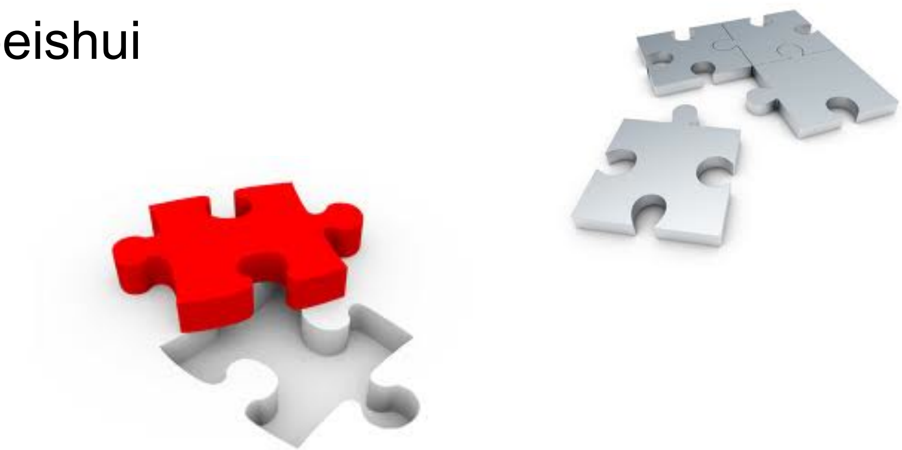
On the Modularity of Argumentation

Beishui Liao
Zhejiang University
baiseliao@zju.edu.cn
http://mypage.zju.edu.cn/beishui



Outline

- Background: Dung's abstract argumentation
- Why modularity is important?
- Components: sub-framework/argumentation multipole
- Incremental computation/argumentation dynamics based on sub-framework
- Semantics interchangeability based on argumentation multipole

Arguments

A: The sun is near to us at daybreak and far away at noon: when the sun first appears, it is as big as the canopy of a carriage, but at noon it is only the size of a plate or a bowl; isn't it true that objects farther away seem smaller while those nearer seem bigger？

B: The sun is far away at dawn and nearby at midday: when the sun comes out, it is very cool, but at midday it is as hot as putting your hands in boiling water; isn't it true that what is nearer to us is hotter and what is farther off is cooler？

Arguments

A

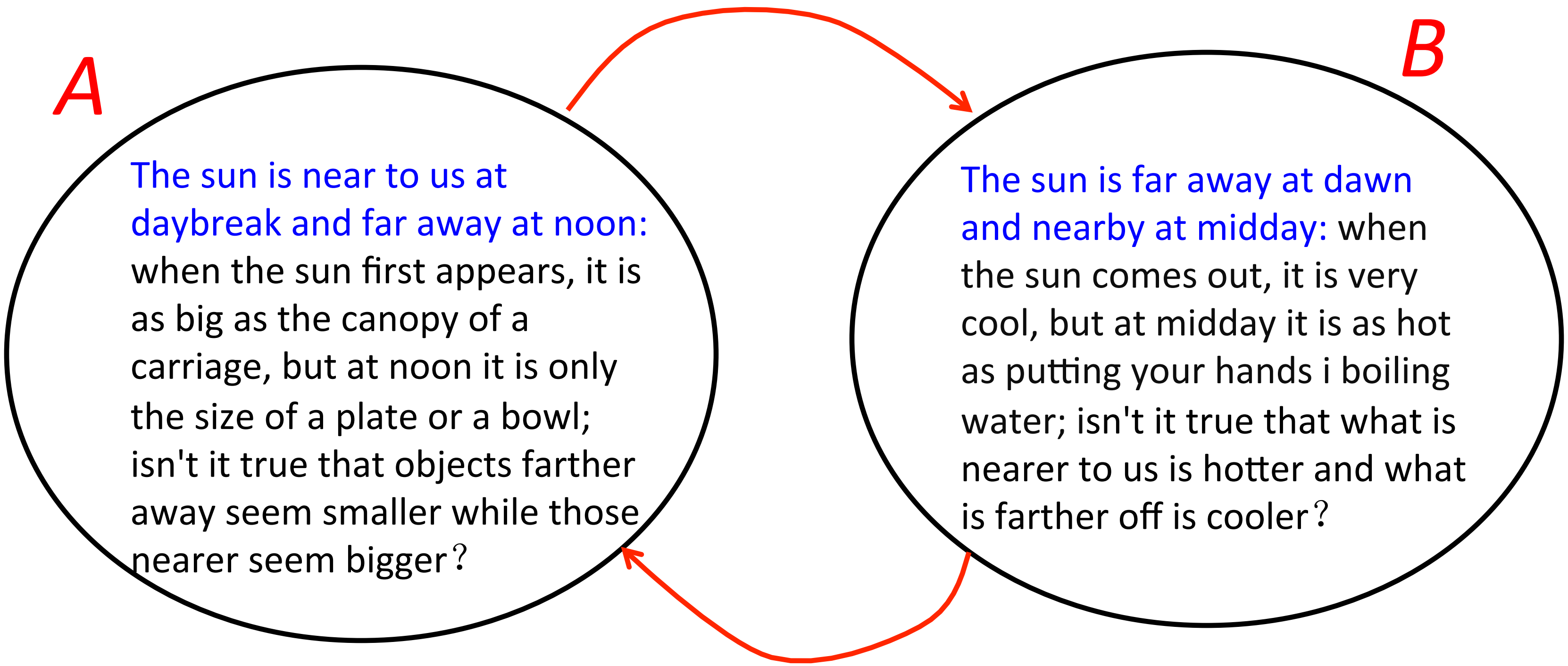
The sun is near to us at daybreak and far away at noon: when the sun first appears, it is as big as the canopy of a carriage, but at noon it is only the size of a plate or a bowl; isn't it true that objects farther away seem smaller while those nearer seem bigger？

B

The sun is far away at dawn and nearby at midday: when the sun comes out, it is very cool, but at midday it is as hot as putting your hands i boiling water; isn't it true that what is nearer to us is hotter and what is farther off is cooler？

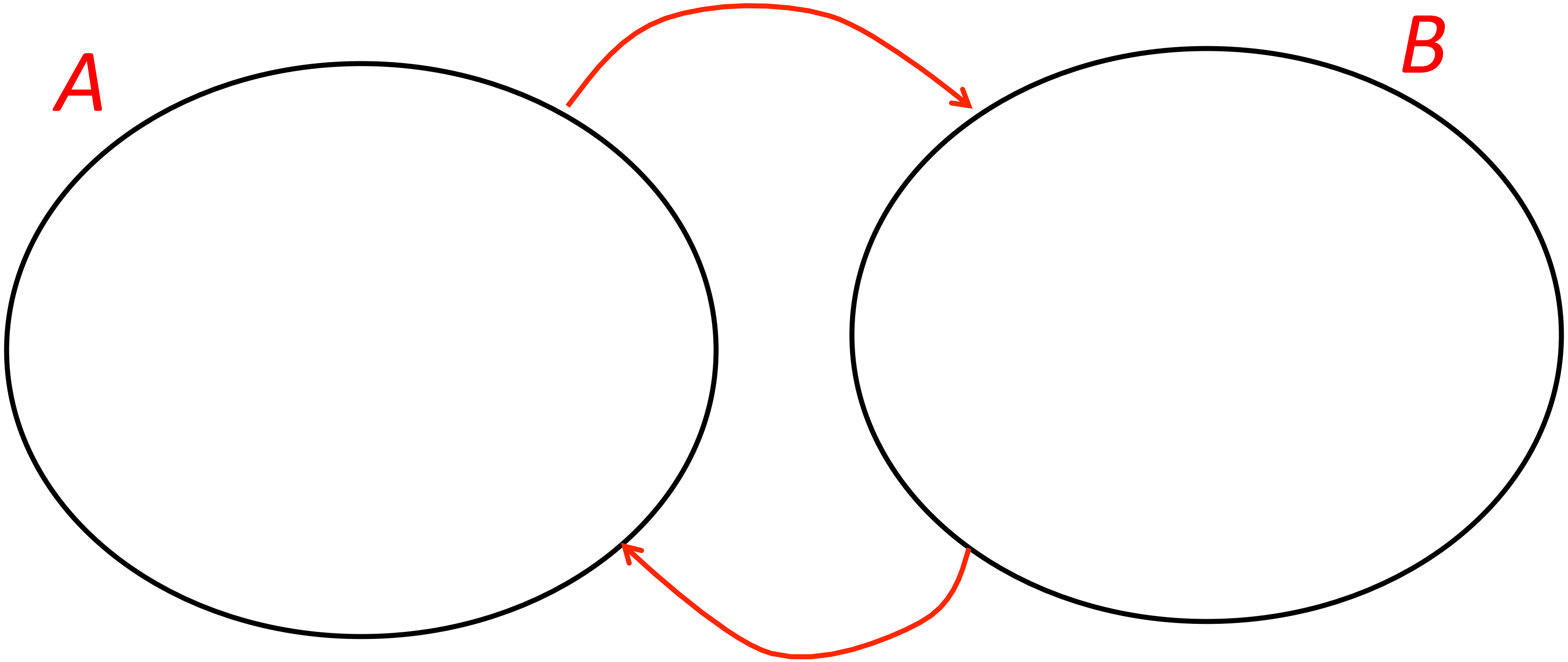
Each argument is internally consistent.

Attack relation over a set of arguments



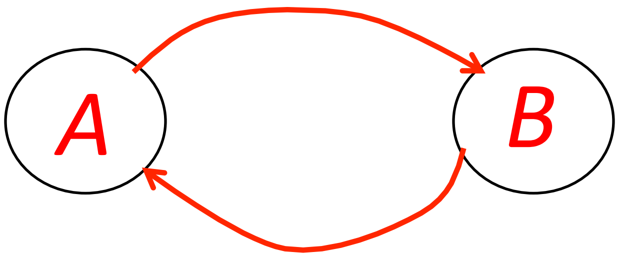
A rebuts B, and vice versa.

Abstract argumentation framework



Arguments are treated as atomic entities.

Abstract argumentation framework



Defeat graph

An abstract argumentation framework (or briefly, AF) is defined as a tuple: [Dung 1995]

(*Args*, *attacks*)

Args is a set of arguments

attacks $\subseteq$ *Args* × *Args* is a set of attacks

Argumentation semantics

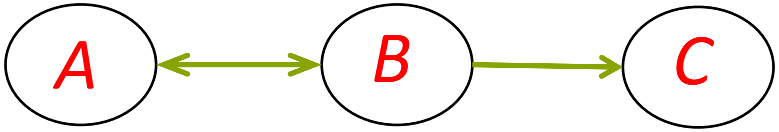
Extension-based approach

- Given an AF, to identify sets of arguments, called extensions, that can be regarded as collectively acceptable according to some criteria:
  - ✓ Admissible extension
  - ✓ Complete extension
  - ✓ Preferred extension
  - ✓ Grounded extension
  - ✓ …

Argumentation semantics

Extension-based approach --- Admissible extensions

- Let (*Args*, *attacks*) be an AF. A set of arguments $E \subseteq Args$ is admissible iff E is conflict-free and each argument in E is defended by E.
- E is conflict-free iff there exist no arguments in E such that one is attacked by another.
- $A \in Args$ is defended by E iff for all $C \in Args$ that attacks A, there exists an argument B in E such that B attacks C.



ALL ARGUMENTS IN THE FRAMEWORK ARE CONSIDERED

$E_1 = \{\ \}$ $E_3 = \{A, C\}$

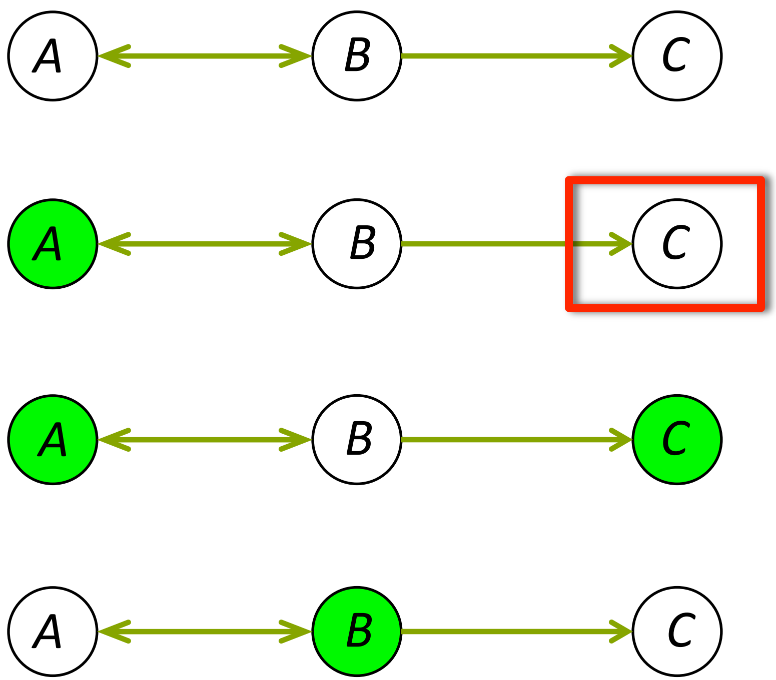
$E_2 = \{A\}$ $E_4 = \{B\}$

Beishui Liao 15-6-2

Argumentation semantics

Extension-based approach --- Complete extensions

- Let (*Args*, *attacks*) be an AF. A set of arguments $E \subseteq Args$ is a complete extension iff E is admissible and each argument A that is defended by E is in E.



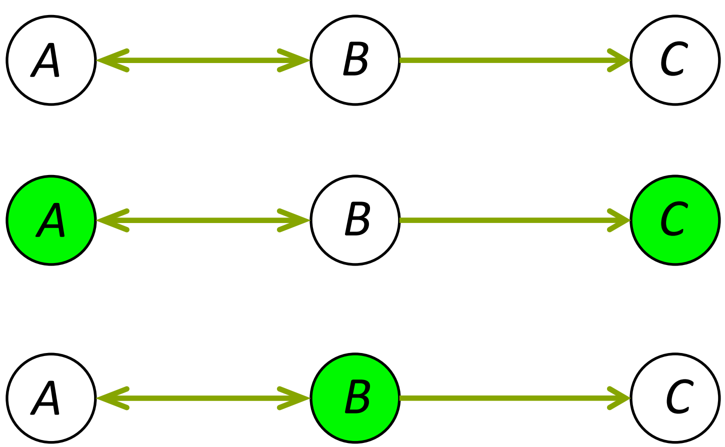
Admissible	Complete
$E_1 = \{\ \}$	✔
$E_2 = \{A\}$	✗
$E_3 = \{A, C\}$	✔
$E_4 = \{B\}$	✔

Argumentation semantics

Extension-based approach --- Preferred extensions

- Let (*Args*, *attacks*) be an AF. A set of arguments $E \subseteq Args$ is a preferred extension iff E is a maximal complete extension (with respect to set-inclusion).

A GLOBAL PROPERTY



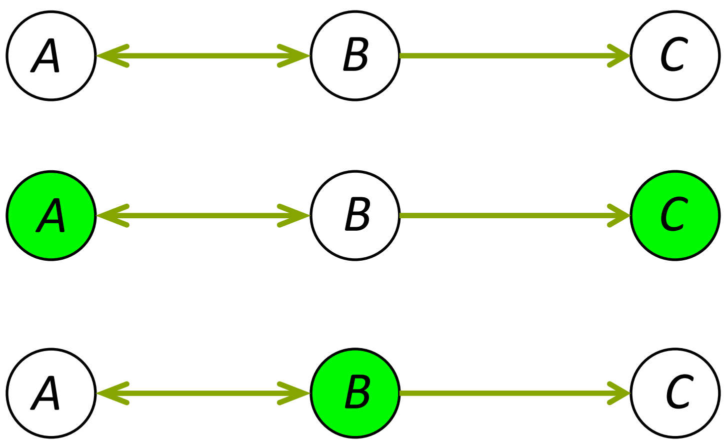
	Complete	Preferred
	$E_1 = \{\ \}$	✗
	$E_3 = \{A, C\}$	✔
	$E_4 = \{B\}$	✔

Argumentation semantics

Extension-based approach --- Grounded extension

- Let (*Args*, *attacks*) be an AF. A set of arguments $E \subseteq Args$ is a grounded extension iff E is the minimal complete extension (with respect to set-inclusion).

A GLOBAL PROPERTY



Complete	Grounded
$E_1 = \{\ \}$	✔
$E_3 = \{A, C\}$	✗
$E_4 = \{B\}$	✗

Argumentation semantics

Labelling-based approach [Caminada & Gabbay 2009]

- Given an AF, to assign a label (IN, OUT, UNDEC) to each argument, according to some criteria:
  - ✓ Admissible labelling
  - ✓ Complete labelling
  - ✓ Preferred labelling
  - ✓ Grounded labelling
  - ✓ …

Argumentation semantics

Labelling-based approach --- Admissible labellings

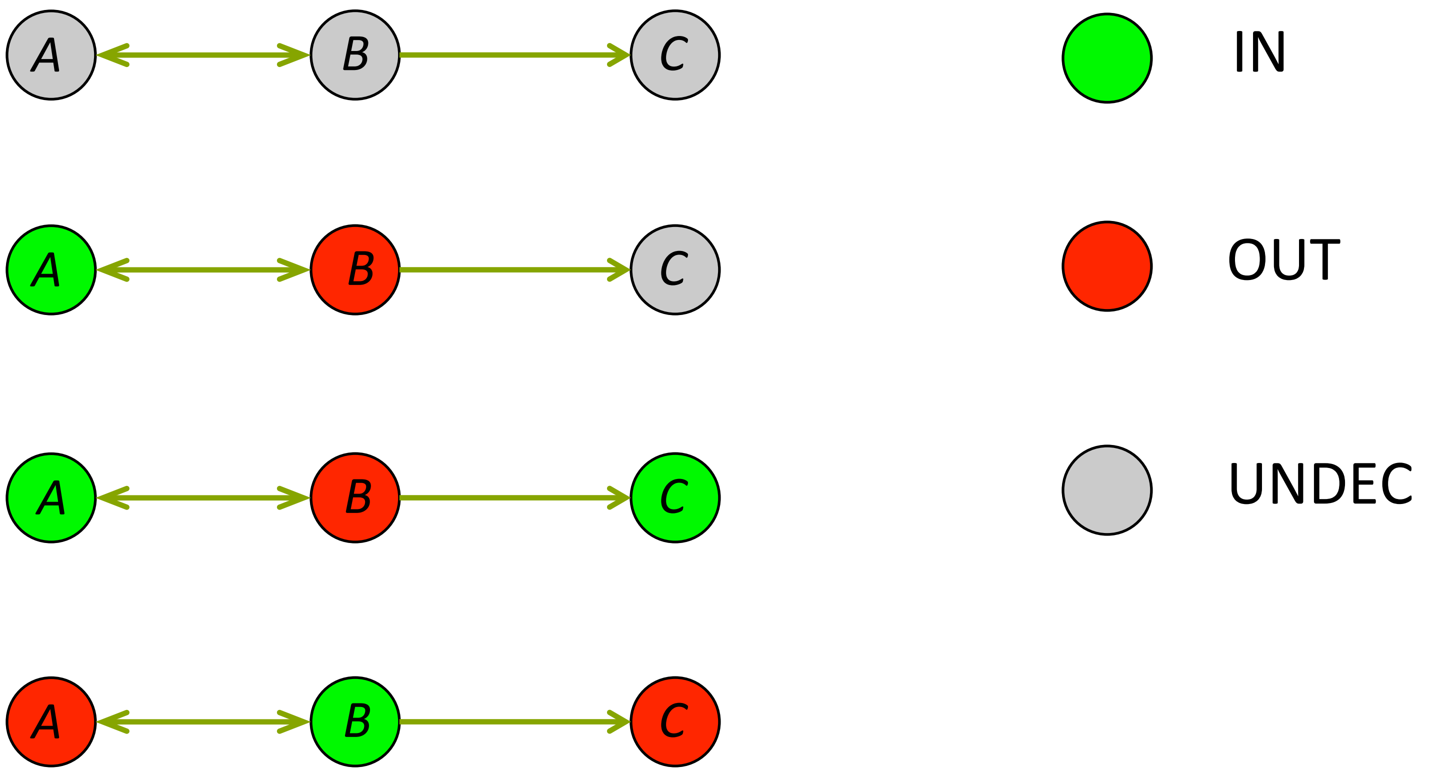
- Let (*Args*, *attacks*) be an AF. A labelling *L* is defined as a total function:

  L: Args $\mapsto$ {IN, OUT, UNDEC}

- An IN-labelled argument is legally IN, iff all its attackers are labelled OUT; an OUT-labelled argument is legally OUT, iff there exists an attacker that is labelled IN; an UNDEC-labelled argument is legally UNDEC, iff (1) it is not the case that all its attackers are labelled OUT, and (2) there exists no attacker that is labelled IN.
- A labelling *L* is admissible, iff each IN-labelled argument is legally IN, and each OUT-labelled argument is legally OUT.

Argumentation semantics

Labelling-based approach --- Admissible labellings

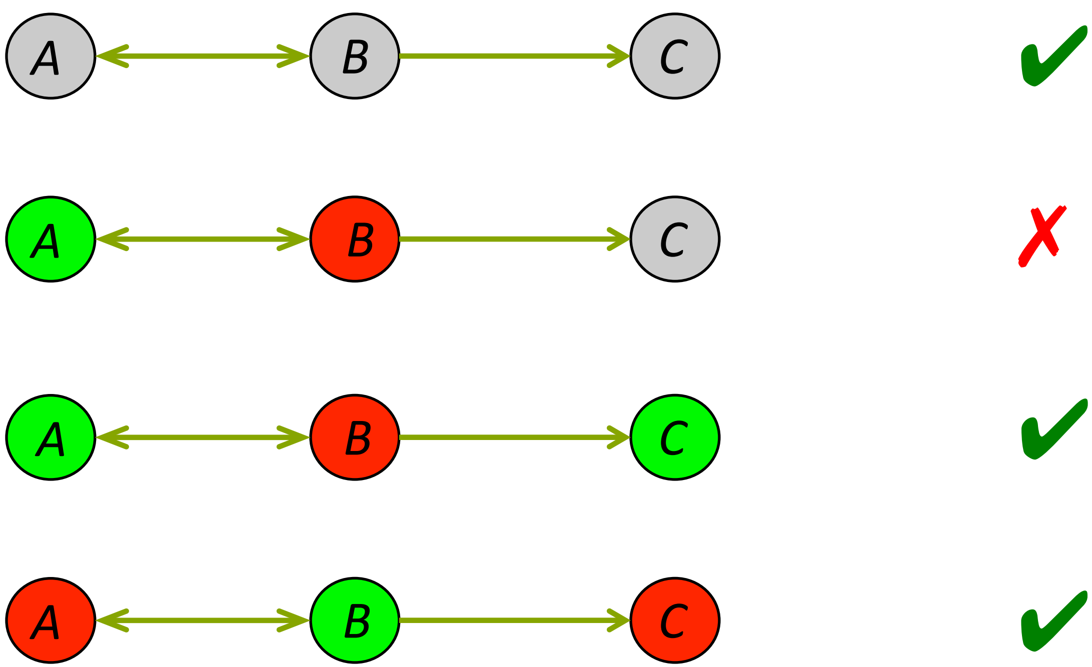


Argumentation semantics

Labelling-based approach --- Complete labellings

- Let (*Args*, *attacks*) be an AF. A labelling *L* is a complete labelling iff *L* is an admissible labelling and each UNDEC-labelled argument is legally UNDEC.

Admissible labellings | Complete labellings

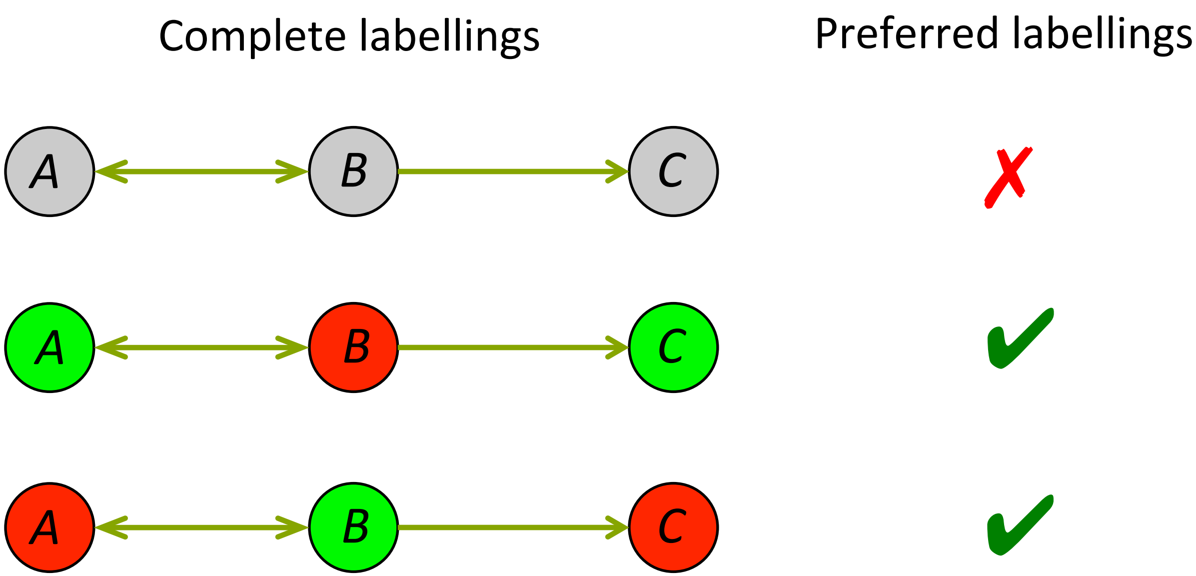


Argumentation semantics

Labelling-based approach --- Preferred labellings

- Let (*Args*, *attacks*) be an AF. A labelling *L* is a preferred labelling iff it is a complete labelling, and the set of IN-labelled arguments is maximal (with respect to set-inclusion).

A GLOBAL PROPERTY



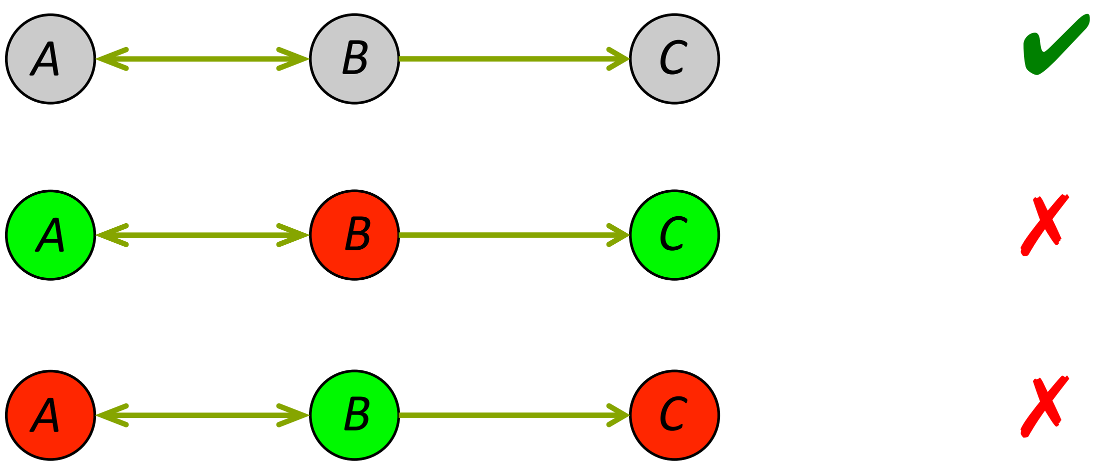
Argumentation semantics

Labelling-based approach --- Grounded labelling

- Let (*Args*, *attacks*) be an AF. A labelling *L* is a grounded labelling iff it is a complete labelling, and the set of IN-labelled arguments is minimal (with respect to set-inclusion).

A GLOBAL PROPERTY

Complete labellings　　　　Grounded labelling



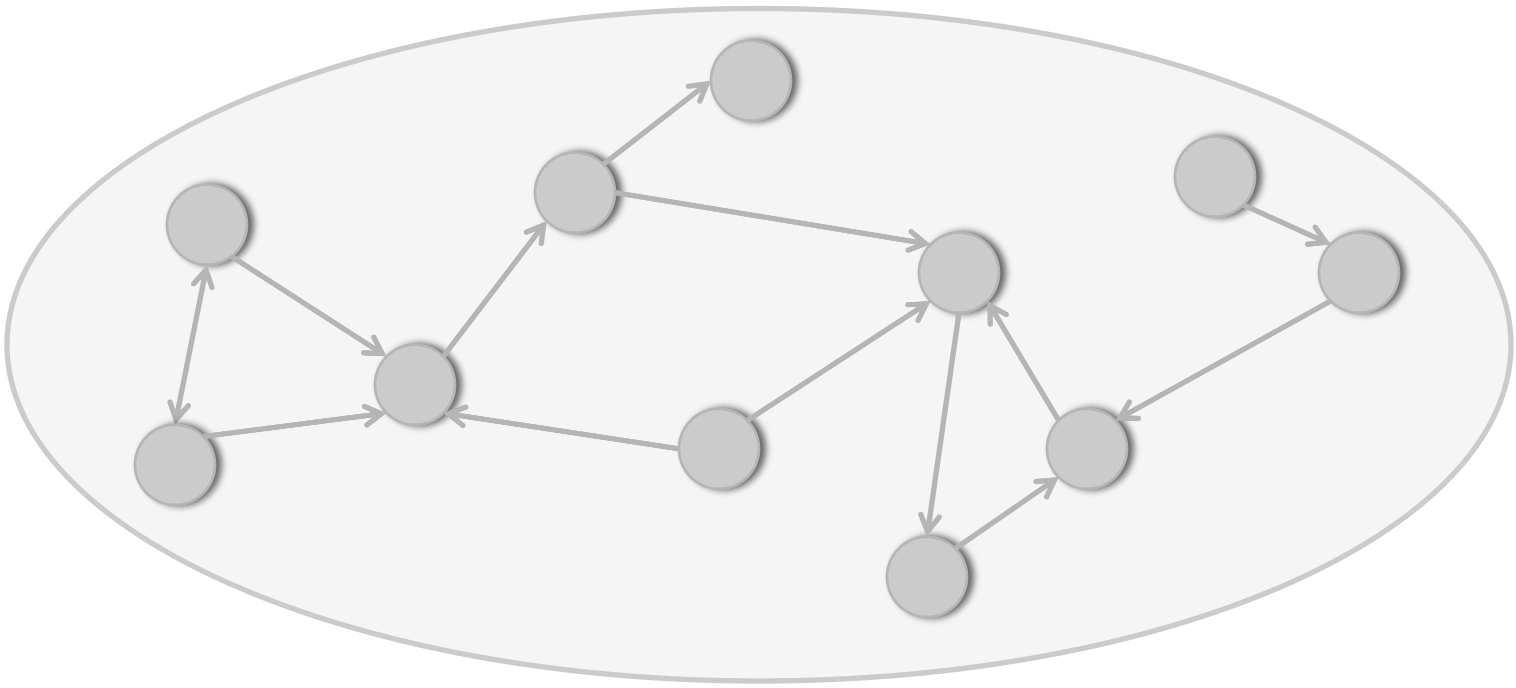
Outline

- Background: Dung's abstract argumentation
- **Why modularity is important?**
- Components: sub-framework/argumentation multipole
- Incremental computation/argumentation dynamics based on sub-framework
- Semantics interchangeability based on argumentation multipole

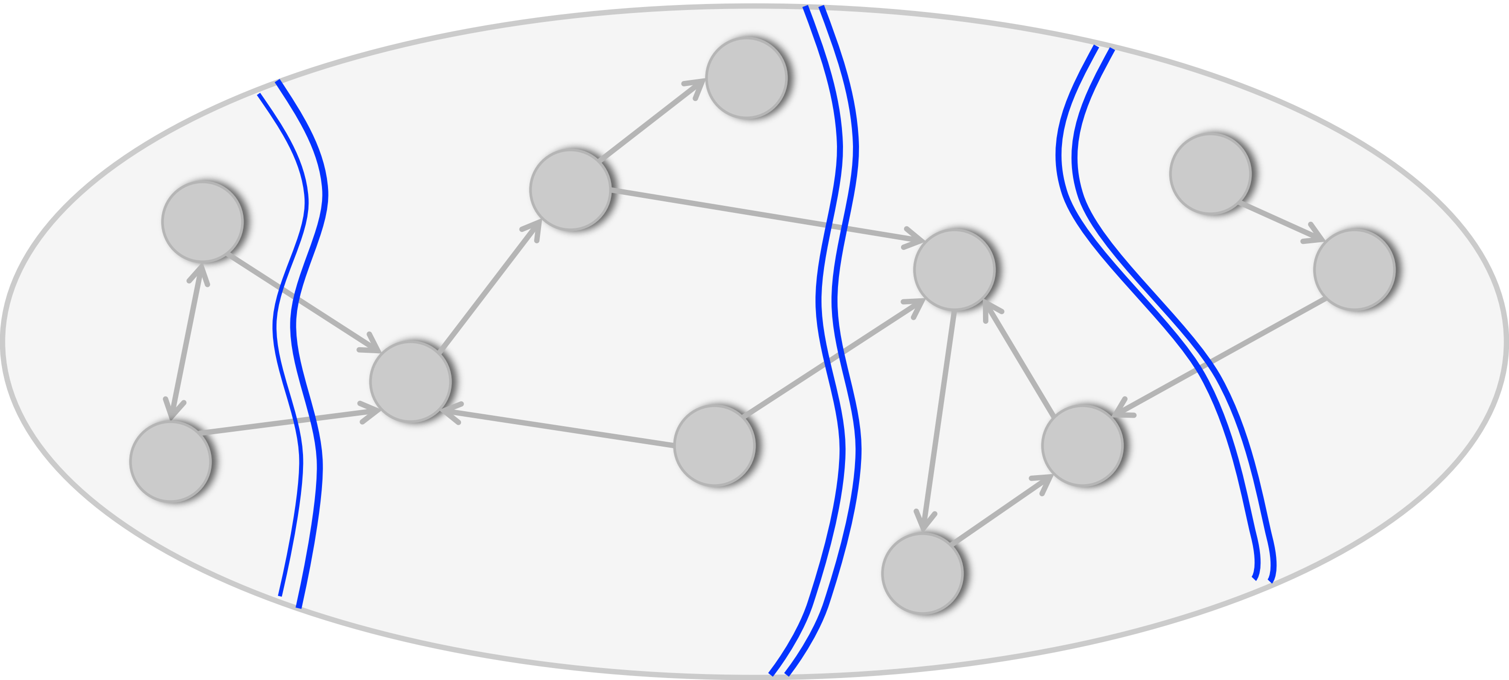
Why modularity is important

- Developing divide and conquer algorithms (efficiency)
- Handling dynamics of argumentation (efficiency)
- Replacement of some parts of system (interchangeability)
- …

Divide and conquer algorithms



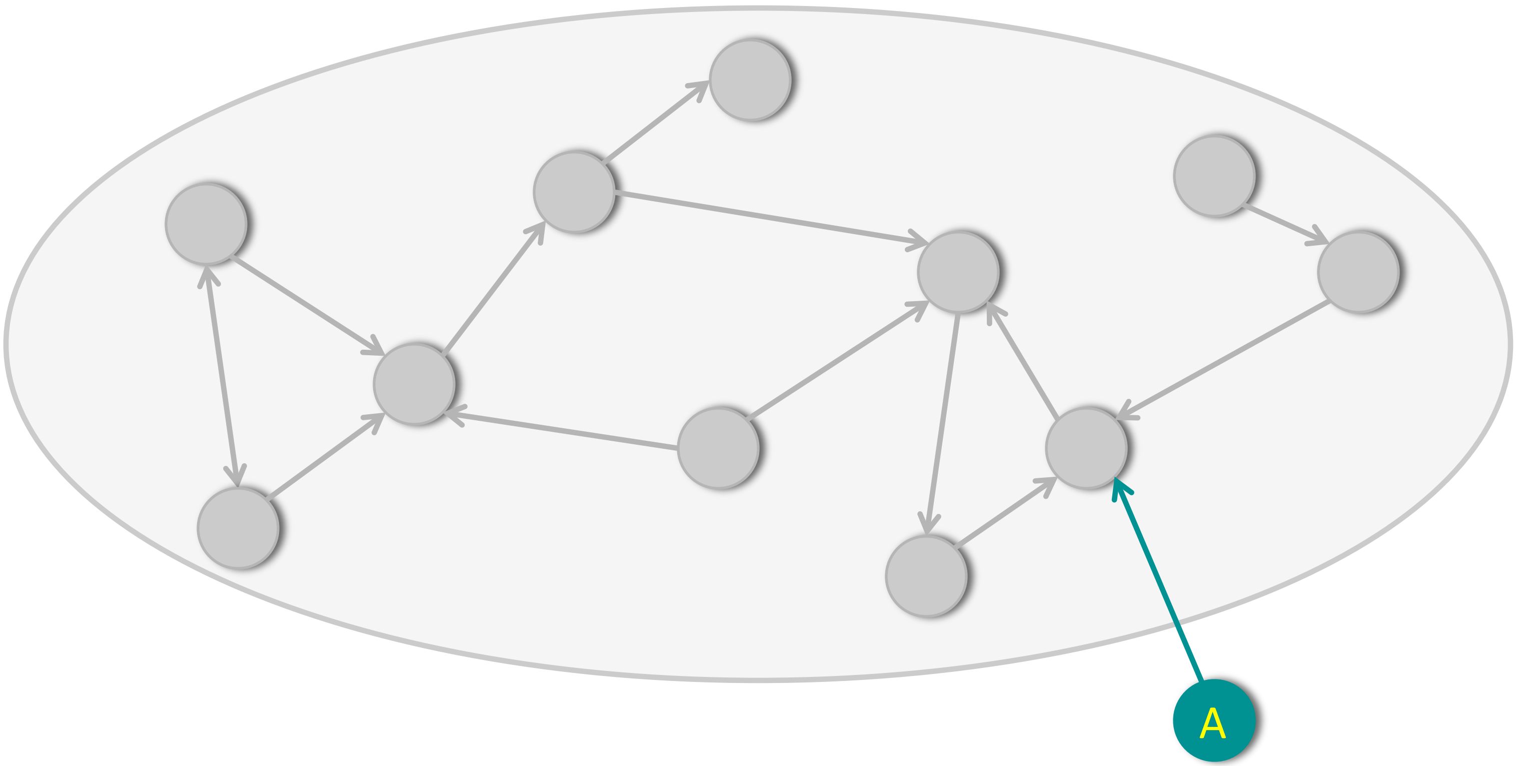
Divide and conquer algorithms



Dynamics: exogenous

Addition of new arguments and/or attacks

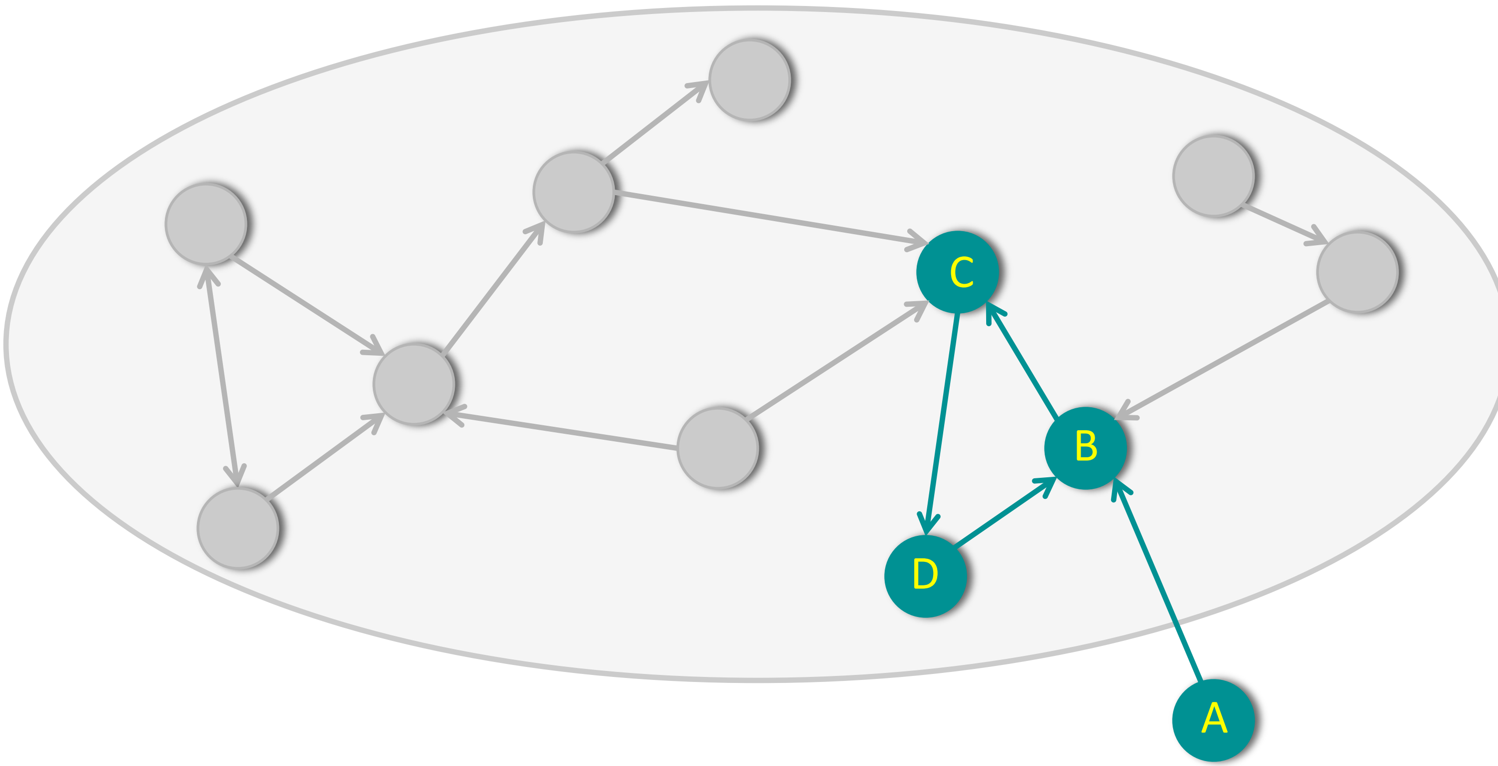
Removal of existing arguments and/or attacks



Dynamics: exogenous

Addition of new arguments and/or attacks

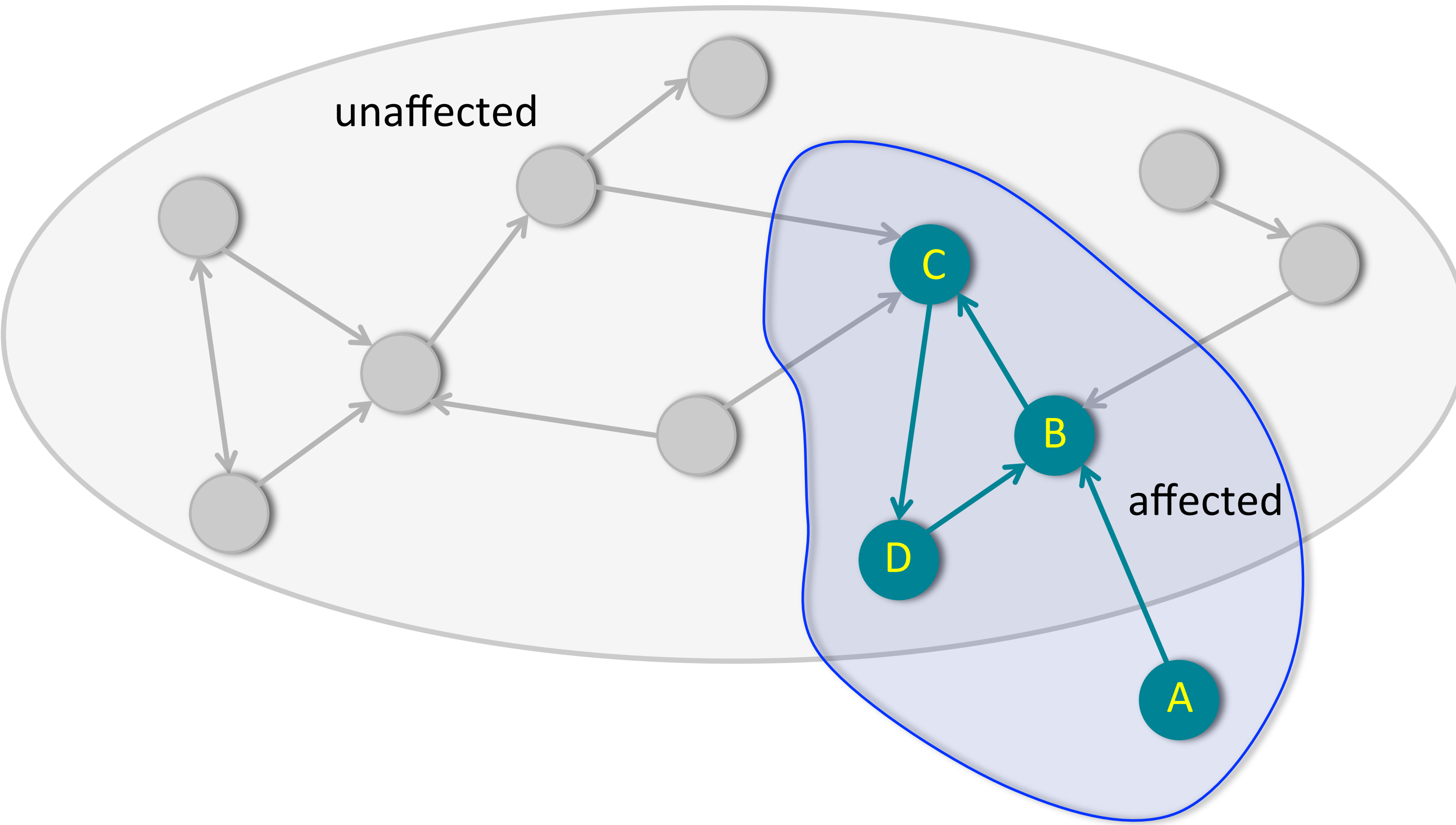
Removal of existing arguments and/or attacks



Dynamics: exogenous

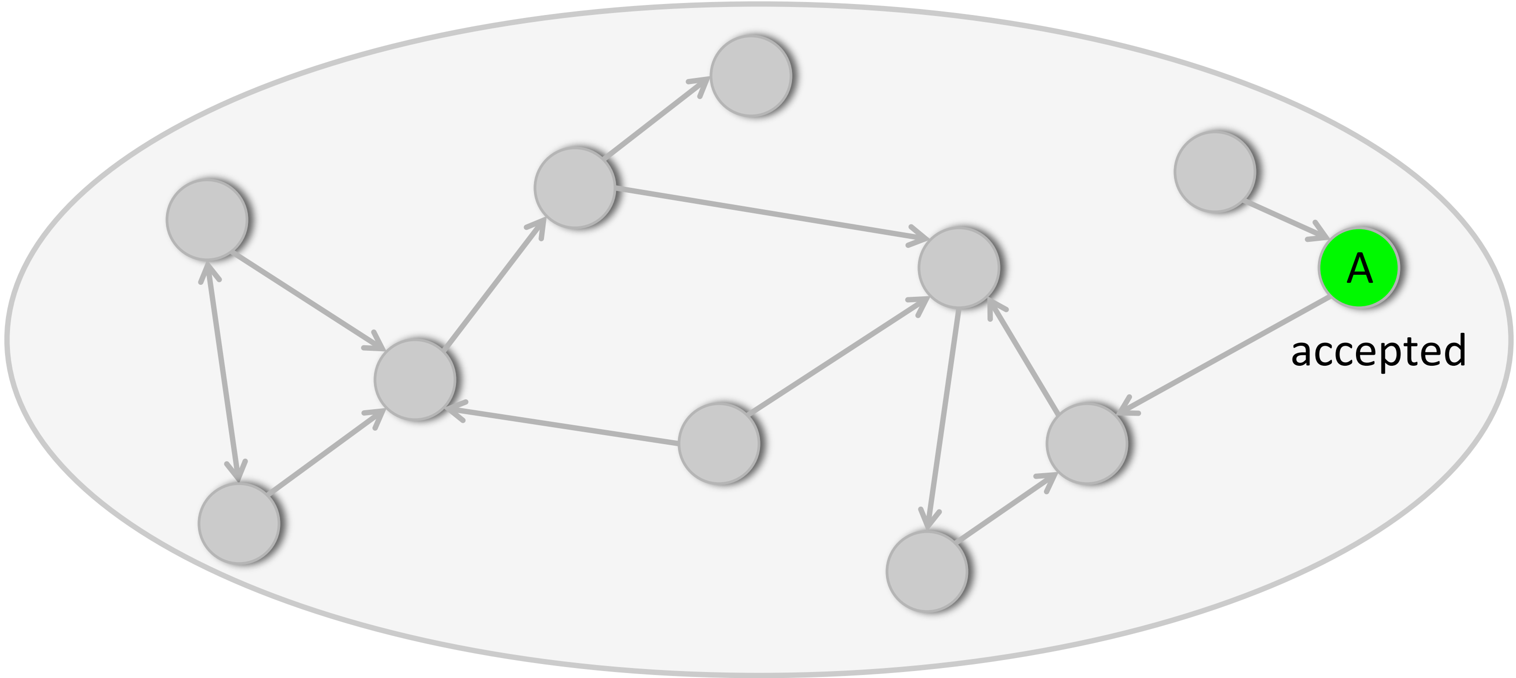
Addition of new arguments and/or attacks

Removal of existing arguments and/or attacks



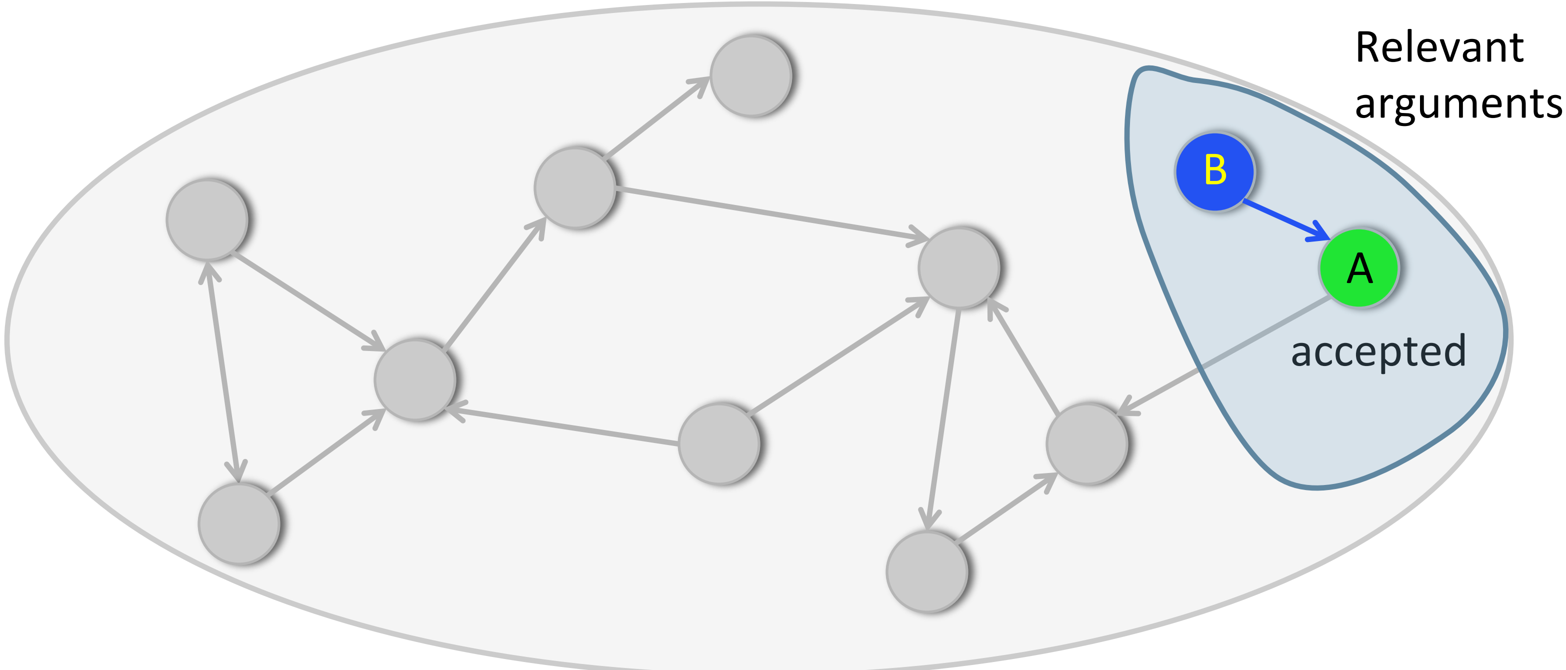
Dynamics: goal-driven

Enforce some arguments to be accepted or rejected

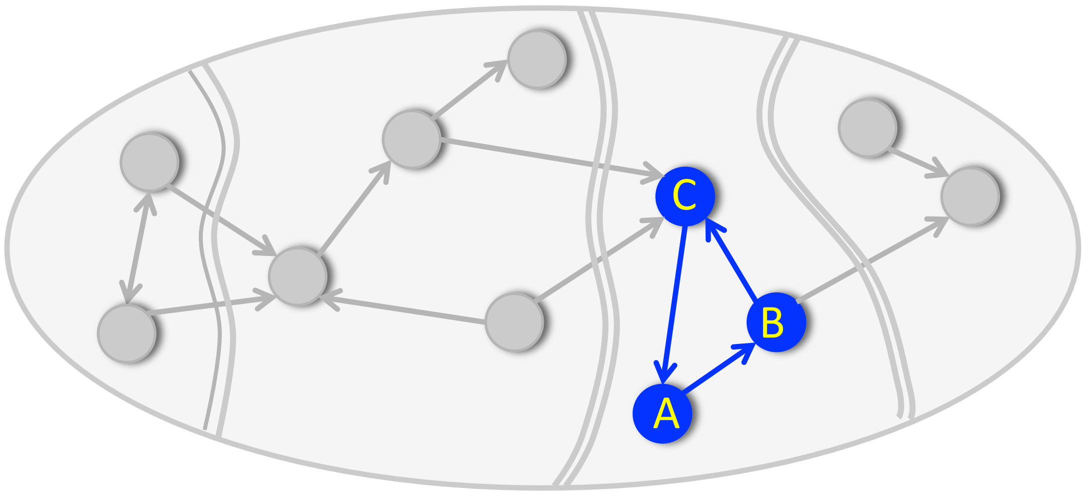


Dynamics: goal-driven

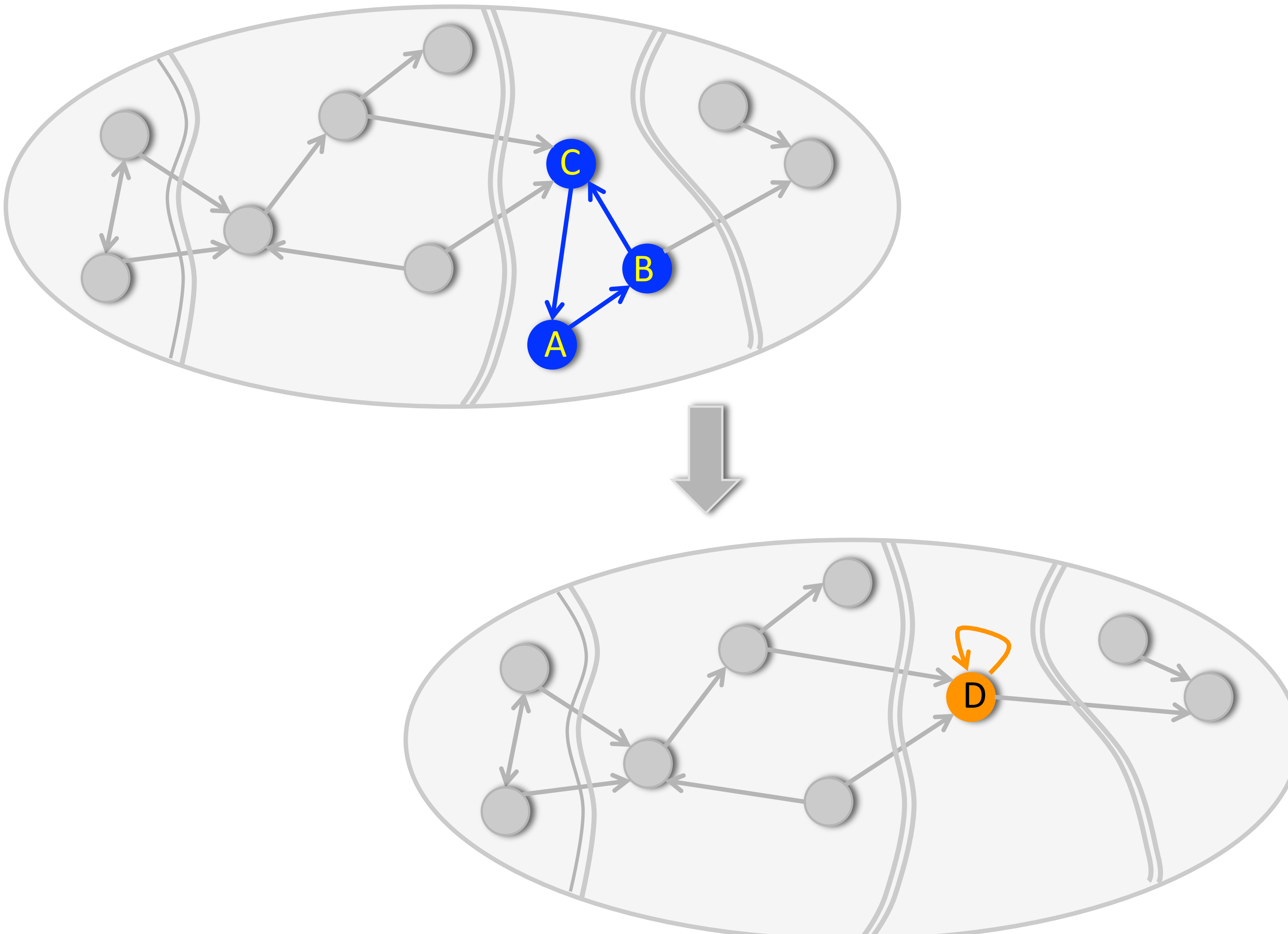
Enforce some arguments to be accepted or rejected



Replacement of components



Replacement of components

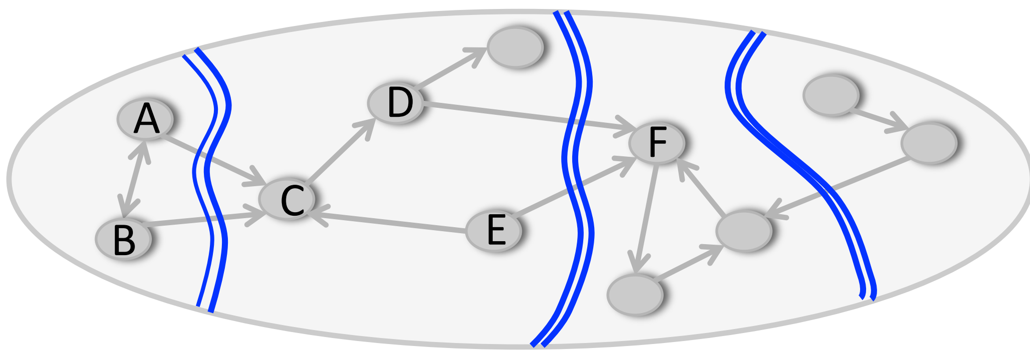


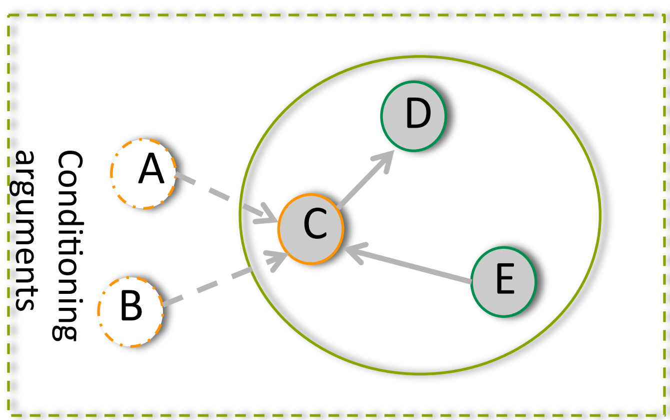
Beishui Liao 15-6-2

Outline

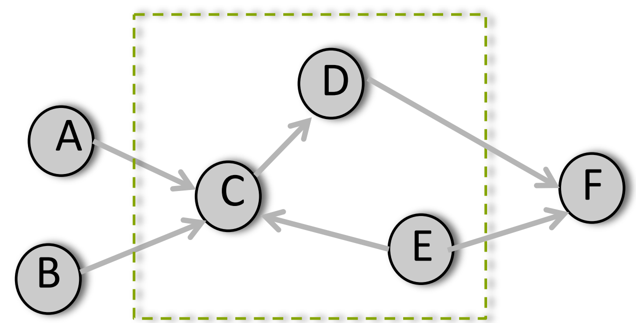
- Background: Dung's abstract argumentation
- Why modularity is important?
- Components: sub-framework/argumentation multipole
- Incremental computation/argumentation dynamics based on the notion of sub-framework
- Semantics interchangeability based on the notion of argumentation multipole

Component: sub-framework [Liao 2011]
Argumentation multipole [Baroni et al 2014]





sub-framework



Argumentation multipole

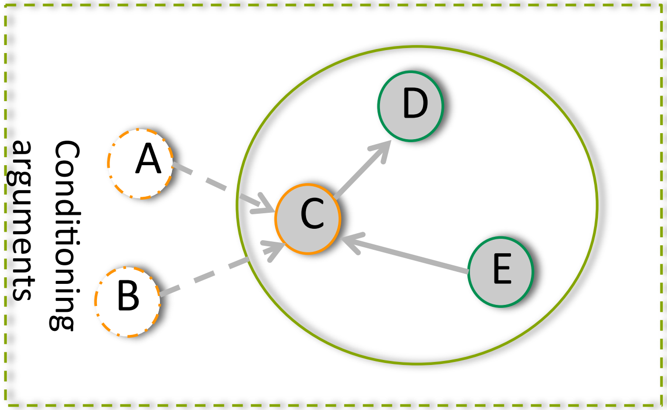
The set of conditioning arguments can be regarded as input, while output is not explicitly represented.

sub-frameworks [Liao 2011]

- Let (*Args*,*attacks*) be an AF, and S be a subset of *Args*
- $S^{-} = \{A \in Args \backslash S \mid \exists B \in S$, s.t. $(A, B) \in attacks\}$ denotes the set of outside parents (called conditioning arguments) of the arguments in S
- A sub-framework of (*Args*,*attacks*) is then defined as:

$$(S \cup S^{-}, R_S \cup I_S) \text{ where } R_S = attacks \cap (S \times S)$$

$$I_S = attacks \cap (S^{-} \times S)$$



$S = \{C, D, E\}$

$S^{-} = \{A, B\}$

$R_S = \{(C, D), (E, C)\}$

$I_S = \{(A, C), (B, C)\}$

Beishui Liao 15-6-2

Definition 4.2 (Dependence relation between sub-frameworks). Let $F = (A, R)$ be an argumentation framework, and $A_1, A_2 \subseteq A$ be subsets of A, $A_1 \neq A_2$. $(A_2 \cup A_2^-, R_{A_2} \cup I_{A_2})$ is *dependent* on $(A_1 \cup A_1^-, R_{A_1} \cup I_{A_1})$, if and only if $\exists \alpha \in A_1$ and $\beta \in A_2$ such that there is a path from α to β with respect to R. For convenience, $(A_1 \cup A_1^-, R_{A_1} \cup I_{A_1})$ is called a *conditioning sub-framework* of $(A_2 \cup A_2^-, R_{A_2} \cup I_{A_2})$.

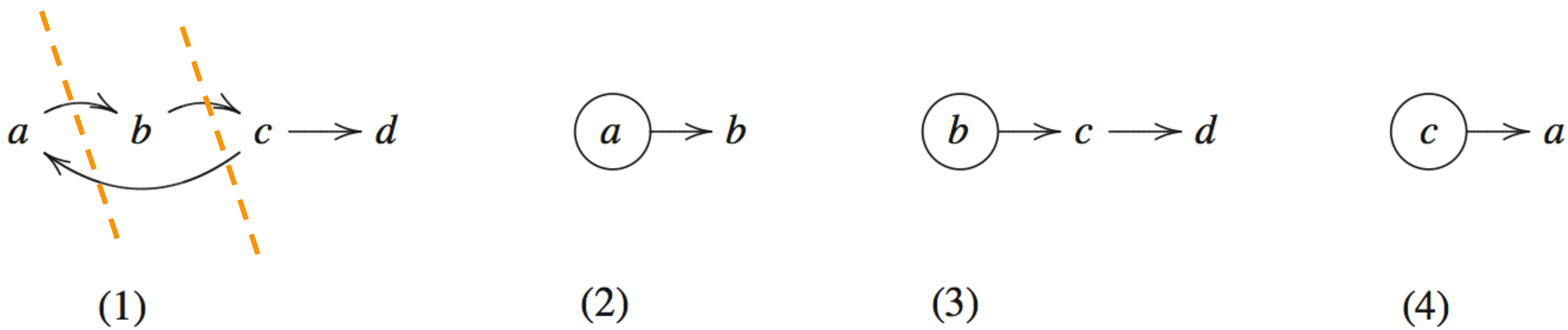


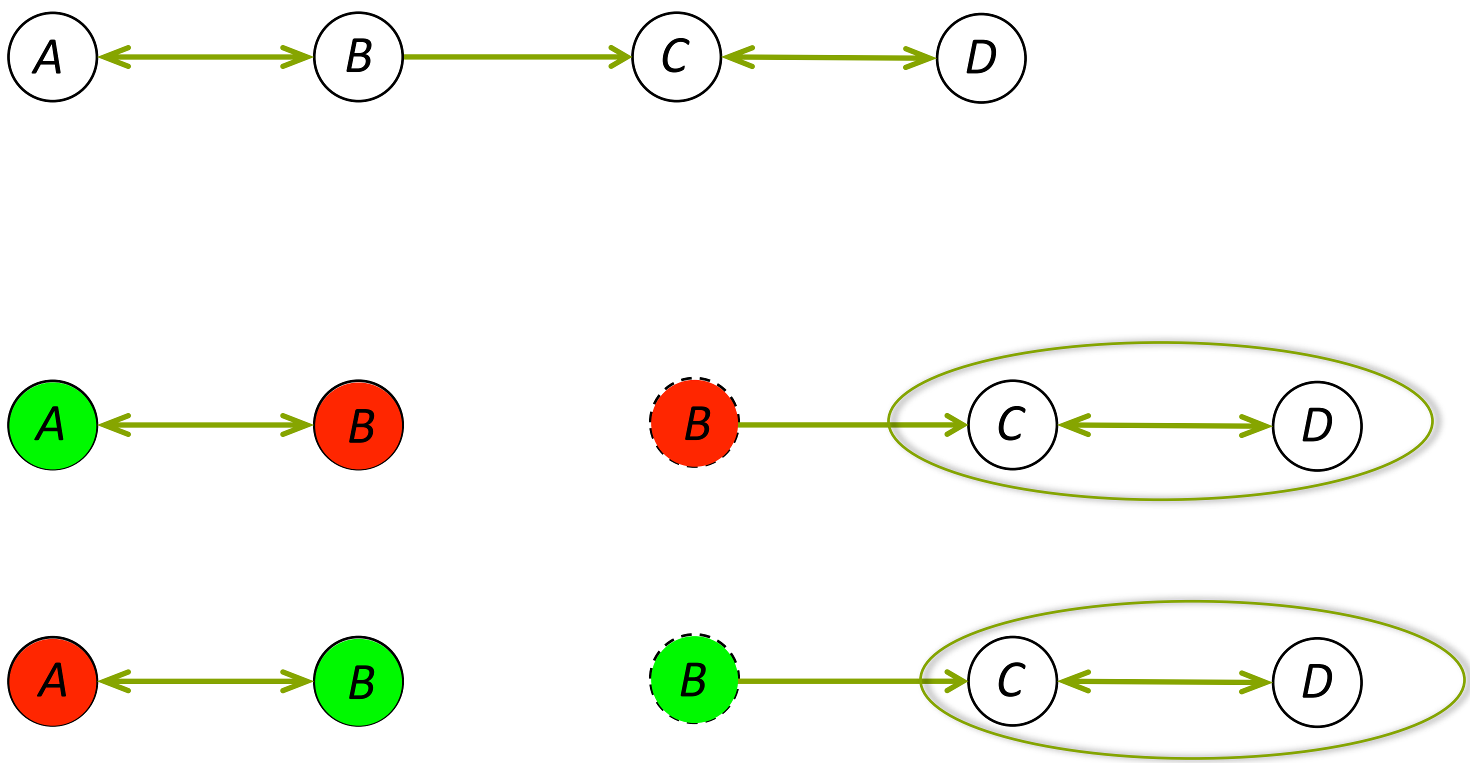
Figure 4.3 $F_{4.3}$ **and its sub-frameworks.**

Semantics of a sub-framework

- Let (C, R_C) be a sub-framework of (*Args*, *attacks*) (called a conditioning sub-framework), s.t. $S^- \subseteq C$. According to each labelling of (C, R_C), $(S \cup S^-, R_S \cup I_S)$ is partially labelled, and therefore called a partially labelled sub-framework (or briefly, PLSF).

Definition 4.3 (Partially labelled sub-framework). Let $(B \cup B^-, R_B \cup I_B)$ and (C, R_C) be sub-frameworks of $F = (A, R)$, such that $B^- \subseteq C$. Let $\mathfrak{L}$ be a labelling of (C, R_C). We call $(B \cup B^-, R_B \cup I_B)^{\mathfrak{L}}$ a partially labelled sub-framework, denoting that the labels of arguments in B^- conform to $\mathfrak{L}$.

Partial labelled sub-framework



Semantics of a sub-framework (Cont.)

- The labellings of a partially labelled sub-framework are then defined on the basis of the corresponding notions of a standard argumentation framework.

Definition 4.4 (Labelling of a partially labelled sub-framework). Based on Definition 4.3, a *labelling* of $(B \cup B^{-}, R_B \cup I_B)^{\mathfrak{L}}$ is defined as a total function

$$\mathfrak{L}' : B \cup B^{-} \mapsto \{\text{IN, OUT, UNDEC}\}$$

such that for all $\alpha \in B^{-}$, $\mathfrak{L}'(\alpha) = \mathfrak{L}(\alpha)$.

Definition 4.5 (Legal/illegal labelling of a PLSF). Based on Definition 4.3, let $\mathfrak{L}'$ be a labelling of $(B \cup B^{-}, R_B \cup I_B)^{\mathfrak{L}}$. For all $\alpha \in B$,

- α is legally IN in $\mathfrak{L}'$ with respect to $\mathfrak{L}$ if and only if α is labelled IN in $\mathfrak{L}'$ and for all $\beta \in B$, if $(\beta, \alpha) \in R_B$ then β is labelled OUT in $\mathfrak{L}'$; and for all $\beta \in B^{-}$, if $(\beta, \alpha) \in I_B$ then β is labelled OUT in $\mathfrak{L}$ (and so in $\mathfrak{L}'$);
- α is legally OUT i [...] in $\mathfrak{L}'$ and there exists $\beta \in B$, such [...] xists $\beta \in B^{-}$, such that $(\beta, \alpha) \in$ [...]
- α is legally UNDE [...] NDEC in $\mathfrak{L}'$ and there exists no $\beta \in$ [...] d IN in $\mathfrak{L}$ or $\mathfrak{L}'$, and it is not the case that: for all $\beta \in B \cup B^{-}$, if $(\beta, \alpha) \in R_B \cup I_B$ then β is labelled OUT in $\mathfrak{L}$ or $\mathfrak{L}'$;
- For $l \in \{$ IN, OUT, UNDEC$\}$, α is illegally l in $\mathfrak{L}'$ with respect to $\mathfrak{L}$ if and only if α is labelled l in $\mathfrak{L}'$, but it is not legally l in $\mathfrak{L}'$ with respect to $\mathfrak{L}$.

The status of arguments is evaluated locally.

Semantics of a sub-framework (Cont.)

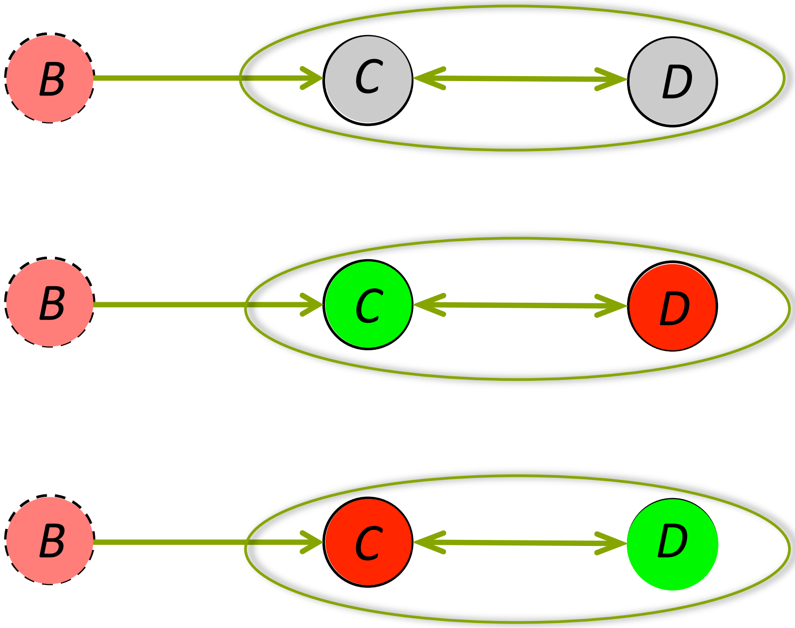
Definition 4.6 (Labelling-based semantics of a PLSF). Based on Definition 4.3, let $\mathfrak{L}'$ be a labelling of $(B \cup B^-, R_B \cup I_B)^{\mathfrak{L}}$.

- $\mathfrak{L}'$ is called an *admissible labelling* with respect to $\mathfrak{L}$, if and only if the following two conditions hold:
  - $\mathfrak{L}$ is an admissible labelling; and
  - each argument in B that is labelled IN in $\mathfrak{L}'$ is legally IN in $\mathfrak{L}'$ with respect to $\mathfrak{L}$, and each argument in B that is labelled OUT in $\mathfrak{L}'$ is legally OUT in $\mathfrak{L}'$ with respect to $\mathfrak{L}$.
- $\mathfrak{L}'$ is called a *complete labelling* with respect to $\mathfrak{L}$, if and only if the following two conditions hold:
  - $\mathfrak{L}$ is a complete labelling; and
  - $\mathfrak{L}'$ is an admissible labelling with respect to $\mathfrak{L}$ and each argument in B that is labelled UNDEC in $\mathfrak{L}'$ is legally UNDEC in $\mathfrak{L}'$ with respect to $\mathfrak{L}$.
- $\mathfrak{L}'$ is called a *preferred labelling* with respect to $\mathfrak{L}$, if and only if the following two conditions hold:
  - $\mathfrak{L}$ is a preferred labelling; and
  - $\mathfrak{L}'$ is a complete labelling with respect to $\mathfrak{L}$, and $in(\mathfrak{L}')$ is maximal (with respect to set-inclusion).

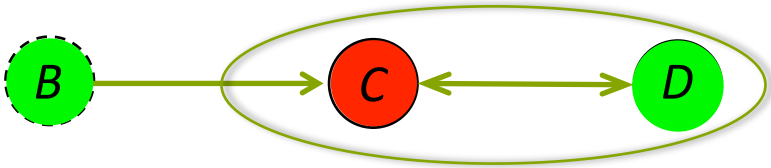
A LOAL PROPERTY

Semantics of a sub-framework (Exp.)

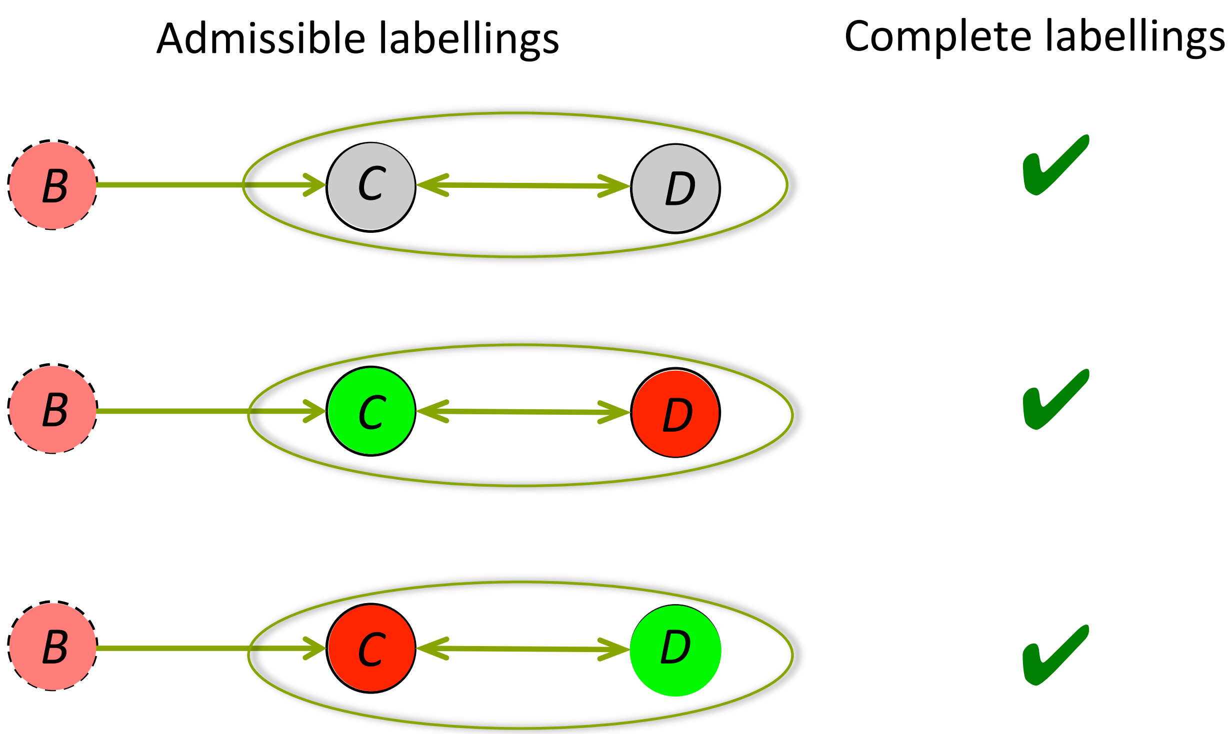
Admissible labellings



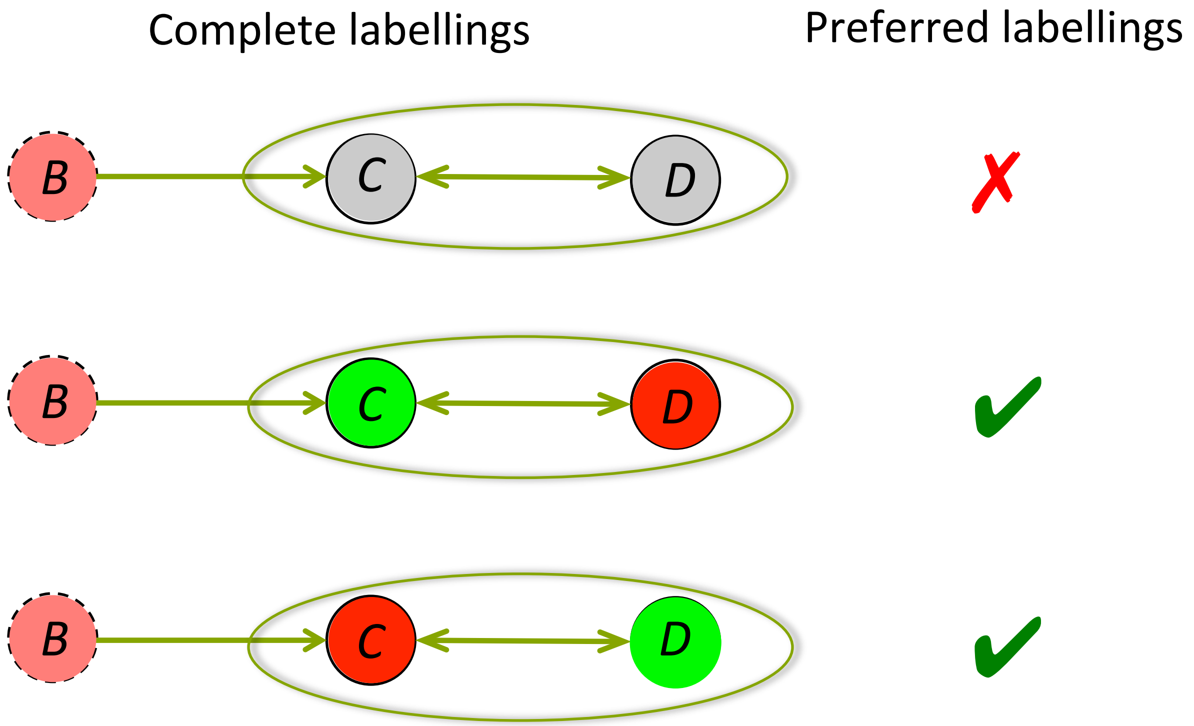
Admissible/complete/grounded/
preferred labelling



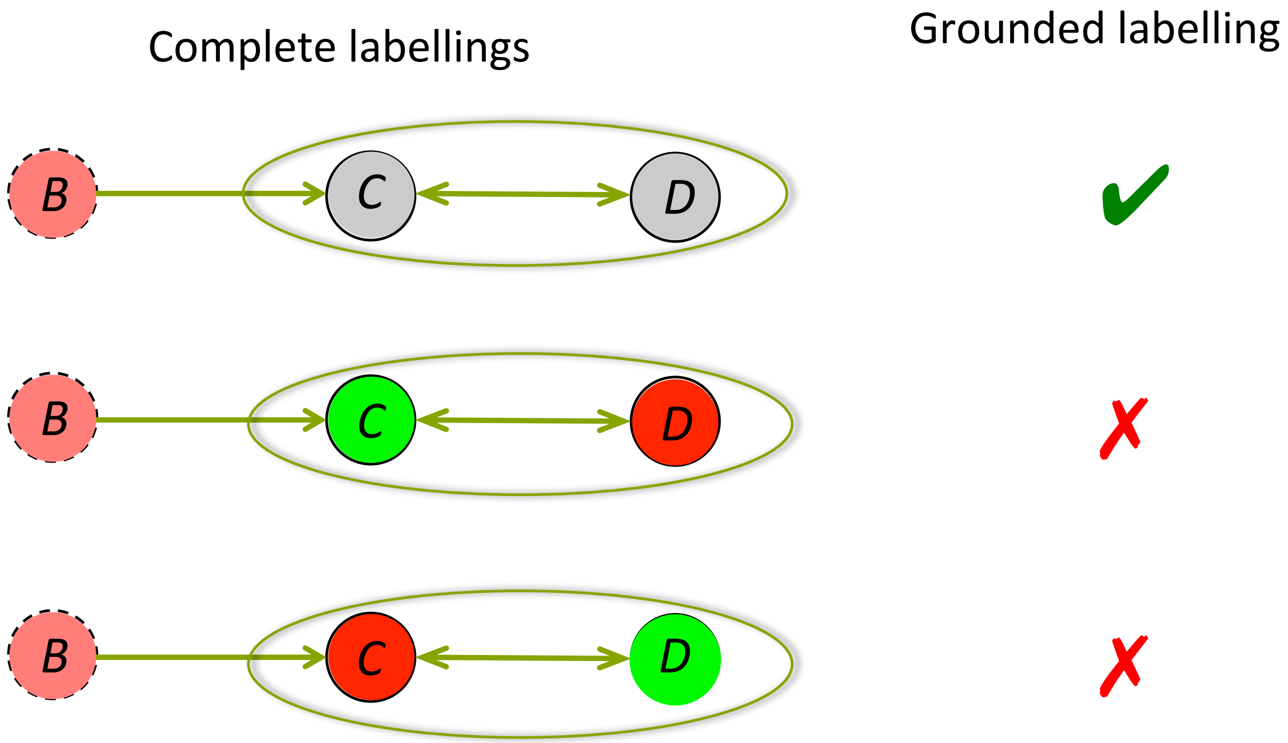
Semantics of a sub-framework (Cont.)



Semantics of a sub-framework (Cont.)

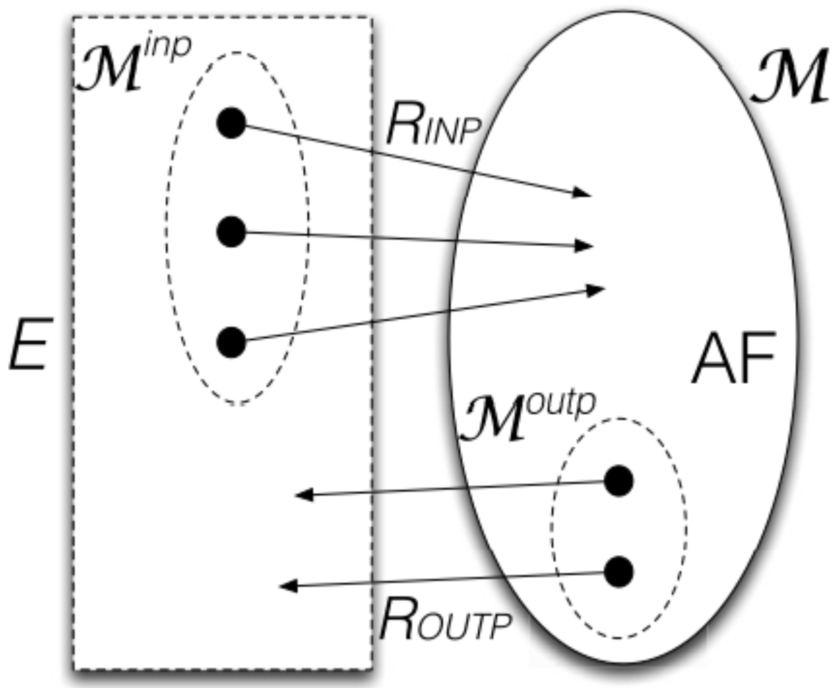


Semantics of a sub-framework (Cont.)



Argumentation multipole [Baroni et al 2014]

Definition 28. An Argumentation Multipole (or, briefly, multipole) $\mathscr{M}$ w.r.t. a set E is a tuple (AF, R_{INP}, R_{OUTP}), where letting $AF = (Ar, att)$ it holds that $Ar \cap E = \emptyset$, $R_{INP} \subseteq E \times Ar$, and $R_{OUTP} \subseteq Ar \times E$. Extending the notation introduced in Definition 10, we denote as $\mathscr{M}^{\text{inp}}$ the set $\{A \in E \mid \exists B \in Ar, (A, B) \in R_{INP}\}$, i.e. including the arguments of E which attack Ar through R_{INP}. Moreover, we denote as $\mathscr{M}^{\text{outp}}$ the set $\{A \in Ar \mid \exists B \in E, (A, B) \in R_{OUTP}\}$, i.e. including the arguments of AF attacking E through R_{OUTP}.



Beishui Liao 15-6-2

Outline

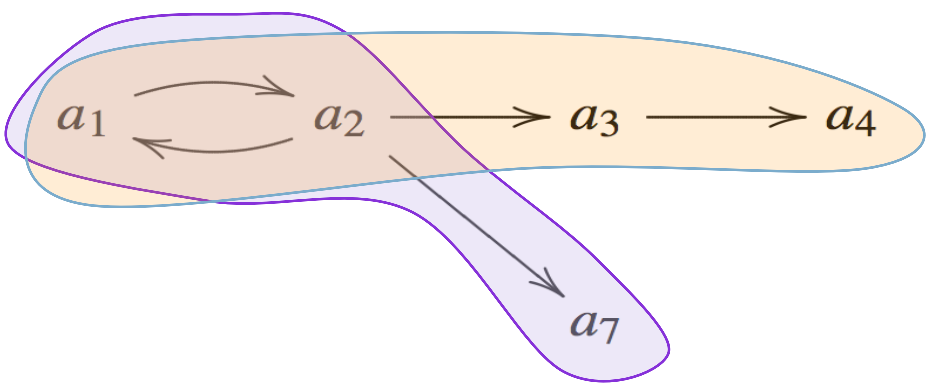
- Background: Dung's abstract argumentation
- Why modularity is important?
- Components: sub-framework vs argumentation multipole
- Incremental computation/argumentation dynamics based on sub-frameworks
- Semantics interchanibility based on argumentation multipole

Incremental computation (from local semantics to global semantics)

Definition 5.3 (Combined extensions of two unconditioned sub-frameworks). Let (B_1, R_{B_1}) and (B_2, R_{B_2}) be two unconditioned sub-frameworks of an argumentation framework $F = (A, R)$, and $Int = B_1 \cap B_2$. Let σ be a semantics under which every argumentation framework has at least one extension. The set of combined extensions of $(B_1 \cup B_2, R_{B_1} \cup R_{B_2})$, denoted as $CombExt_\sigma((B_1 \cup B_2, R_{B_1} \cup R_{B_2}))$, is defined as:

$$CombExt_\sigma((B_1 \cup B_2, R_{B_1} \cup R_{B_2})) = \{E_1 \cup E_2 \mid E_1 \in \mathcal{E}_\sigma((B_1, R_{B_1})) \wedge E_2 \in \mathcal{E}_\sigma((B_2, R_{B_2})) \wedge (E_1 \cap Int = E_2 \cap Int)\}$$

Proposition 5.2. *Let (B_1, R_{B_1}) and (B_2, R_{B_2}) be two unconditioned sub-frameworks of an argumentation framework $F = (A, R)$, and $Int = B_1 \cap B_2$. For each $\sigma \in \{adm, co, gr, pr\}$, it holds that: $CombExt_\sigma((B_1 \cup B_2, R_{B_1} \cup R_{B_2})) = \mathcal{E}_\sigma((B_1 \cup B_2, R_{B_1} \cup R_{B_2}))$.*



$B_1 = \{a_1, a_2, a_3, a_4\}$

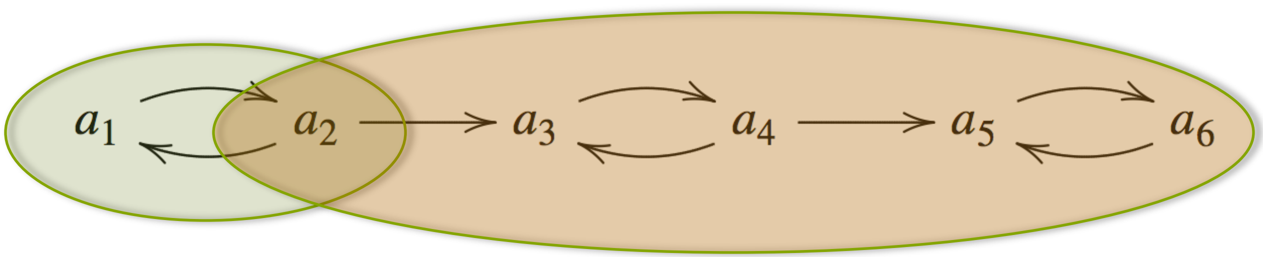
$B_2 = \{a_1, a_2, a_7\}$

$Int = \{a_1, a_2\}$

Incremental computation (from local semantics to global semantics)

Definition 5.4 (Combined extensions of a conditioned sub-framework and those of an unconditioned sub-framework). Let $F = (A, R)$ be an argumentation framework, (C, R_C) and $(B \cup B^-, R_B \cup I_B)$ be the sub-frameworks of F, in which $B^- \subseteq C$. The set of *combined extensions* of $(B \cup C, R_{B \cup C})$, denoted as $CombExt_\sigma((B \cup C, R_{B \cup C}))$, is defined as follows:

$$CombExt_\sigma((B \cup C, R_{B \cup C})) = \{E_1 \cup E_2 \mid E_1 \in \mathscr{E}_\sigma((C, R_C)) \wedge E_2 \in \mathscr{E}_\sigma((B \cup B^-, R_B \cup I_B)^{E_1})\}$$



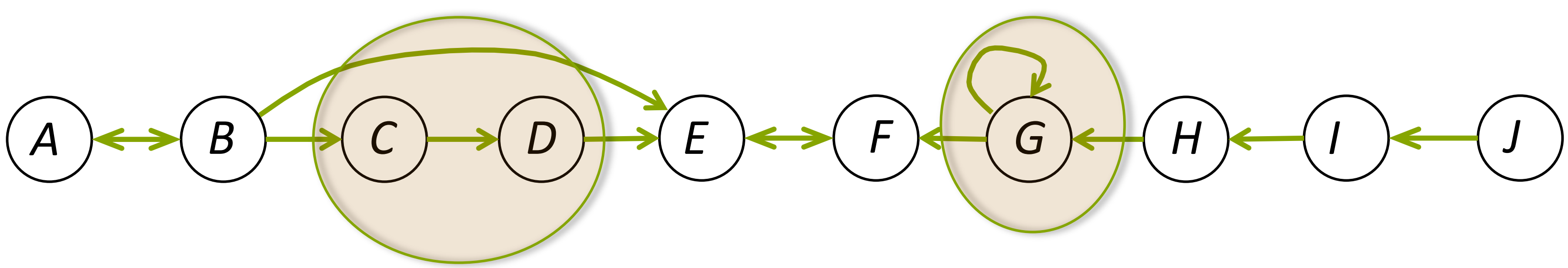
$B = \{a_3, a_4, a_5, a_6\}$

$C = \{a_1, a_2\}$

Incremental computation (from local semantics to global semantics)

Definition 5.6 (Combined labellings of two conditioned sub-frameworks). Let $(B_1 \cup B_1^-, R_{B_1} \cup I_{B_1})$ and $(B_2 \cup B_2^-, R_{B_2} \cup I_{B_2})$ be conditioned sub-frameworks of $F = (A, R)$, such that $B_1 \cap B_2 = \emptyset$, $B_1^- \cap B_2 = \emptyset$ and $B_1 \cap B_2^- = \emptyset$. Let $B = B_1 \cup B_2$. So, $(B \cup B^-, R_B \cup I_B)$ is a combined framework of $(B_1 \cup B_1^-, R_{B_1} \cup I_{B_1})$ and $(B_2 \cup B_2^-, R_{B_2} \cup I_{B_2})$. Let (C_1, R_{C_1}) and (C_2, R_{C_2}) be two unconditioned sub-frameworks of F, such that $B_1^- \subseteq C_1$ and $B_2^- \subseteq C_2$. Let σ be a semantics, under which every argumentation framework has at least one labelling. For all $\mathfrak{L}_1 \in \mathscr{L}_\sigma((C_1, R_{C_1}))$, $\mathfrak{L}_2 \in \mathscr{L}_\sigma((C_2, R_{C_2}))$, let $\mathfrak{L} = \mathfrak{L}_1 + \mathfrak{L}_2$.

$$
\begin{aligned}
CombLab_\sigma((B \cup B^-, R_B \cup I_B)^{\mathfrak{L}}) = \{ & \mathfrak{L}_1' + \mathfrak{L}_2' \mid \\
& \mathfrak{L}_1' \in \mathscr{L}_\sigma((B_1 \cup B_1^-, R_{B_1} \cup I_{B_1})^{\mathfrak{L}_1}) \\
& \wedge \mathfrak{L}_2' \in \mathscr{L}_\sigma((B_2 \cup B_2^-, R_{B_2} \cup I_{B_2})^{\mathfrak{L}_2})\}
\end{aligned}
$$



An incremental computation approach [Liao 2013]

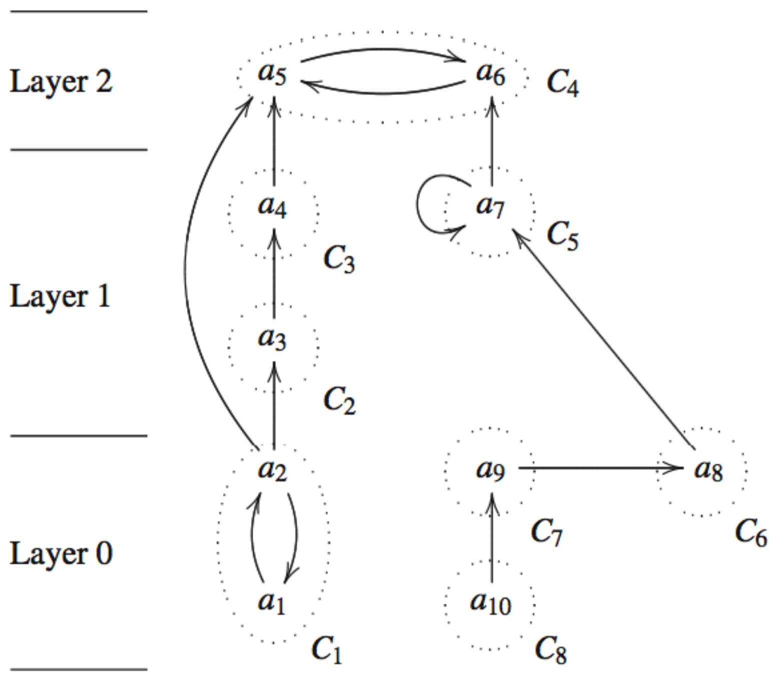


Figure 6.3 Layered decomposition of the strongly connected components of $F_{6.2}$.

An incremental computation approach [Liao 2013]

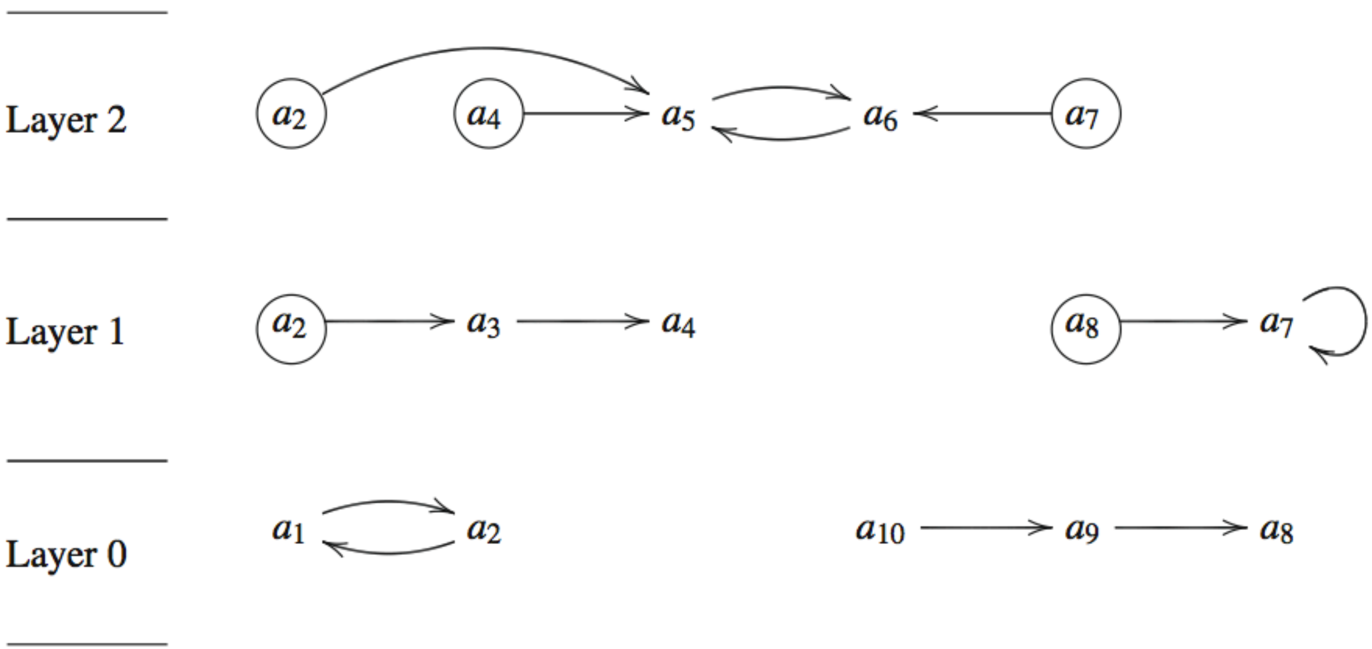
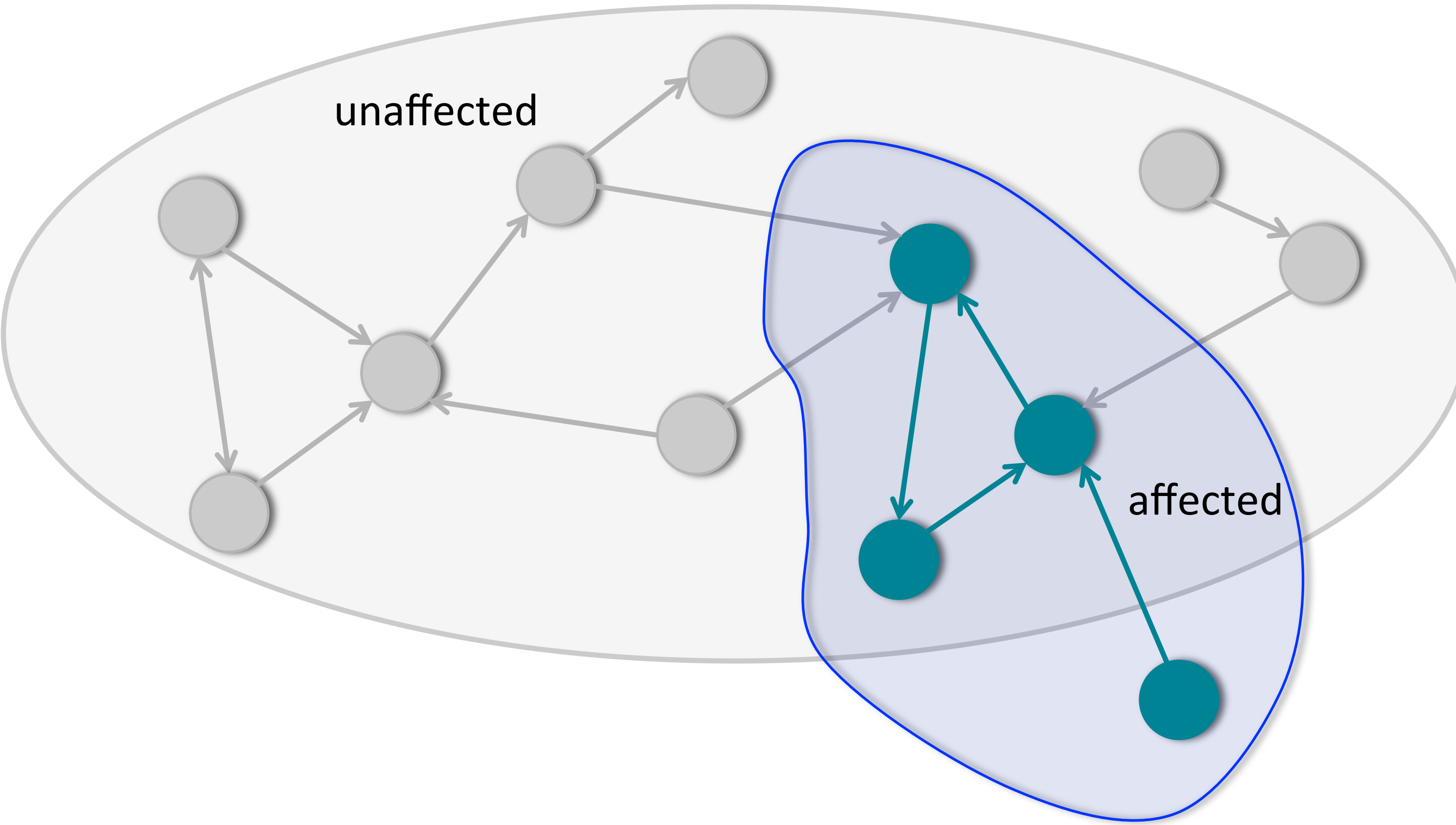


Figure 6.4 A decomposition of $F_{6.2}$.

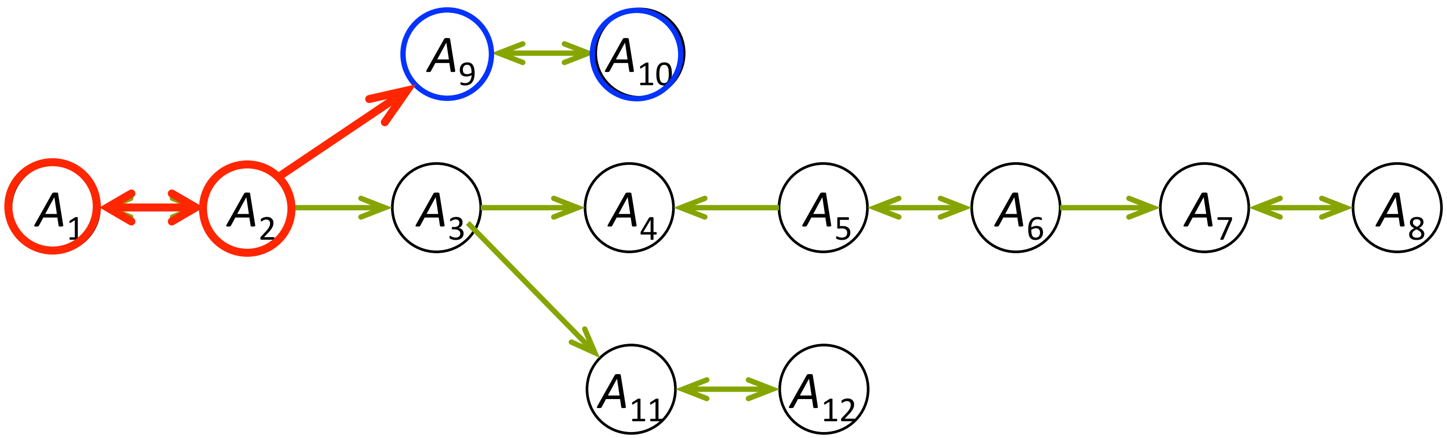
Table 6.1 Average results of the two algorithms.

Ratio	#nodes	#SCCS	Our Alg.		MC Alg.	
			Time (seconds)	#timeout	Time (seconds)	#timeout
	100	95	0.013	0	1.444	0
	120	114	0.008	0	0.022	3
1:1	140	135	0.012	0	0.055	3
	160	156	0.015	0	5.033	1
	180	**171**	**0.017**	**0**	**1.656**	**4**
	200	192	0.019	0	0.328	1
	15	**10**	**0.451**	**0**	**9.818**	**0**
	17	11	0.015	0	54.236	2
1.5:1	19	11	0.001	0	57.379	2
	21	12	0.002	1	1.775	3
	23	16	4.496	0	40.126	2
	25	18	0.002	0	2.933	3
	15	7	2.133	0	95.911	4
	17	7	1.501	1	12.971	8
2:1	19	9	0	3	0.576	5
	21	10	0.001	1	0.59	6
	23	9	0.451	3	24.94	7
	25	10	0.119	3	34.87	4
	8	2	0.026	0	0.050	0
	9	3	0.122	0	0.153	0
3:1	10	3	0.342	0	0.970	0
	11	3	1.685	0	8.185	0
	12	3	11.510	0	72.841	0

Argumentation dynamics [Liao 2011]



Partial semantics of argumentation [Liao 2013]



When querying the status of arguments in $\{A_9, A_{10}\}$, …

The set of relevant arguments is $\{A_9, A_{10}, A_1, A_2\}$, so we need only to compute the semantics of the sub-framework induced by this set.

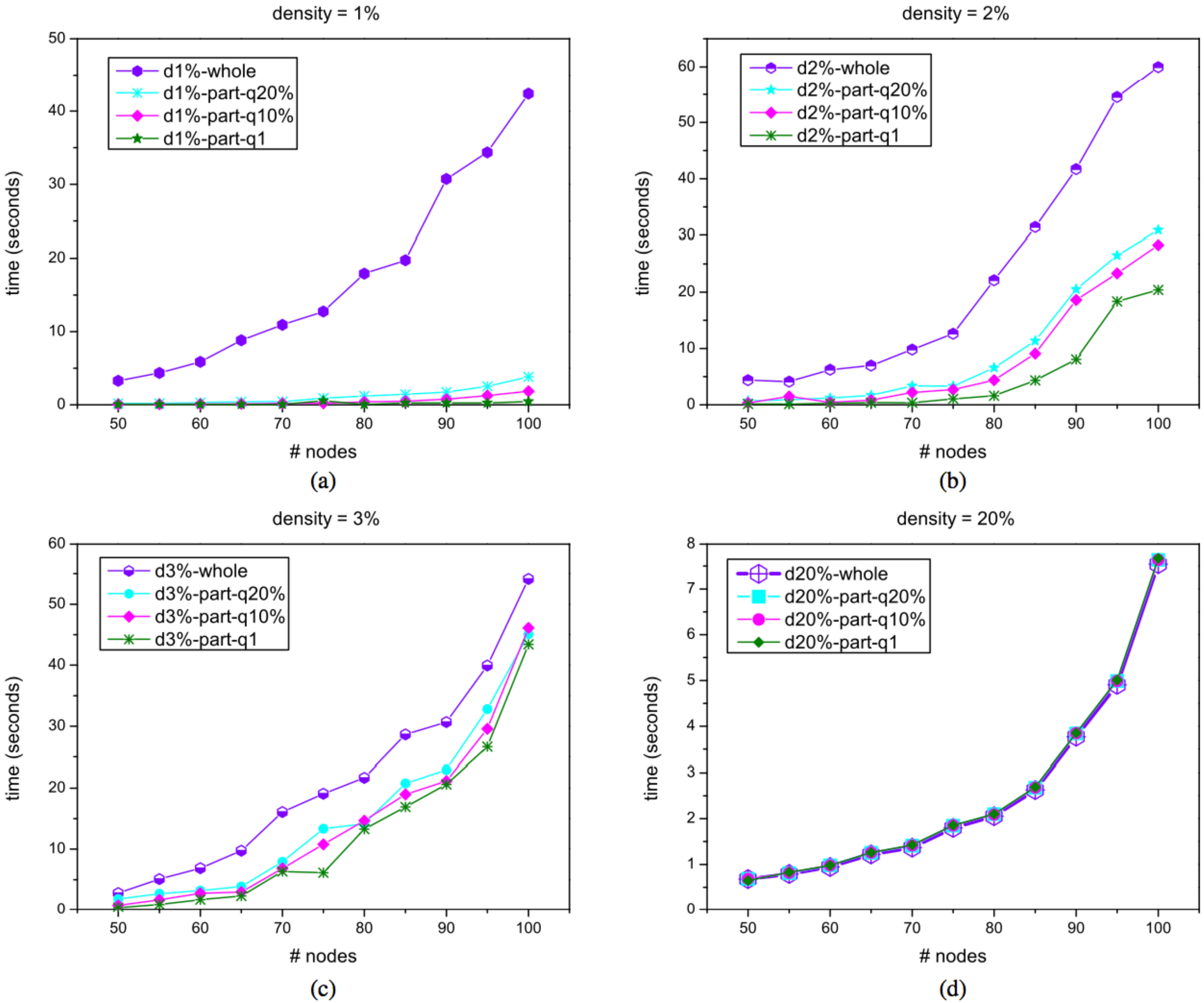


Figure 8.4 Plots showing the average execution time (revised by dropping some cases where the execution time of computing the extensions of the whole argumentation framework is 3 times more than the average execution time in remaining cases) when the edge density of argumentation frameworks is 1%, 2%, 3% and 20%, respectively.

$$density = \frac{|R|}{|A| * (|A| - 1)} * 100\%$$

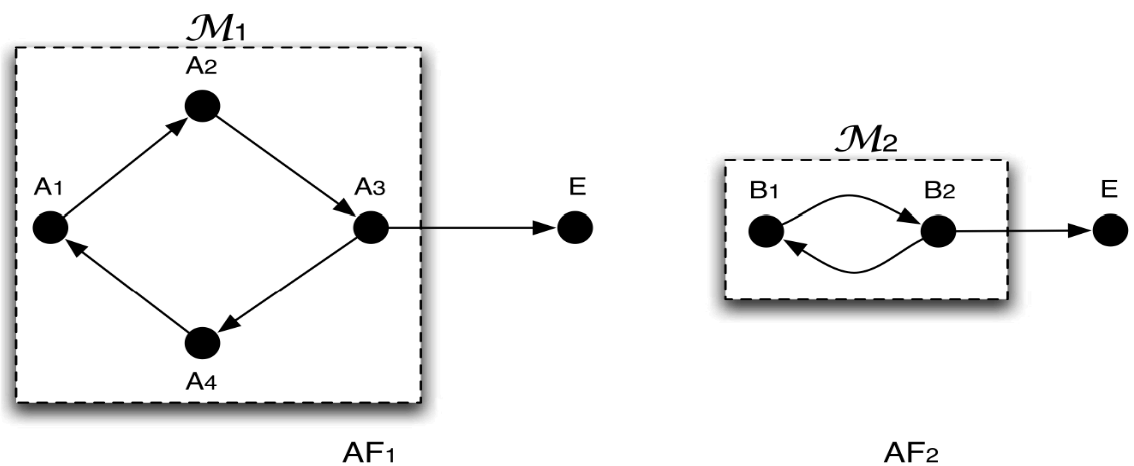
Outline

- Background: Dung's abstract argumentation
- Why modularity is important?
- Components: sub-framework vs argumentation multipole
- Incremental computation/argumentation dynamics based on sub-framework
- Semantics interchangeability based on argumentation multipole

Semantics interchangeability [Baroni et al 2014]

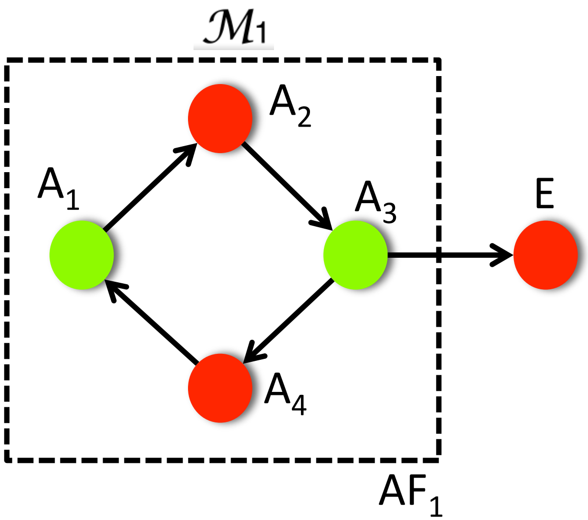
- Input/Output equivalence of argumentation multipoles

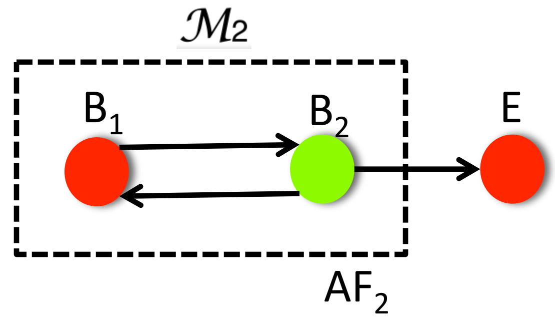
Given a semantics **S**, two multipoles wrt the same set *E* are **S**-equivalent if for any labelling of *E* they have the same effect on *E*.

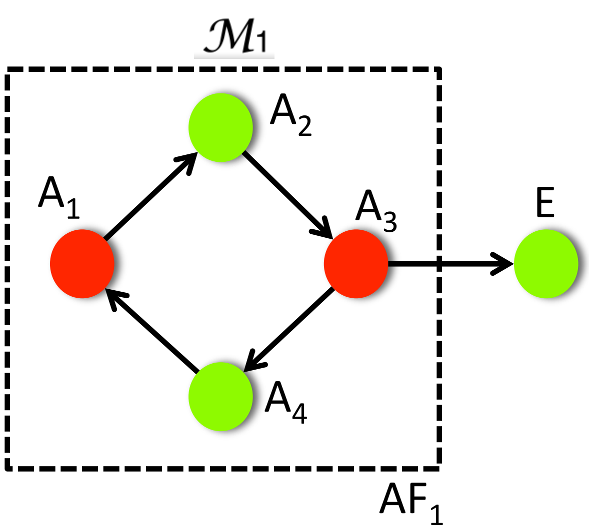


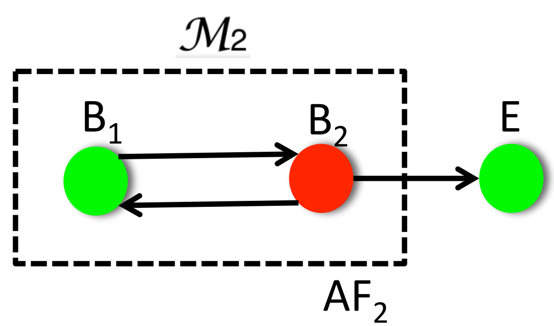
$\mathscr{M}_1$ and $\mathscr{M}_2$ are **GR**-equivalent and **PR**-equivalent

$\mathscr{M}_1$ and $\mathscr{M}_2$ are **PR**-equivalent

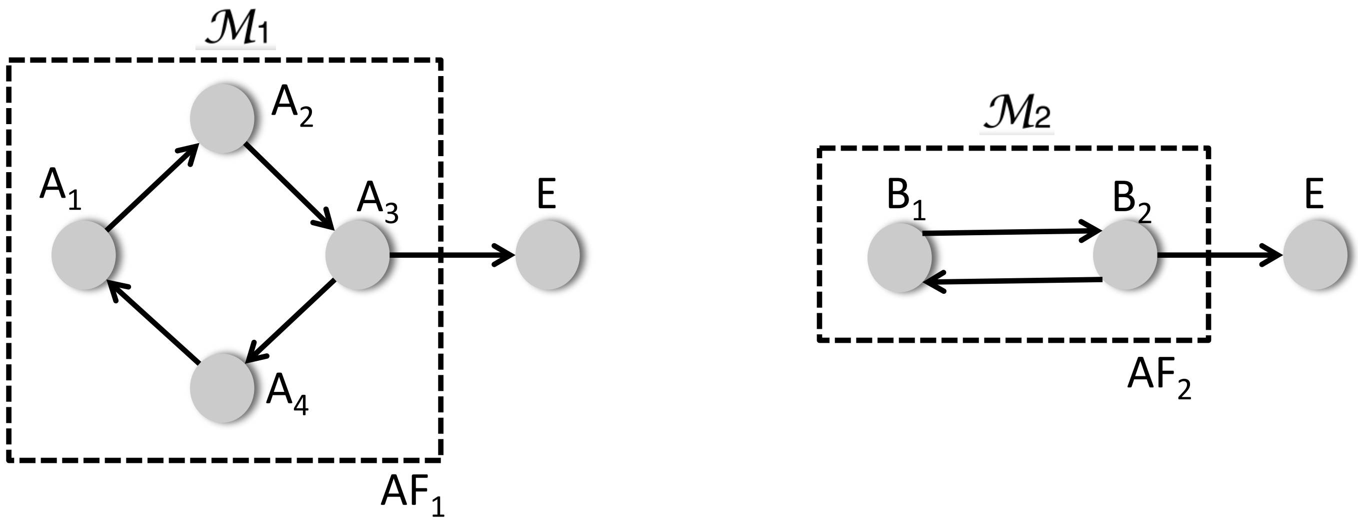








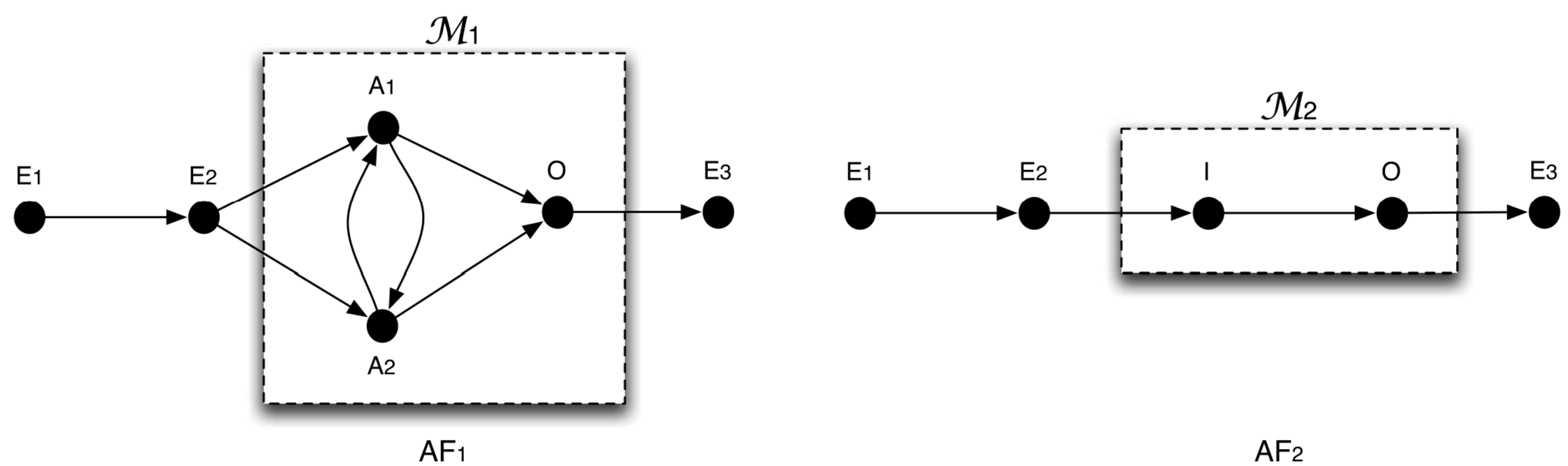
$\mathscr{M}_1$ and $\mathscr{M}_2$ are **GR**-equivalent



Semantics interchangeability [Baroni et al 2014]

- Input/Output equivalence of argumentation multipoles
- Replacement within an AF

PR-legitimate replacement



Conclusions and future work

- As a methodology, modularity of argumentation facilitates several lines of research: efficient computation, argumentation dynamics, local properties of argumentation, replacement and summarization, etc.
- The existing work is oriented to Dung's abstraction argumentation. It is possible to extend the existing theories and methods to some other sub-fields of argumentation (for instance, structured argumentation, probabilistic argumentation, etc).

References

P. Baroni, G. Boella, F. Cerutti, M. Giacomin, L. van der Torre, S.Villata: On the Input/Output behavior of argumentation frameworks. Artificial Intelligence 217 (2014) 144-197.

M. Caminada, D. Gabbay, A logical account of formal argumentation, Studia Logica 93 (2–3) (2009) 109–145.

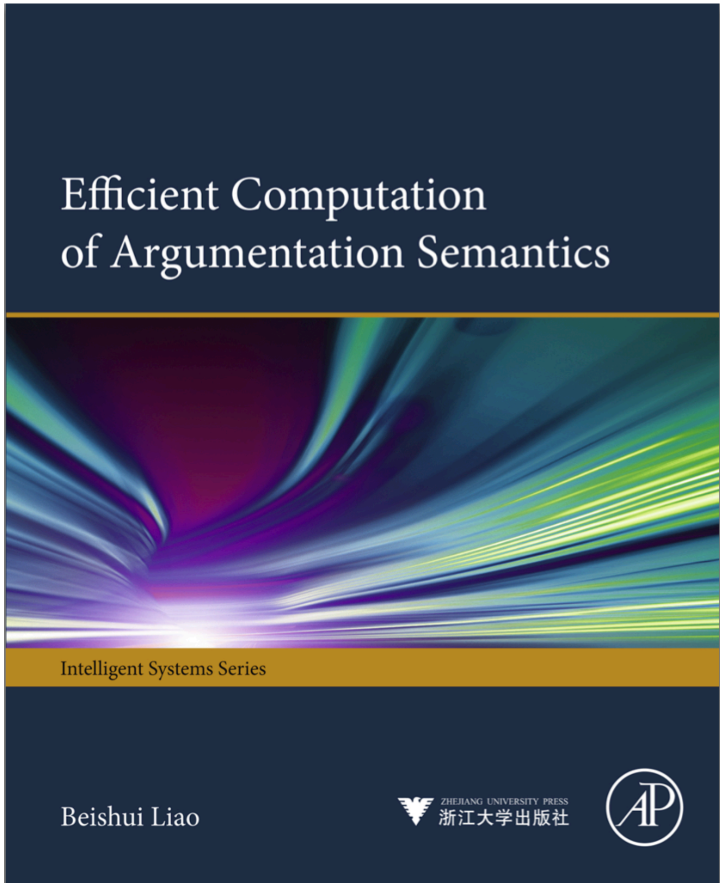
P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artificial Intelligence 77 (1995) 321–357.

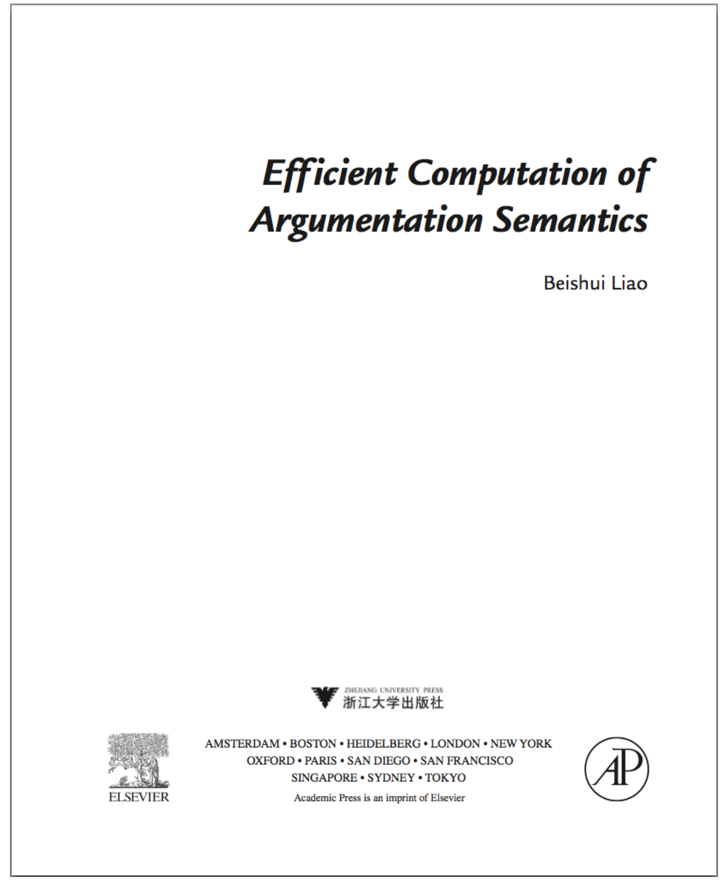
Beishui Liao, Efficient Computation of Argumentation Semantics. Intelligent systems series, Academic Press 2014, ISBN 978-0-12-410406-8, pp. I-X, 1-137.

Beishui Liao, Huaxin Huang, Partial Semantics of Argumentation: Basic Properties and Empirical Results. Journal of Logic and Computation (2013) 23(3): 541-562.

Beishui Liao, Toward Incremental Computation of Argumentation Semantics: A Decomposition-based Approach. Annals of Mathematics and Artificial Intelligence, 67(2013): 319-358.

Beishui Liao, Li Jin, Robert C, Koons.Dynamics of Argumentation Systems: A Division-Based Method. Artificial Intelligence, 175 (2011) 1790-1814.




Efficient Computation of Argumentation Semantics

Beishui Liao

ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK
OXFORD • PARIS • SAN DIEGO • SAN FRANCISCO
SINGAPORE • SYDNEY • TOKYO
Academic Press is an imprint of Elsevier

AP

Thanks!