Argumentation Meets Computational Social Choice

PART I: Preservation of Semantic Properties

Verifying Semantics in Incomplete AFs

PART II: Gradual Acceptance in Argumentation

PART III: Rationalization

Discussion and Outlook

Dorothea Baumeister, **Daniel Neugebauer**, and Jörg Rothe

July 14, 2018

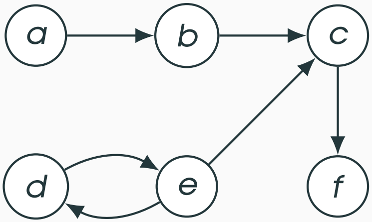
Tutorial 23 at IJCAI-ECAI-18 in Stockholm, Sweden

NRW-FORTSCHRITTSKOLLEG ONLINE-PARTIZIPATION

HEINRICH HEINE UNIVERSITÄT DÜSSELDORF

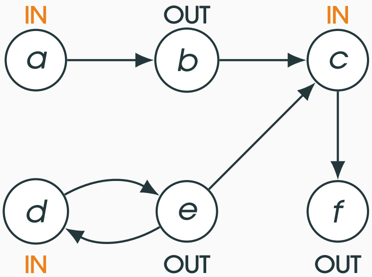
The Granularity of Argument Acceptance

Preferred extensions:



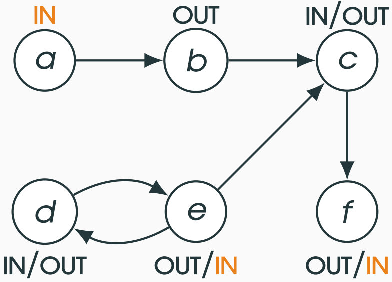
Preferred extensions:

$\{a, c, d\}$



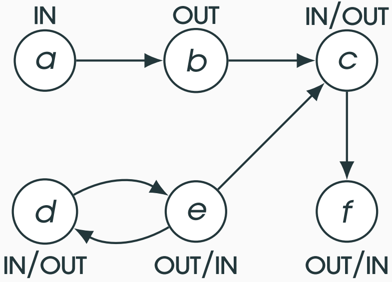
Preferred extensions:

$\{a, c, d\}$, $\{a, e, f\}$



Preferred extensions:

$\{a, c, d\}, \{a, e, f\}$

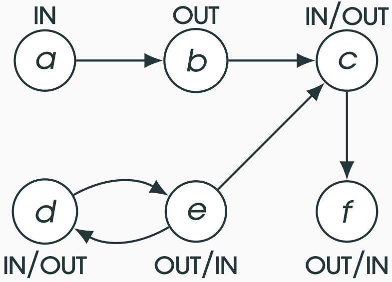


Argument acceptance using the preferred semantics:

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

Preferred extensions:

$\{a, c, d\}, \{a, e, f\}$

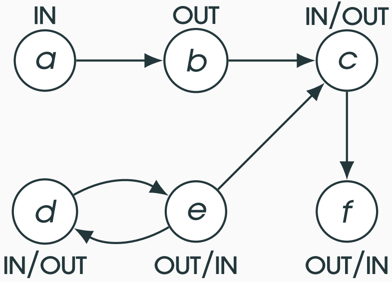


Argument acceptance using the preferred semantics:

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

Preferred extensions:

$\{a, c, d\}, \{a, e, f\}$

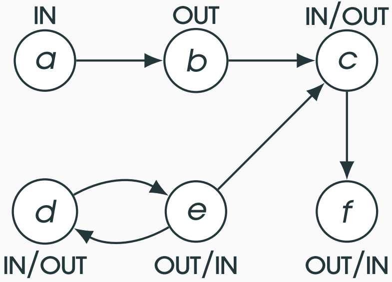


Argument acceptance using the preferred semantics:

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

Preferred extensions:

$\{a, c, d\}, \{a, e, f\}$



Argument acceptance using the preferred semantics:

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

The Granularity of Argument Acceptance

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

The Granularity of Argument Acceptance

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

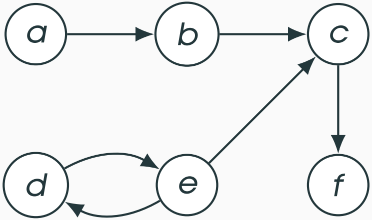
. . . Why not more?

Skept. accepted	Cred. accepted	Unaccepted
a	c, d, e, f	b

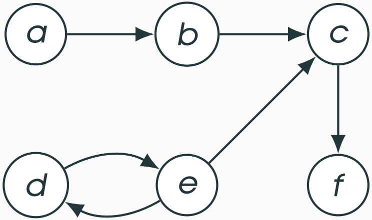
. . . Why not more?

$$a_1 \gtrsim a_2 \gtrsim a_3 \gtrsim a_4 \gtrsim a_5 \gtrsim a_6$$

Definition. A ranking semantics **s** maps an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ to a complete and transitive preorder $\leqslant_{\mathbf{s}}$ (called a ranking) on $\mathcal{A}$.

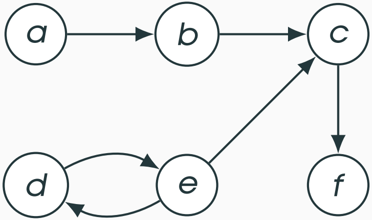


Definition. A ranking semantics **s** maps an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ to a complete and transitive preorder $\leqslant_{\mathbf{s}}$ (called a ranking) on $\mathcal{A}$.



$$a > c > d \sim e > f > b$$

Definition. A ranking semantics **s** maps an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ to a complete and transitive preorder $\preccurlyeq_{\mathbf{s}}$ (called a ranking) on $\mathcal{A}$.

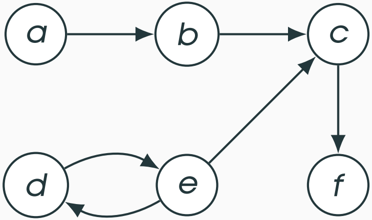


$a > c > d \sim e > f > b$

$a > c \sim d \sim e \sim f > b$

Ranking Semantics

Definition. A ranking semantics **s** maps an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ to a complete and transitive preorder $\leqslant_{\mathbf{s}}$ (called a ranking) on $\mathcal{A}$.

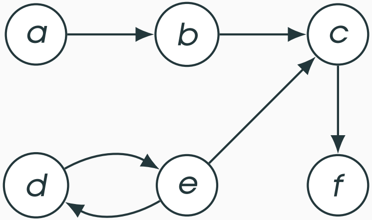


$a > c > d \sim e > f > b$

$a > c \sim d \sim e \sim f > b$

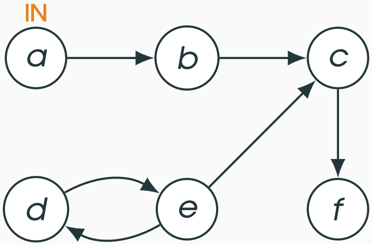
$a \sim d \sim e > c \sim f > b$

Example: Preferred Semantics as a (degenerate) Ranking Semantics



Ranking induced by the preferred semantics:

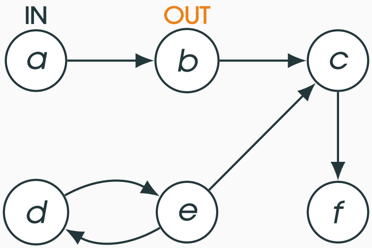
Example: Preferred Semantics as a (degenerate) Ranking Semantics



Ranking induced by the preferred semantics:

$a >$

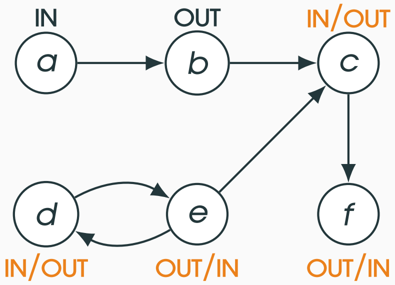
Example: Preferred Semantics as a (degenerate) Ranking Semantics



Ranking induced by the preferred semantics:

$a >$ $> b$

Example: Preferred Semantics as a (degenerate) Ranking Semantics



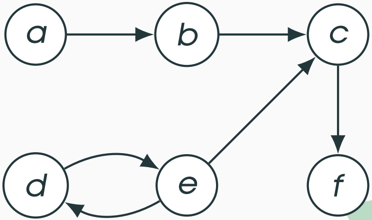
Ranking induced by the preferred semantics:

$$a > d \sim e \sim c \sim f > b$$

Existing Ranking Semantics

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

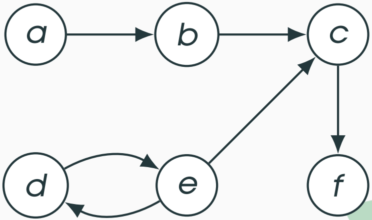


[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$



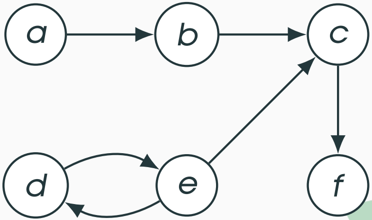
[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$

$$Cat(b) = {}^{1}/_{(1+Cat(a))} = 0.5$$

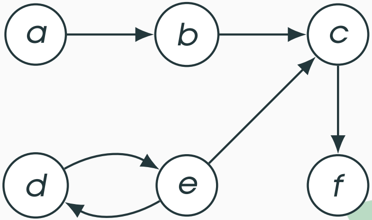


[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$
$$Cat(b) = {}^{1}/_{(1+Cat(a))} = 0.5$$
$$Cat(d) = {}^{1}/_{(1+Cat(e))}$$
$$Cat(e) = {}^{1}/_{(1+Cat(d))}$$

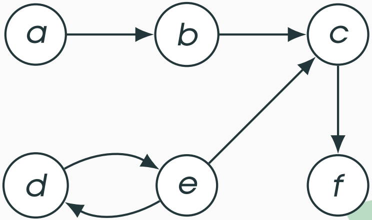


[1] P. Besnard and A. Hunter. “A logic-based theory of deductive arguments”. In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$
$$Cat(b) = {}^1/_{(1+Cat(a))} = 0.5$$
$$Cat(d) = {}^1/_{(1+Cat(e))} \approx 0.618$$
$$Cat(e) = {}^1/_{(1+Cat(d))} \approx 0.618$$

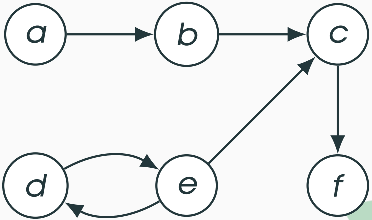


[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b \in \downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$
$$Cat(b) = {}^1/_{(1+Cat(a))} = 0.5$$
$$Cat(d) = {}^1/_{(1+Cat(e))} \approx 0.618$$
$$Cat(e) = {}^1/_{(1+Cat(d))} \approx 0.618$$
$$Cat(c) = {}^1/_{(1+Cat(b)+Cat(e))} \approx 0.472$$

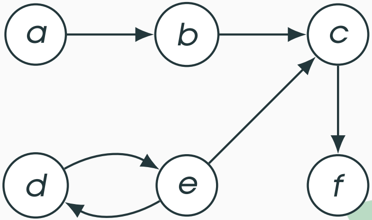


[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Categorizer semantics ranks arguments by values of a categorizer function:

$$Cat(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \frac{1}{1+\sum_{b\in\downarrow(a)} Cat(b)} & \text{otherwise} \end{cases}$$

$$Cat(a) = 1$$
$$Cat(b) = 1/(1+Cat(a)) = 0.5$$
$$Cat(d) = 1/(1+Cat(e)) \approx 0.618$$
$$Cat(e) = 1/(1+Cat(d)) \approx 0.618$$
$$Cat(c) = 1/(1+Cat(b)+Cat(e)) \approx 0.472$$
$$Cat(f) = 1/(1+Cat(c)) \approx 0.679$$

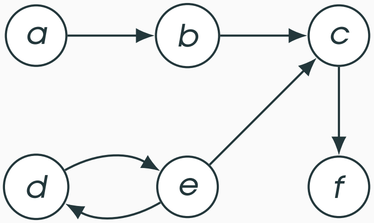


$$a > f > d \sim e > b > c$$

[1] P. Besnard and A. Hunter. "A logic-based theory of deductive arguments". In: *Artificial Intelligence* 128.1 (2001), pp. 203–235.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

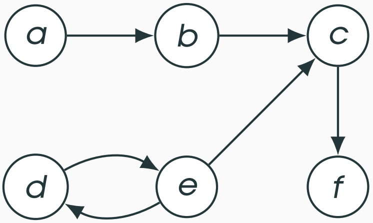
$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$



[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$



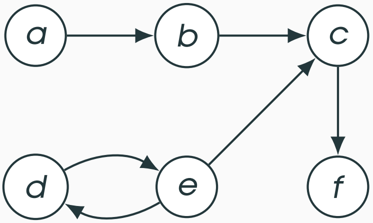
$Bur(a)$		
$Bur(b)$		
$Bur(c)$		
$Bur(d)$		
$Bur(e)$		
$Bur(f)$		

[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$

$Bur(a)$	1
$Bur(b)$	1
$Bur(c)$	1
$Bur(d)$	1
$Bur(e)$	1
$Bur(f)$	1

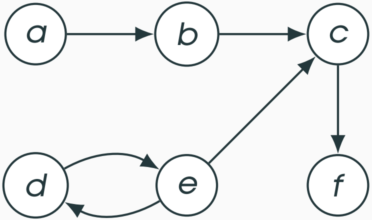


[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$

$Bur(a)$	1	1
$Bur(b)$	1	2
$Bur(c)$	1	3
$Bur(d)$	1	2
$Bur(e)$	1	2
$Bur(f)$	1	2



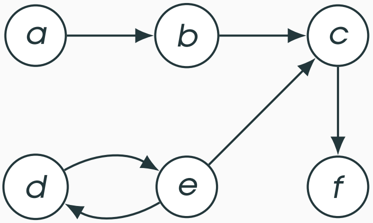
$a >$ $\quad > c$

[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$

$Bur(a)$	1	1	-
$Bur(b)$	1	2	2
$Bur(c)$	1	3	-
$Bur(d)$	1	2	1.5
$Bur(e)$	1	2	1.5
$Bur(f)$	1	2	1.33

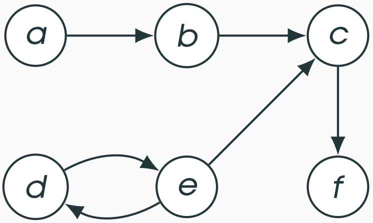


$a > f > \quad > b > c$

[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Burden-based semantics (Bbs) ranks arguments lexicographically by burden numbers:

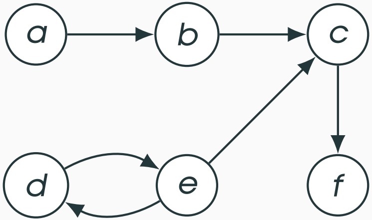
$$Bur_i(a) = \begin{cases} 1 & i = 0 \\ 1 + \sum_{b \in \downarrow(a)} Bur_{i-1}(b)^{(-1)} & \text{otherwise} \end{cases}$$



$Bur(a)$	1	1	-	-
$Bur(b)$	1	2	2	-
$Bur(c)$	1	3	-	-
$Bur(d)$	1	2	1.5	...
$Bur(e)$	1	2	1.5	...
$Bur(f)$	1	2	1.33	-

$$a > f > d \sim e > b > c$$

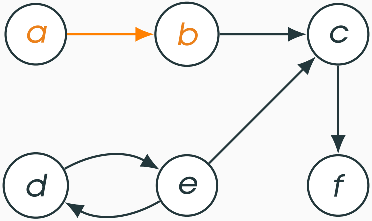
[2] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.



Definition. A directed path in an argumentation framework is an attack branch if it has an odd number of edges and a defense branch if it has an even number (> 0) of edges.

- Examples for attack branches: (a, b) and $(a, b), (b, c), (c, f)$.
- Examples for defense branches: $(a, b), (b, c)$ and $(d, e), (e, d)^*$.

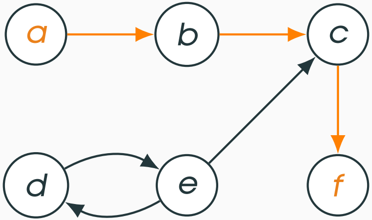
* some authors require paths that start from unattacked arguments



Definition. A directed path in an argumentation framework is an attack branch if it has an odd number of edges and a defense branch if it has an even number (> 0) of edges.

- Examples for attack branches: (a, b) and $(a, b), (b, c), (c, f)$.
- Examples for defense branches: $(a, b), (b, c)$ and $(d, e), (e, d)^*$.

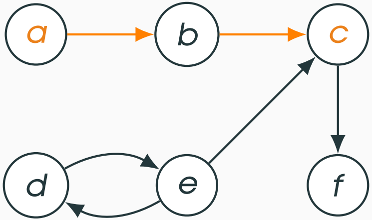
* some authors require paths that start from unattacked arguments



Definition. A directed path in an argumentation framework is an attack branch if it has an odd number of edges and a defense branch if it has an even number (> 0) of edges.

- Examples for attack branches: (a, b) and $(a, b), (b, c), (c, f)$.
- Examples for defense branches: $(a, b), (b, c)$ and $(d, e), (e, d)^*$.

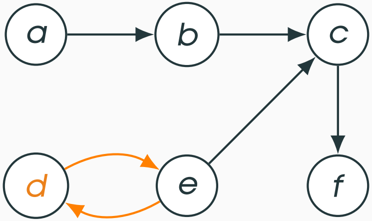
* some authors require paths that start from unattacked arguments



Definition. A directed path in an argumentation framework is an attack branch if it has an odd number of edges and a defense branch if it has an even number (> 0) of edges.

- Examples for attack branches: (a, b) and $(a, b), (b, c), (c, f)$.
- Examples for defense branches: $(a, b), (b, c)$ and $(d, e), (e, d)^*$.

* some authors require paths that start from unattacked arguments

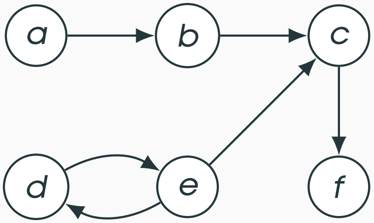


Definition. A directed path in an argumentation framework is an attack branch if it has an odd number of edges and a defense branch if it has an even number (> 0) of edges.

- Examples for attack branches: (a, b) and $(a, b), (b, c), (c, f)$.
- Examples for defense branches: $(a, b), (b, c)$ and $(d, e), (e, d)$*.

* some authors require paths that start from unattacked arguments

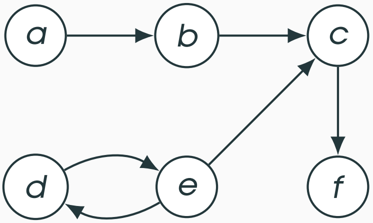
Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).



[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).

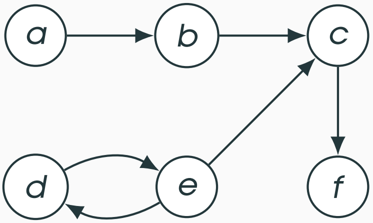
$Dis(a)$
$Dis(b)$
$Dis(c)$
$Dis(d)$
$Dis(e)$
$Dis(f)$



[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).

$Dis(a)$	0
$Dis(b)$	1
$Dis(c)$	2
$Dis(d)$	1
$Dis(e)$	1
$Dis(f)$	1

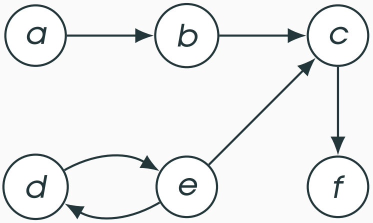


$a >$ $> c$

[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).

$Dis(a)$	0	-
$Dis(b)$	1	0
$Dis(c)$	2	-
$Dis(d)$	1	-1
$Dis(e)$	1	-1
$Dis(f)$	1	-2

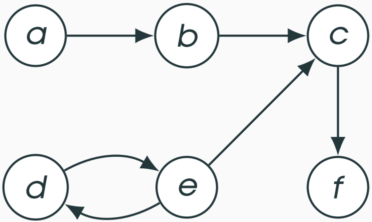


$a > f > \quad > b > c$

[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).

$Dis(a)$	0	-	-
$Dis(b)$	1	0	-
$Dis(c)$	2	-	-
$Dis(d)$	1	-1	1
$Dis(e)$	1	-1	1
$Dis(f)$	1	-2	-

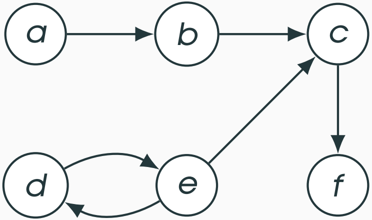


$a > f > \quad > b > c$

[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. Discussion-based semantics (Dbs) ranks by an argument's number of incoming attack and defense branches (lexicographically w.r.t. path length).

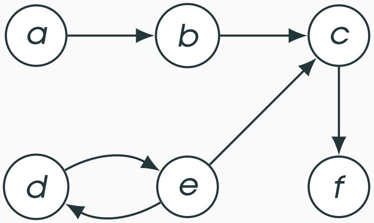
$Dis(a)$	0	-	-	-	-
$Dis(b)$	1	0	-	-	-
$Dis(c)$	2	-	-	-	-
$Dis(d)$	1	-1	1	-1	...
$Dis(e)$	1	-1	1	-1	...
$Dis(f)$	1	-2	-	-	-



$a > f > d \sim e > b > c$

[3] L. Amgoud and J. Ben-Naim. "Ranking-based semantics for argumentation frameworks". In: *Proceedings of the 7th International Conference on Scalable Uncertainty Management*. Springer. 2013, pp. 134–147.

Definition. $Propa_\epsilon$ propagates the weight of an argument iteratively to adjacent arguments.



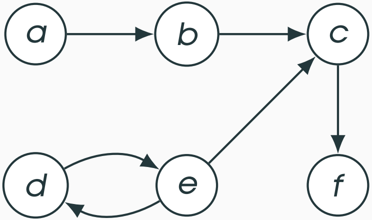
[4]E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^\epsilon(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^\epsilon(a) = P_{i-1}^\epsilon(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^\epsilon(b)$$

$P^{0.75}(a)$
$P^{0.75}(b)$
$P^{0.75}(c)$
$P^{0.75}(d)$
$P^{0.75}(e)$
$P^{0.75}(f)$



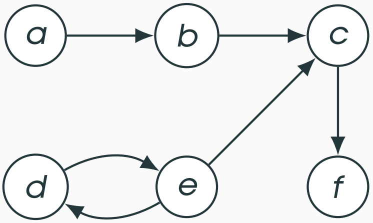
[4] E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1
$P^{0.75}(b)$	0.75
$P^{0.75}(c)$	0.75
$P^{0.75}(d)$	0.75
$P^{0.75}(e)$	0.75
$P^{0.75}(f)$	0.75



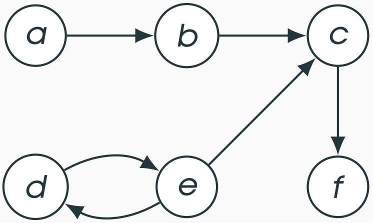
[4] E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1	1
$P^{0.75}(b)$	0.75	-0.25
$P^{0.75}(c)$	0.75	-0.75
$P^{0.75}(d)$	0.75	0
$P^{0.75}(e)$	0.75	0
$P^{0.75}(f)$	0.75	0



[4]E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

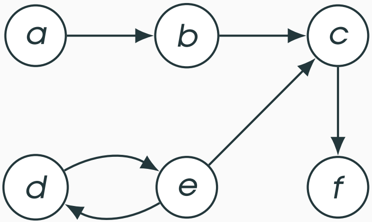
Ranking Semantics

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1	1	...
$P^{0.75}(b)$	0.75	−0.25	−0.25
$P^{0.75}(c)$	0.75	−0.75	1
$P^{0.75}(d)$	0.75	0	0.75
$P^{0.75}(e)$	0.75	0	0.75
$P^{0.75}(f)$	0.75	0	1.5



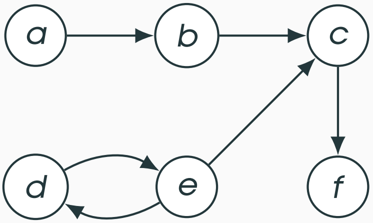
[4]E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1	1	...	...
$P^{0.75}(b)$	0.75	−0.25	−0.25	...
$P^{0.75}(c)$	0.75	−0.75	1	0.25
$P^{0.75}(d)$	0.75	0	0.75	0.25
$P^{0.75}(e)$	0.75	0	0.75	0.25
$P^{0.75}(f)$	0.75	0	1.5	0.25



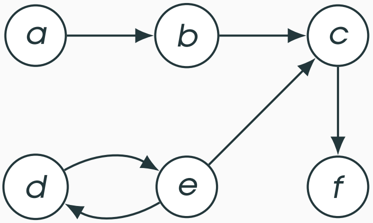
[4]E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1	1	...	...	...
$P^{0.75}(b)$	0.75	−0.25	−0.25	...	...
$P^{0.75}(c)$	0.75	−0.75	1	0.25	1
$P^{0.75}(d)$	0.75	0	0.75	0.25	1
$P^{0.75}(e)$	0.75	0	0.75	0.25	1
$P^{0.75}(f)$	0.75	0	1.5	0.25	1



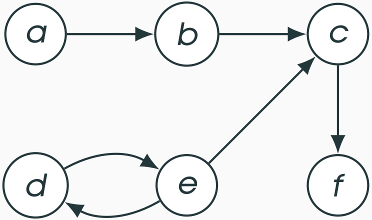
[4] E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

Definition.

$$P_0^{\epsilon}(a) = \begin{cases} 1 & a \text{ is unattacked} \\ \epsilon & a \text{ is not unattacked} \end{cases}$$

$$P_i^{\epsilon}(a) = P_{i-1}^{\epsilon}(a) + (-1)^i \sum_{b \in \downarrow^i(a)} P_0^{\epsilon}(b)$$

$P^{0.75}(a)$	1	1	...	...	...	...
$P^{0.75}(b)$	0.75	−0.25	−0.25	...	...	...
$P^{0.75}(c)$	0.75	−0.75	1	0.25	1	...
$P^{0.75}(d)$	0.75	0	0.75	0.25	1	...
$P^{0.75}(e)$	0.75	0	0.75	0.25	1	...
$P^{0.75}(f)$	0.75	0	1.5	0.25	1	...



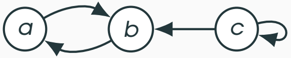
$$a > c \sim d \sim e \sim f > b$$

[4] E. Bonzon et al. "Argumentation Ranking Semantics based on Propagation". In: *Proceedings of the 6th International Conference on Computational Models of Argument*. IOS Press, Sept. 2016, pp. 139–150.

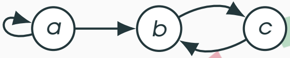
Comparison of Ranking Semantics

Amgoud and Ben-Naim (2013) propose these postulates:

1. **Abstraction**: Isomorphic argumentation frameworks must produce the same rankings.



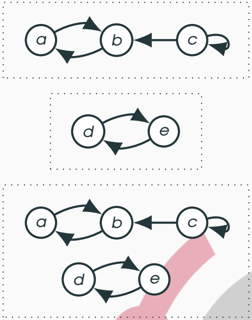
$a > b > c$



$c > b > a$

Amgoud and Ben-Naim (2013) propose these postulates:

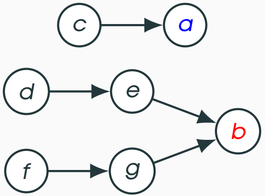
1. **Abstraction**: Isomorphic argumentation frameworks must produce the same rankings.
2. **Independence**: Each weakly connected component must be evaluated independently of the rest of the argumentation framework.



3. **Void Precedence**: Unattacked arguments must be ranked higher than attacked arguments.

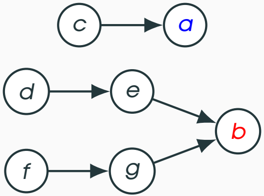
$$a > \ldots$$

3. **Void Precedence**: Unattacked arguments must be ranked higher than attacked arguments.
4. **Cardinality Precedence**: If *a* has fewer attackers than *b*, then *a* must be ranked higher than *b*.



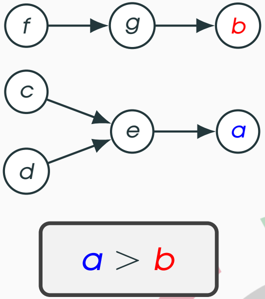
$a > b$

3. **Void Precedence**: Unattacked arguments must be ranked higher than attacked arguments.
4. **Cardinality Precedence**: If a has fewer attackers than b, then a must be ranked higher than b.
5. **Quality Precedence**: If at least one attacker of a is ranked above all attackers of b, then b must be ranked higher than a.

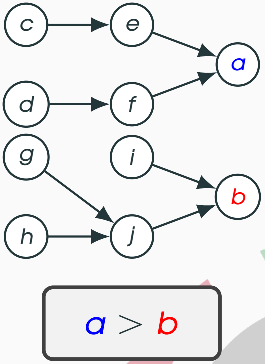


$b > a$

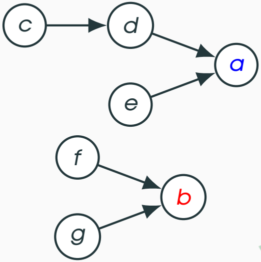
6. **Defense Precedence**: If *a* and *b* have the same number of attackers and *a* is defended by more arguments than *b*, then *a* must be ranked higher than *b*.



6. **Defense Precedence**: If *a* and *b* have the same number of attackers and *a* is defended by more arguments than *b*, then *a* must be ranked higher than *b*.
7. **Distributed-Defense Precedence**: For two arguments *a* and *b* with the same number of attackers and the same number of defenders, if *a*'s defense is simple and distributed and *b*'s defense is simple and NOT distributed, then *a* must be ranked higher than *b*.

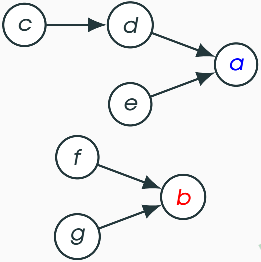


8. **Counter-Transitivity**: If the attackers of *b* are at least as many and ranked as high as the attackers of *a*, then *a* must not be ranked below *b*.



$a \geqslant b$

8. **Counter-Transitivity**: If the attackers of *b* are at least as many and ranked as high as the attackers of *a*, then *a* must not be ranked below *b*.
9. **Strict Counter-Transitivity**: If the attackers of *b* are at least as many and ranked as high as the attackers of *a*, and either strictly more or ranked strictly higher, then *a* must be ranked higher than *b*.



$a > b$

Outlook

Connection to Computational Social Choice?

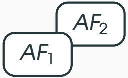
Connection to Computational Social Choice?

Aggregating Rankings of Alternatives!

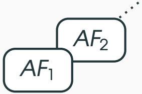
Outlook

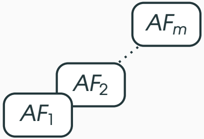
AF_1

Outlook



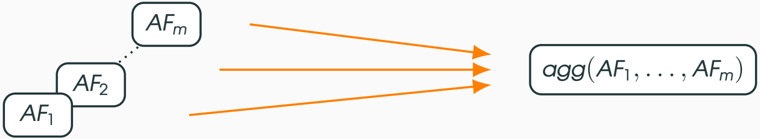
Outlook




AF_m
AF_2
AF_1

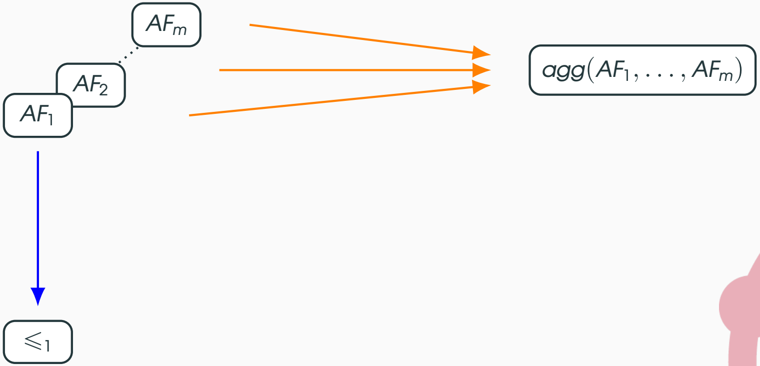
Outlook

- Use an AF aggregator to merge AFs into one

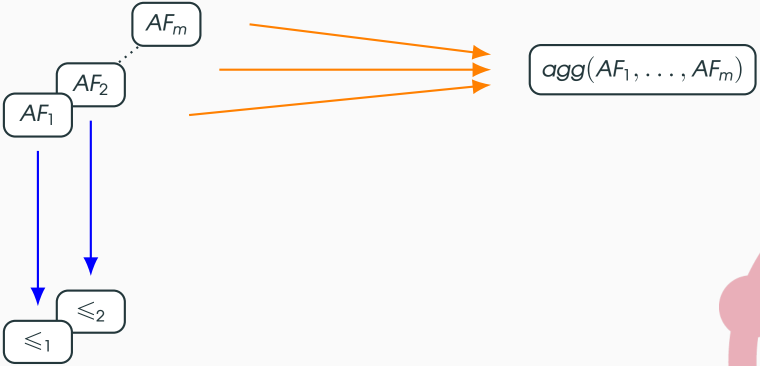


Outlook

- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs

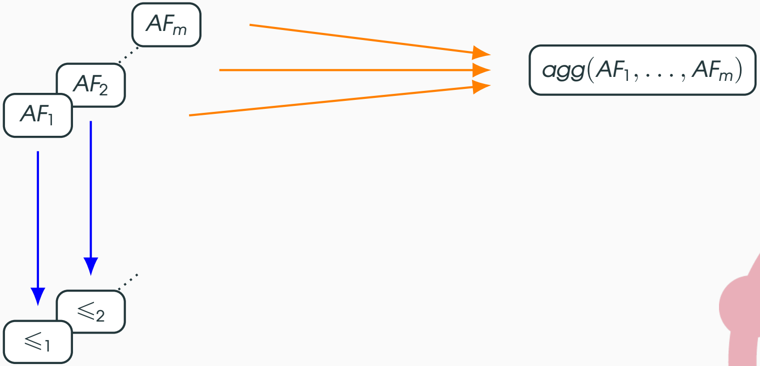


- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs

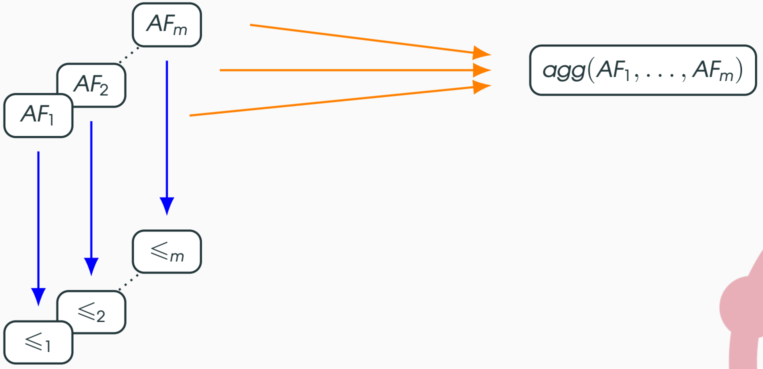


Outlook

- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs

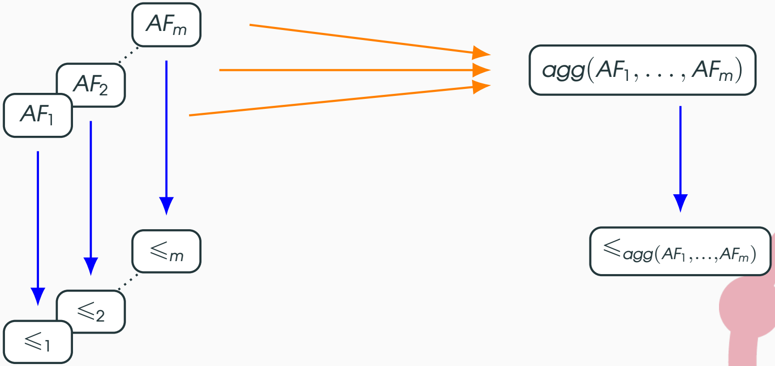


- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs



Outlook

- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs



Outlook

- Use an AF aggregator to merge AFs into one
- Use a ranking semantics to evaluate individual AFs
- Use a social welfare function to merge rankings into one