negotiation

Conclusion

Extension Enforcement: Intuition

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ E \subseteq A \\ \text{should be (part of)} \\ \text{an extension} \end{array} \right\} \Rightarrow F' = \langle A', R' \rangle \text{ with } E \\ \text{(part of) an extension}$$

Different approaches

- (Baumann and Brewka) No change of the existing attacks, new arguments and attacks can be added
- (Coste-Marquis et al) The existing attacks can be modified
- (Doutre and Mailly) The semantics can be modified

R. Baumann, G. Brewka: Expanding Argumentation Frameworks: Enforcing and Monotonicity Results. COMMA 2010: 75-86

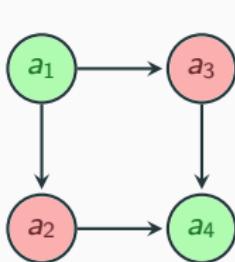
S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

S. Doutre, J.-G. Mailly: Semantic Change and Extension Enforcement in Abstract Argumentation. SUM 2017: 194-207

Example of Strong Enforcement

R. Baumann, G. Brewka: Expanding Argumentation Frameworks: Enforcing and Monotonicity Results. COMMA 2010: 75-86

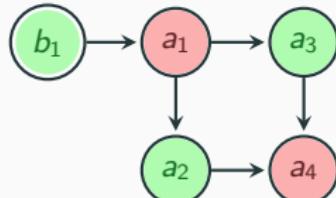
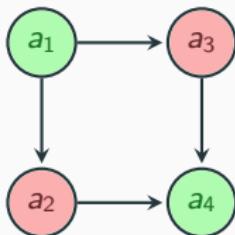
- How to enforce $E = \{a_2, a_3\}$ in F ?



Example of Strong Enforcement

R. Baumann, G. Brewka: Expanding Argumentation Frameworks: Enforcing and Monotonicity Results. COMMA 2010: 75-86

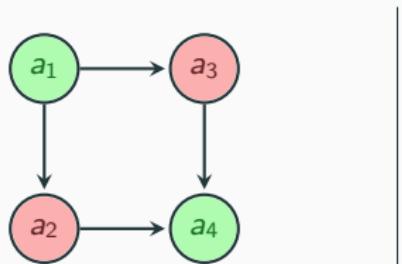
- How to enforce $E = \{a_2, a_3\}$ in F ?



Example of Argument-fixed Enforcement

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

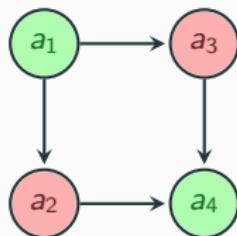
- How to enforce $E = \{a_2, a_3\}$ in F ?



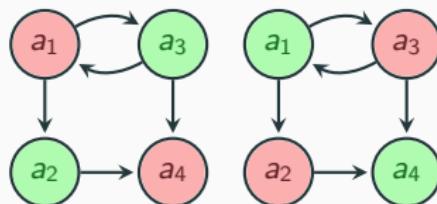
Example of Argument-fixed Enforcement

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

- How to enforce $E = \{a_2, a_3\}$ in F ?



Two extensions:



- Example: stable semantics
- Idea: generalize the propositional encoding to take into account the attack relation

$$\phi_{st}(F) = \bigwedge_{a \in A} \left(a \leftrightarrow \bigwedge_{(b,a) \in R} \neg b \right) \quad \mid \quad \phi'_{st}(A) = \bigwedge_{a \in A} \left(acc_a \leftrightarrow \bigwedge_{b \in A} (att_{b,a} \rightarrow \neg acc_b) \right)$$

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

- Example: stable semantics
- Idea: generalize the propositional encoding to take into account the attack relation

$$\phi_{st}(F) = \bigwedge_{a \in A} \left(a \leftrightarrow \bigwedge_{(b,a) \in R} \neg b \right) \quad \mid \quad \phi'_{st}(A) = \bigwedge_{a \in A} \left(acc_a \leftrightarrow \bigwedge_{b \in A} (att_{b,a} \rightarrow \neg acc_b) \right)$$

- If $F = \langle A, R \rangle$ is known, $\phi'_{st}(A) \wedge \bigwedge_{(a,b) \in R} att_{a,b} \wedge \bigwedge_{(a,b) \notin R} \neg att_{a,b}$ corresponds to $\phi_{st}(F)$

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

- Example: stable semantics
- Idea: generalize the propositional encoding to take into account the attack relation

$$\phi_{st}(F) = \bigwedge_{a \in A} \left(a \leftrightarrow \bigwedge_{(b,a) \in R} \neg b \right) \quad \mid \quad \phi'_{st}(A) = \bigwedge_{a \in A} \left(acc_a \leftrightarrow \bigwedge_{b \in A} (att_{b,a} \rightarrow \neg acc_b) \right)$$

- If $F = \langle A, R \rangle$ is known, $\phi'_{st}(A) \wedge \bigwedge_{(a,b) \in R} att_{a,b} \wedge \bigwedge_{(a,b) \notin R} \neg att_{a,b}$ corresponds to $\phi_{st}(F)$
- To enforce, $E \subseteq A$, search a model of $\phi'_{st}(A) \wedge \bigwedge_{a \in E} acc_a \wedge \bigwedge_{a \notin E} \neg acc_a$: the values of the $att_{a,b}$ variables correspond to an AF

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

Minimal Enforcement as PB and MaxSAT

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ E \subseteq A \\ \text{should be (part of)} \\ \text{an extension} \end{array} \right\} \implies \begin{array}{l} F' = \langle A', R' \rangle \text{ with } E \\ (\text{part of}) \text{ an extension and} \\ F' \text{ as close as possible to } F \end{array}$$

Minimal Enforcement as PB and MaxSAT

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ E \subseteq A \\ \text{should be (part of)} \\ \text{an extension} \end{array} \right\} \implies \begin{array}{l} F' = \langle A', R' \rangle \text{ with } E \\ (\text{part of}) \text{ an extension and} \\ F' \text{ as close as possible to } F \end{array}$$

- (Coste-Marquis et al) translate the SAT encoding in pseudo-Boolean constraints, optimize w.r.t. an objective function (distance between F and F')

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

Minimal Enforcement as PB and MaxSAT

$$\left. \begin{array}{l} F = \langle A, R \rangle \\ E \subseteq A \\ \text{should be (part of)} \\ \text{an extension} \end{array} \right\} \implies \begin{array}{l} F' = \langle A', R' \rangle \text{ with } E \\ (\text{part of}) \text{ an extension and} \\ F' \text{ as close as possible to } F \end{array}$$

- (Coste-Marquis et al) translate the SAT encoding in pseudo-Boolean constraints, optimize w.r.t. an objective function (distance between F and F')
- (Wallner et al) translate into partial MaxSAT: the SAT encoding of enforcement corresponds to hard clauses, and unit clauses corresponding to the $att_{a,b}$ variables correspond to the soft clauses. CEGAR algorithm for second level of the polynomial hierarchy
 - Available software:
<https://bitbucket.org/andreasniskanen/pakota/src/master/>
- Scalability: up to 350 arguments for NP-complete problems, 200 arguments for Σ_2^P -complete problems ($\simeq 10$ seconds for most instances)

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, P. Marquis: Extension Enforcement in Abstract Argumentation as an Optimization Problem. IJCAI 2015: 2876-2882

J. P. Wallner, A. Niskanen, M. Järvisalo: Complexity Results and Algorithms for Extension Enforcement in Abstract Argumentation. J. Artif. Intell. Res. 60: 1-40 (2017)

Incomplete Argumentation Framework (IAF)

$I = \langle A, A^?, R, R^? \rangle$ where

- $A, A^?$ are disjoint sets of arguments
- $R, R^?$ are disjoint sets of attacks over $A \cup A^?$

such that

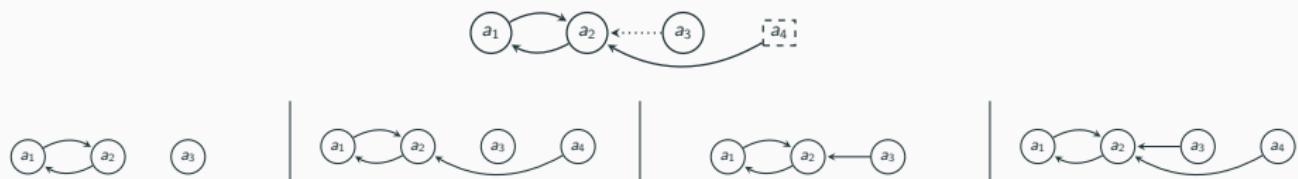
- A, R are certain arguments and attacks
- $A^?, R^?$ are uncertain arguments and attacks



J.-G. Mailly: Yes, no, maybe, I don't know: Complexity and application of abstract argumentation with incomplete knowledge. Argument Comput. 13(3): 291-324 (2022)

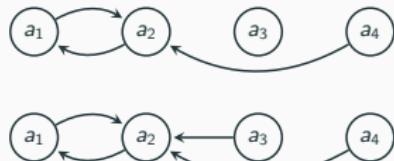
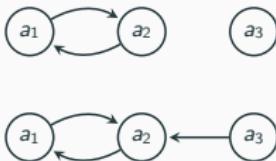
Completions of an IAF

Completions = AFs compatible with the incomplete knowledge encoded in the IAF
 \simeq possible worlds



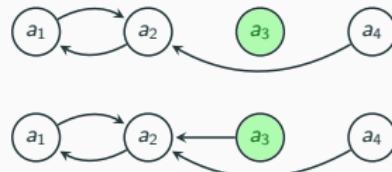
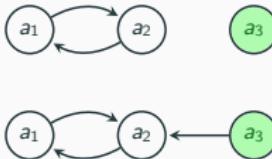
Possible and Necessary Reasoning

- Possible reasoning: some property is true for some completion of the IAF
- Necessary reasoning: some property is true for each completion of the IAF



Possible and Necessary Reasoning

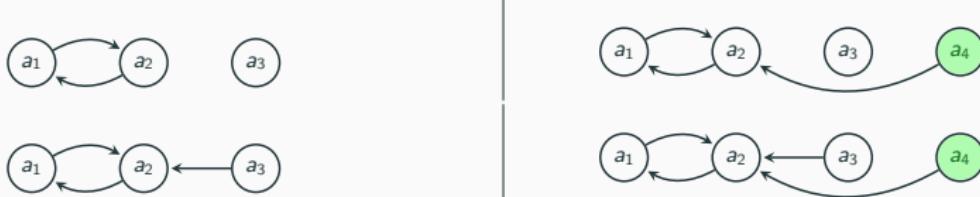
- Possible reasoning: some property is true for some completion of the IAF
- Necessary reasoning: some property is true for each completion of the IAF



- a_3 is skeptically accepted in each completion \rightarrow necessarily skeptically accepted

Possible and Necessary Reasoning

- Possible reasoning: some property is true for some completion of the IAF
- Necessary reasoning: some property is true for each completion of the IAF



- a_3 is skeptically accepted in each completion \rightarrow necessarily skeptically accepted
- a_4 is skeptically accepted in some completion \rightarrow possibly skeptically accepted

Possible and Necessary Reasoning

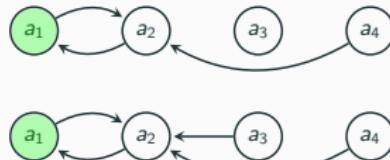
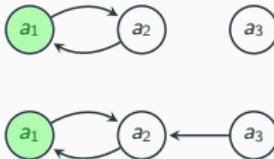
- Possible reasoning: some property is true for some completion of the IAF
- Necessary reasoning: some property is true for each completion of the IAF



- a_3 is skeptically accepted in each completion \rightarrow necessarily skeptically accepted
- a_4 is skeptically accepted in some completion \rightarrow possibly skeptically accepted
- a_2 is credulously accepted in some completion \rightarrow possibly credulously accepted

Possible and Necessary Reasoning

- Possible reasoning: some property is true for some completion of the IAF
- Necessary reasoning: some property is true for each completion of the IAF



- a_3 is skeptically accepted in each completion \rightarrow necessarily skeptically accepted
- a_4 is skeptically accepted in some completion \rightarrow possibly skeptically accepted
- a_2 is credulously accepted in some completion \rightarrow possibly credulously accepted
- a_1 is credulously accepted in each completion \rightarrow necessarily credulously accepted

Complexity of Reasoning with IAFs

- Possible Credulous Acceptability: a is in some extension of some completion
- Possible Skeptical Acceptability: a is in each extension of some completion
- Possible Verification: S is an extension of some completion
- Necessary Credulous Acceptability: a is in some extension of each completion
- Necessary Skeptical Acceptability: a is in each extension of each completion
- Necessary Verification: S is an extension of each completion

Semantics	PCA	PSA	PV	NCA	NSA	NV
ad	NP-c	trivial	NP-c	Π_2^P -c	trivial	P
st	NP-c	Σ_2^P -c	NP-c	Π_2^P -c	coNP-c	P
co	NP-c	NP-c	NP-c	Π_2^P -c	coNP-c	P
gr	NP-c	NP-c	NP-c	coNP-c	coNP-c	P
pr	NP-c	Σ_3^P -c	Σ_2^P -c	Π_2^P -c	Π_2^P -c	coNP-c

- SAT-based algorithms, CEGAR for second/third level of PH

A CAF is an argumentation framework where arguments are divided in three parts: *fixed*, *uncertain* and *control*.

fixed background knowledge about a static environment

uncertain changes that may occur in the environment

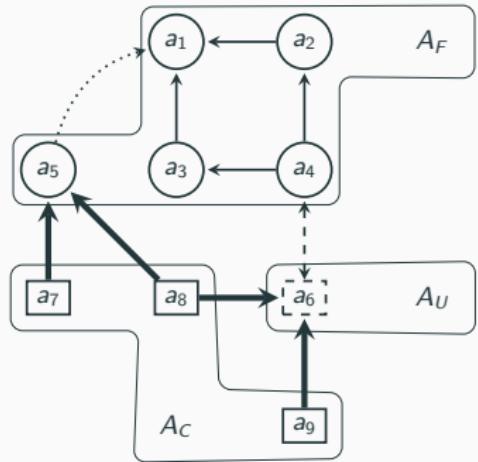
control possible actions of the agent

CAFs generalize IAFs in two directions:

- a new kind of uncertainty about the direction of attacks
- the control part representing how the agent can react to uncertain “threats”

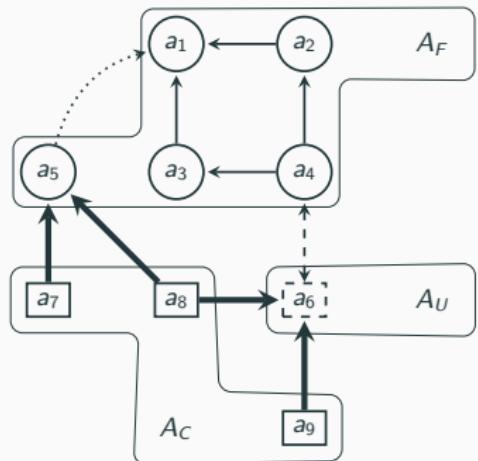
Y. Dimopoulos, J.-G. Mailly, P. Moraitis: Control Argumentation Frameworks. AAAI 2018: 4678-4685

A Picture is Worth a Thousand Words



- Fixed part: circle arguments + plain arrows
- Uncertain part:
 - dashed arguments
 - dotted arrows
 - two-heads dashed arrows
- Control part: square arguments + bold arrows

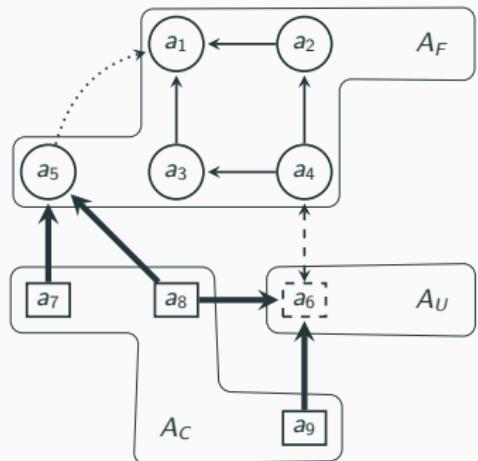
A Picture is Worth a Thousand Words



- Fixed part: **circle arguments + plain arrows**
- Uncertain part:
 - dashed arguments
 - dotted arrows
 - two-heads dashed arrows
- Control part: **square arguments + bold arrows**

- certain knowledge: always exist

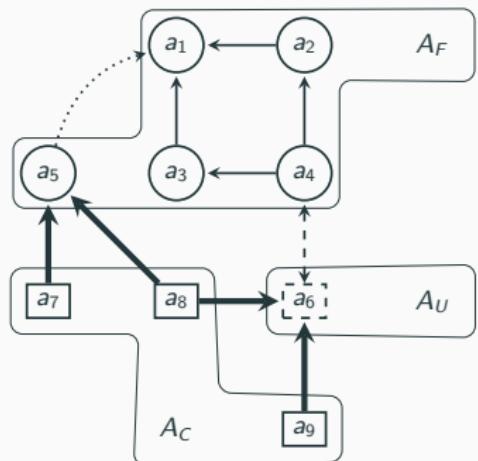
A Picture is Worth a Thousand Words



- Fixed part: circle arguments + plain arrows
- Uncertain part:
 - **dashed arguments**
 - dotted arrows
 - two-heads dashed arrows
- Control part: square arguments + bold arrows

- the argument could exist, or not

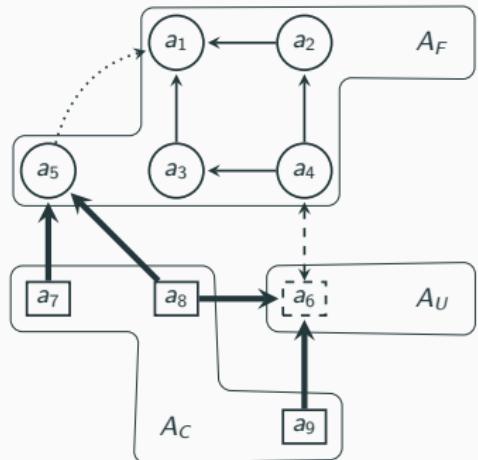
A Picture is Worth a Thousand Words



- Fixed part: circle arguments + plain arrows
- Uncertain part:
 - dashed arguments
 - **dotted arrows**
 - two-heads dashed arrows
- Control part: square arguments + bold arrows

- the attack could exist, or not

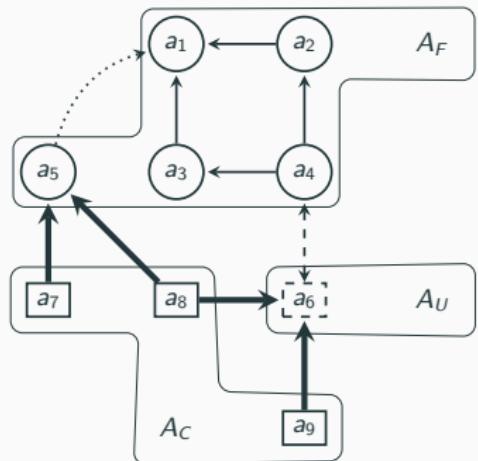
A Picture is Worth a Thousand Words



- Fixed part: circle arguments + plain arrows
- Uncertain part:
 - dashed arguments
 - dotted arrows
 - two-heads dashed arrows
- Control part: square arguments + bold arrows

- the attack exists (if both arguments exist), but we are not sure of the direction

A Picture is Worth a Thousand Words



- Fixed part: circle arguments + plain arrows
- Uncertain part:
 - dashed arguments
 - dotted arrows
 - two-heads dashed arrows
- Control part: **square arguments + bold arrows**

- exist only if the agent selects the arguments

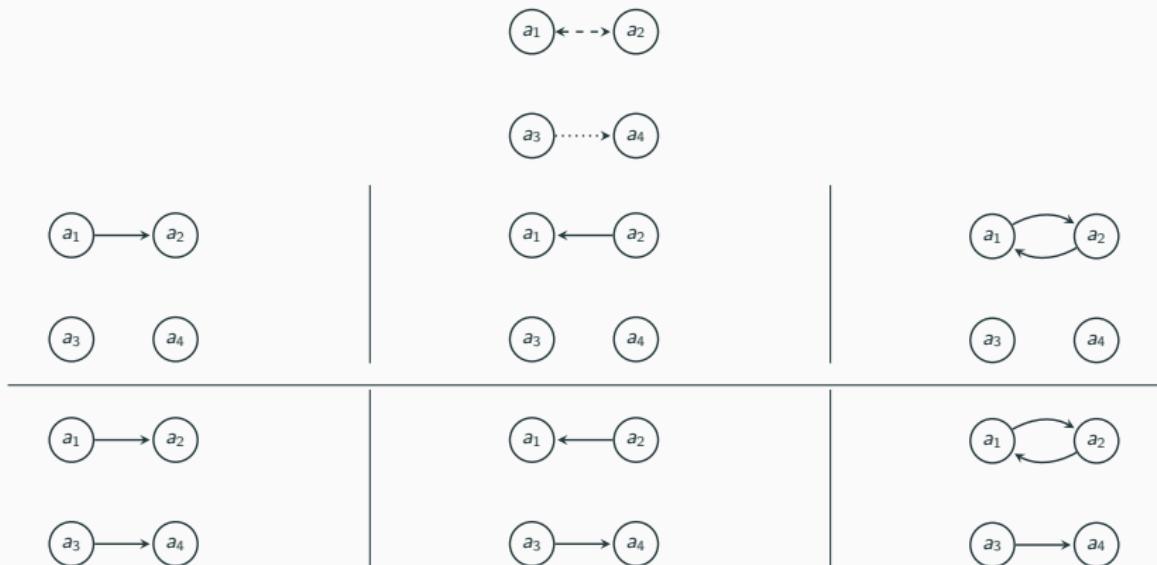
Completion

- Generalizes the notion of completion from IAFs, with 3 possible options for each “unknown direction conflict”



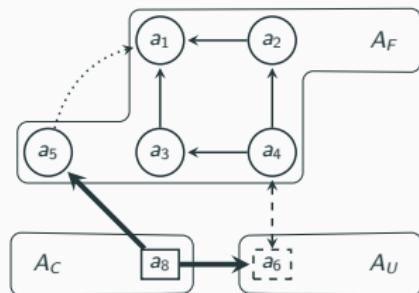
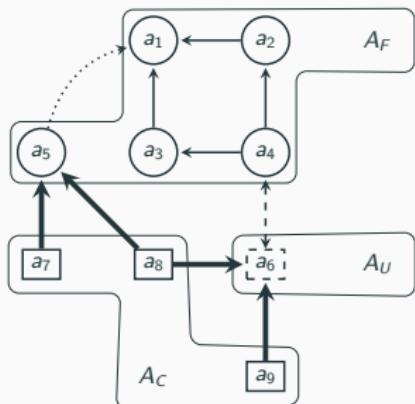
Completion

- Generalizes the notion of completion from IAFs, with 3 possible options for each “unknown direction conflict”



Control Configuration

- A control configuration is a subset $cc \subseteq A_C$
- A configured CAF: remove from the initial CAF the arguments $A_C \setminus cc$ (and their attacks)



Example: In the CAF configured by $cc = \{a_8\}$, $T = \{a_1\}$ is accepted whatever the completion

Formal Definition of Controllability

Given

- a target $T \subseteq A_F$
- a semantics σ

CAF is skeptically (resp. credulously) *controllable* w.r.t. T and σ if $\exists cc \subseteq A_C$ s.t.

- CAF' is the result of configuring CAF by cc
- T is included in every (resp. some) σ -extension of every completion of CAF'

Y. Dimopoulos, J.-G. Mailly, P. Moraitis: Control Argumentation Frameworks. AAAI 2018: 4678-4685

Formal Definition of Controllability

Given

- a target $T \subseteq A_F$
- a semantics σ

CAF is skeptically (resp. credulously) *controllable* w.r.t. T and σ if $\exists cc \subseteq A_C$ s.t.

- CAF' is the result of configuring CAF by cc
- T is included in every (resp. some) σ -extension of every completion of CAF'

Y. Dimopoulos, J.-G. Mailly, P. Moraitis: Control Argumentation Frameworks. AAAI 2018: 4678-4685

- Possible controllability: replace “every completion” by “some completion”

J.-G. Mailly: Possible Controllability of Control Argumentation Frameworks. COMMA 2020: 283-294

	Possible		Necessary	
	Skeptical	Credulous	Skeptical	Credulous
Grounded	NP-c	NP-c	Σ_2^P -c	Σ_2^P -c
Complete	NP-c	NP-c	Σ_2^P -c	Σ_3^P -c
Stable	Σ_2^P -c	NP-c	Σ_2^P -c	Σ_3^P -c
Preferred	Σ_3^P -c	NP-c	Σ_3^P -c	Σ_3^P -c

- QBF encodings proposed in ([Dimopoulos et al, Mailly](#))
- CEGAR algorithms proposed in ([Niskanen et al](#))

Y. Dimopoulos, [J.-G. Mailly](#), P. Moraitsis: Control Argumentation Frameworks. AAAI 2018: 4678-4685

[J.-G. Mailly](#): Possible Controllability of Control Argumentation Frameworks. COMMA 2020: 283-294

A. Niskanen, D. Neugebauer, M. Järvisalo: Controllability of Control Argumentation Frameworks. IJCAI 2020: 1855-1861

Outline

Background: Abstract Argumentation

Dung's Framework

SAT-based Computation for Argumentation

What is Argumentation Dynamics?

Dynamic Computation

The Subgraph-based Approach

The Iterative SAT-based Approach

Extension Enforcement

How to modify arguments acceptability?

Argumentation under Uncertainty

Argumentation Dynamics under Uncertainty

Application to a Multi-Agent Scenario: Automated Negotiation

Conclusion

- Context: several agents have their own knowledge/beliefs, preferences and goals
- Objective: reach an agreement between them (e.g. about an action to perform, a product to buy, etc)
- Example:
 - The car sellers knows the different cars and their features, and wants to sell the most expensive one
 - The potential buyer knows his needs (small car, family car, big trunk,...) and preferences (color of the car, heating seats,...), and wants to buy a car that meets these needs, as cheap as possible

- Argumentation: activity that aims at increasing (or decreasing) the acceptance degree of a controversial point of view
- Arguments: pieces of information that justify (or defeat) a point of view
- Using argumentation in negotiation improves the chances to reach an agreement (Sycara 1990)
 - An offer supported by a good argument is more likely to be accepted
 - An agent can modify its goals and preferences when he receives arguments

K. Sycara: Persuasive argumentation in negotiation. Theory and Decision 28(3):203-242, 1990.

- If available, opponent modeling is a great tool to improve the issue of negotiation
 - possibility to choose offers/arguments that the opponent is likely to accept
 - anticipate what he will say
 - use his own arguments to convince him
- CAFs are a good way to model the knowledge of an agent about the other agent
 - Fixed part: certain knowledge about the opponent
 - Uncertain part: uncertain knowledge about the opponent
 - Control part: possible actions of the agent to convince the opponent

Y. Dimopoulos, J.-G. Mailly, Pavlos Moraitis: Arguing and negotiating using incomplete negotiators profiles.

Auton. Agents Multi Agent Syst. 35(2): 18 (2021)

- No fixed role: both agent act as proponent and opponent in turn
- Each agent has a (incomplete and uncertain) representation of its opponent's theory: fixed and uncertain part of the CAF
- In the CAF, control arguments come from the agent's own theory: persuasive arguments
- The proponent selects the best offer w.r.t. his own theory, but uses arguments from the opponent's theory to defend the offer → facilitates persuasion

Y. Dimopoulos, J.-G. Mailly, Pavlos Moraitis: Arguing and negotiating using incomplete negotiators profiles.

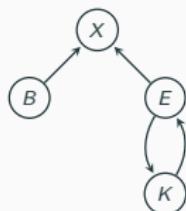
Auton. Agents Multi Agent Syst. 35(2): 18 (2021)

Negotiation theory of an agent α : $\mathcal{T} = \langle \mathcal{O}, \mathcal{T}^\alpha, \mathcal{C}^{\alpha,\beta}, \mathcal{F}^\alpha \rangle$ with

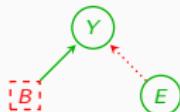
- \mathcal{O} : set of offers
 - Can be “simple” objects (mono-issue negotiation): $\{car_1, car_2, \dots, car_n\}$
 - Or “complex” objects (multi-issue negotiation):
 $\{(car_1, price_1, delivery_1), \dots, (car_n, price_m, delivery_k)\}$
- \mathcal{T}^α : the agent's AF
- $\mathcal{C}^{\alpha,\beta}$: the knowledge of α about his opponent β
 - $\mathcal{C}^{\alpha,\beta}$ is made from \mathcal{T}^β : there is uncertainty, there is ignorance, but there is no mistake or lie
- $\mathcal{F}^\alpha : \mathcal{O} \rightarrow 2^{A_p^\alpha}$ maps offers to the set of practical arguments supporting them

Example of Negotiation Theories

Agent α : X supports O

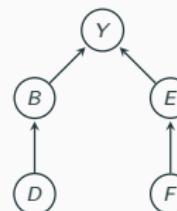


(a) T^α

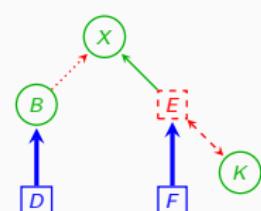


(b) $C^{\alpha,\beta}$

Agent β : Y supports O



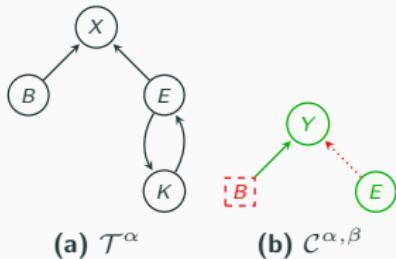
(c) T^β



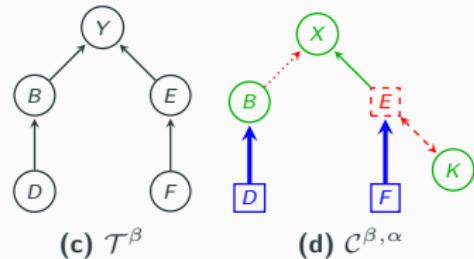
(d) $C^{\beta,\alpha}$

Bidding Strategy

Agent α : X supports O



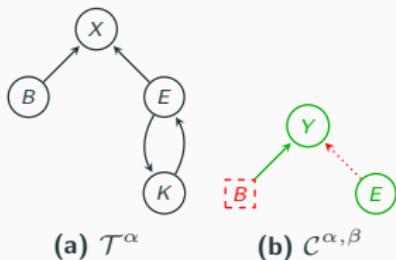
Agent β : Y supports O



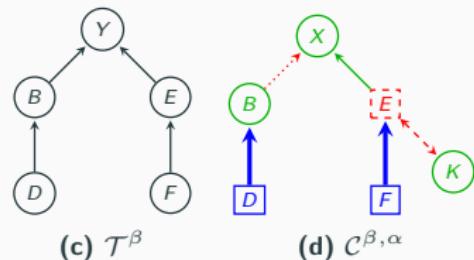
α 's turn: X is not accepted in \mathcal{T}^α , so α cannot support the (unique) offer O
→ the token goes to β

Bidding Strategy

Agent α : X supports O



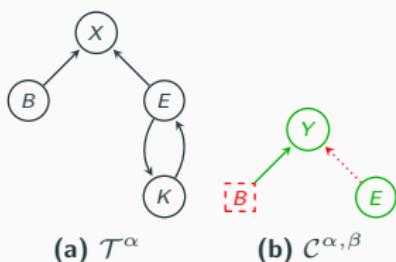
Agent β : Y supports O



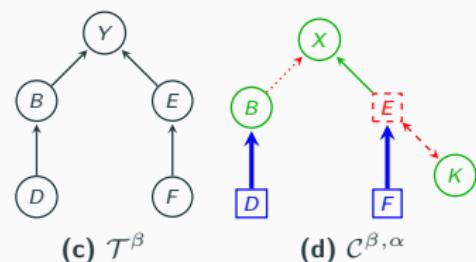
β 's turn: best offer according to β 's personnal AF is O because the supporting argument Y is accepted in \mathcal{T}^β

Bidding Strategy

Agent α : X supports O

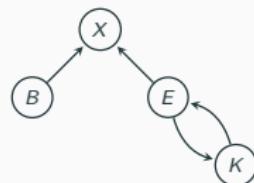


Agent β : Y supports O

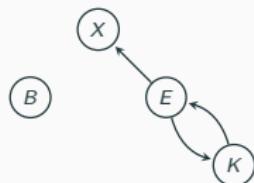


β 's turn: support argument for O in α 's theory is X

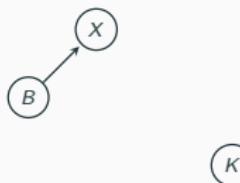
β 's Proposal without Control



(a) Completion 1



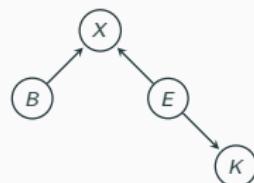
(b) Completion 2



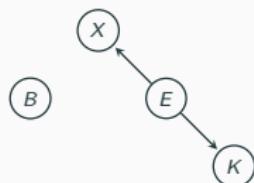
(c) Completion 3



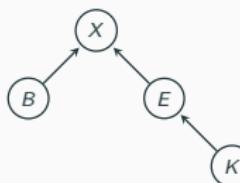
(d) Completion 4



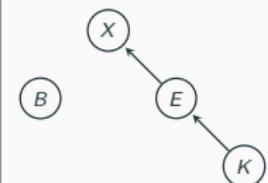
(e) Completion 5



(f) Completion 6



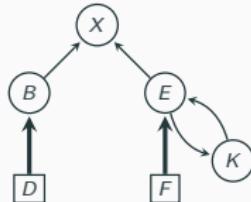
(g) Completion 7



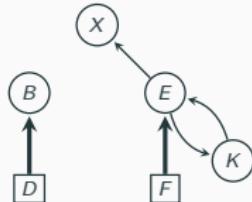
(h) Completion 8

- X is not accepted in each completion (e.g. Completion 1)

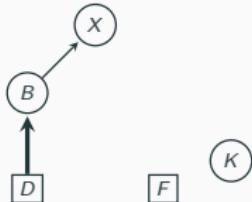
β 's Proposal with Control



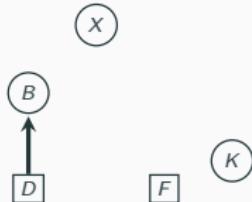
(a) Completion 1



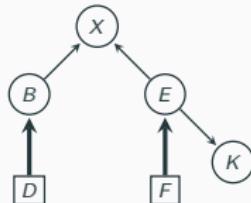
(b) Completion 2



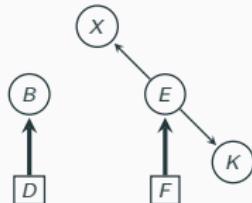
(c) Completion 3



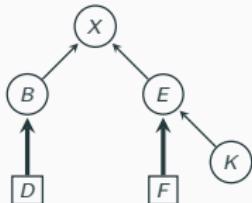
(d) Completion 4



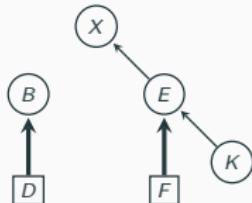
(e) Completion 5



(f) Completion 6



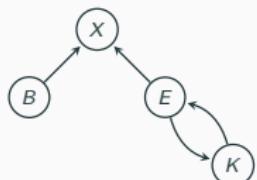
(g) Completion 7



(h) Completion 8

- X is accepted in each completion

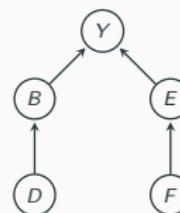
Acceptance Strategy



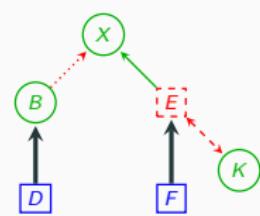
(a) T^α



(b) $C^{\alpha, \beta}$



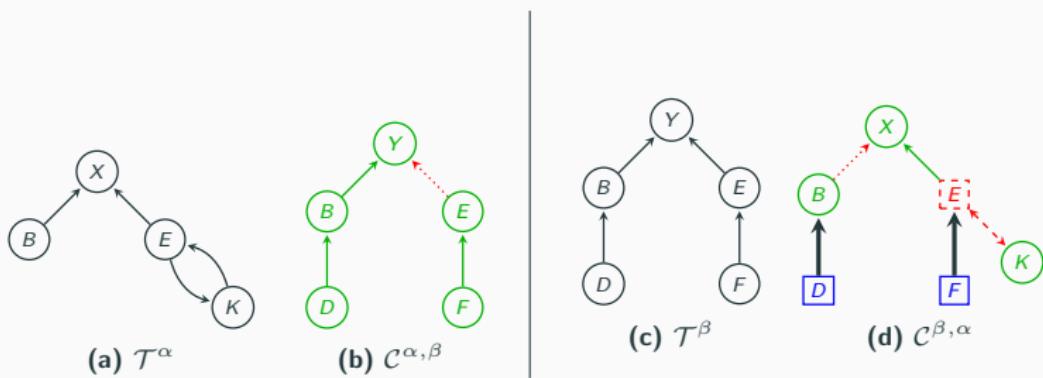
(c) T^β



(d) $C^{\beta, \alpha}$

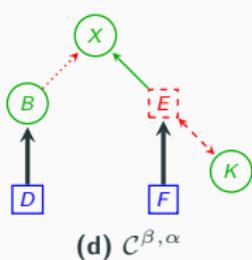
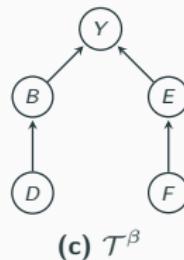
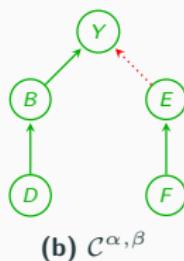
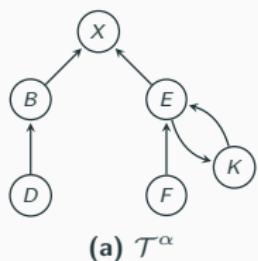
β 's turn: proposal is offer O , supported by argument X , defended by D and F

Acceptance Strategy



α updates its CAF: uncertainty decreases

Acceptance Strategy



α updates its own theory. X is now accepted: **agreement**

- If both agents end with no available offer, then the negotiation fails
- Otherwise, the outcome of the negotiation is the proponent's offer from the last round of the dialogue
- Experiments have shown the interest of using CAFs: better agreement rate if agents have enough information on their opponent and enough control arguments

Outline

Background: Abstract Argumentation

Dung's Framework

SAT-based Computation for Argumentation

What is Argumentation Dynamics?

Dynamic Computation

The Subgraph-based Approach

The Iterative SAT-based Approach

Extension Enforcement

How to modify arguments acceptability?

Argumentation under Uncertainty

Argumentation Dynamics under Uncertainty

Application to a Multi-Agent Scenario: Automated Negotiation

Conclusion

Conclusion

Summary

- Argument-based reasoning is computationally hard, but can be solved thanks to SAT-based techniques and various algorithmic tricks (e.g. subgraph selection in dynamic argumentation)
- Argumentation under uncertainty/in dynamics scenarios has theoretical interest, and practical interest (e.g. automated negotiation)

Future work

- Richer frameworks (supports, collective attacks, weighted, higher order relations, etc)
- Other kinds of semantics (ranking, gradual)

Conclusion

Summary

- Argument-based reasoning is computationally hard, but can be solved thanks to SAT-based techniques and various algorithmic tricks (e.g. subgraph selection in dynamic argumentation)
- Argumentation under uncertainty/in dynamics scenarios has theoretical interest, and practical interest (e.g. automated negotiation)

Future work

- Richer frameworks (supports, collective attacks, weighted, higher order relations, etc)
- Other kinds of semantics (ranking, gradual)

Thanks for your attention!

Feel free to contact me if you have any question

jean-guy.mailly@irit.fr