I don’t care!

On Incompleteness in Abstract Argumentation (and Belief Revision?)

Pietro Baroni
DII - Dip. di Ingegneria dell’Informazione
University of Brescia (Italy)
Based on joint work with
Massimiliano Giacomin and Beishui Liao

Motivations

- Abstract argumentation is focused on evaluating the acceptability of arguments on the basis of their conflicts
- Argumentation semantics can be regarded as a formal approach to answer, for each argument, the question: “Is this argument acceptable?”
- It is interesting to analyze which answers are available beyond “Yes” or “No”

Goals

- Analyzing the answer “I don’t care” (i.e. the tretament of incompleteness) in abstract argumentation literature (with some attention to non-mainstream approaches)

- Pointing out further research directions and connections with other areas

Outline

- Abstract Argumentation (AA)
- Incompleteness in AA: a don't care label
- Incompleteness in AA: partial semantics and decomposability
- Perspectives and conclusions

Abstract argumentation

- Dung's framework …

Dung's framework is (almost) nothing

Definition 2. An *argumentation framework* is a pair

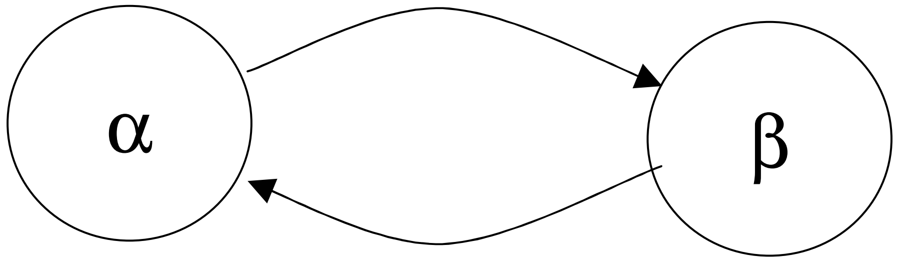
$$AF = \langle AR, attacks \rangle$$

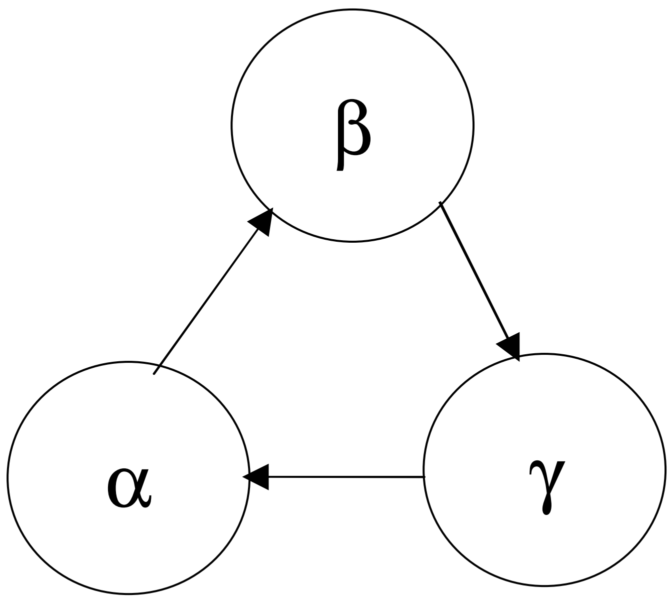
where AR is a set of arguments, and *attacks* is a binary relation on AR, i.e. $attacks \subseteq AR \times AR$.

- A directed graph (called *defeat graph*) where:
  - » arcs are interpreted as attacks
  - » nodes are called arguments "by chance" (let say historical reasons)

Here, an argument is an abstract entity whose role is solely determined by its relations to other arguments. No special attention is paid to the internal structure of the arguments.

Dung's framework is (almost) nothing





Dung's framework is (almost) everything

- Arguments are simply "conflictables"

- Conflicts are everywhere
- Conflict management is a fundamental need with potential spectacular/miserable failures both in real life and in formal contexts (e.g. in classical logic)
- A general abstract framework centered on conflicts has a wide range of potential applications

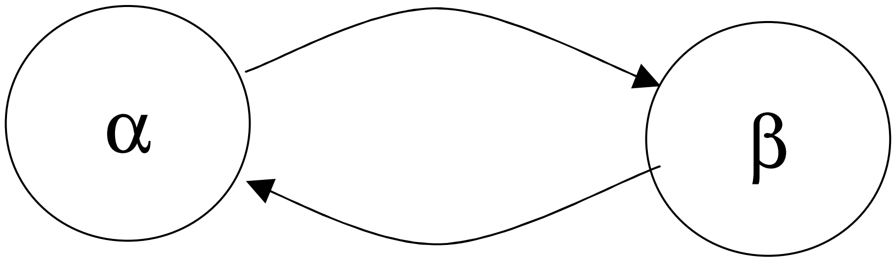
A conflict calculus: abstract argumentation semantics

- A way to identify sets of arguments “surviving the conflict together” given the conflict relation only
- Two main styles for semantics definition: extension-based and labelling-based
- In general, several choices of sets of “surviving arguments” are possible (multiple-status semantics) but some semantics prescribe exactly one extension/labelling (single status semantics)

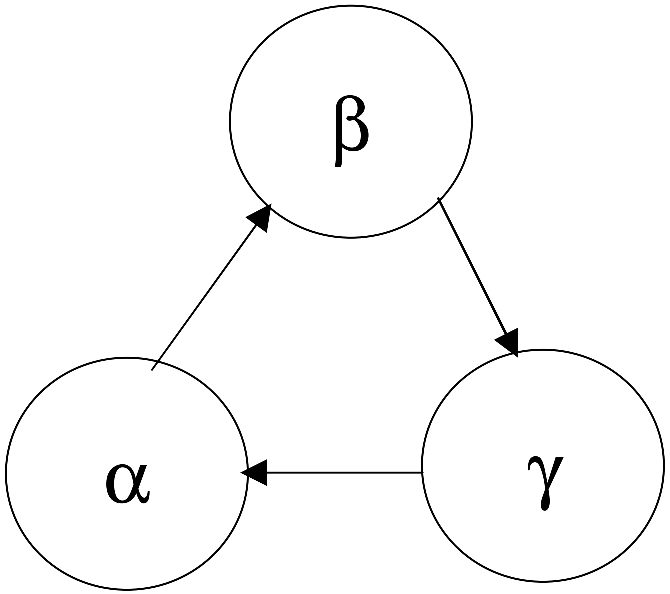
Extension-based semantics

- A set of extensions is identified
- Each extension is a set of arguments which can “survive together” or are “collectively acceptable” i.e. represent a reasonable viewpoint
- The justification status of each argument can be defined on the basis of its extension membership
  - » skeptical justification = membership in all extensions
  - » credulous justification = membership in one extension

Sets of extensions



$$E_1 = \{\{\alpha\},\{\beta\}\}$$
$$E_2 = \{\varnothing\}$$

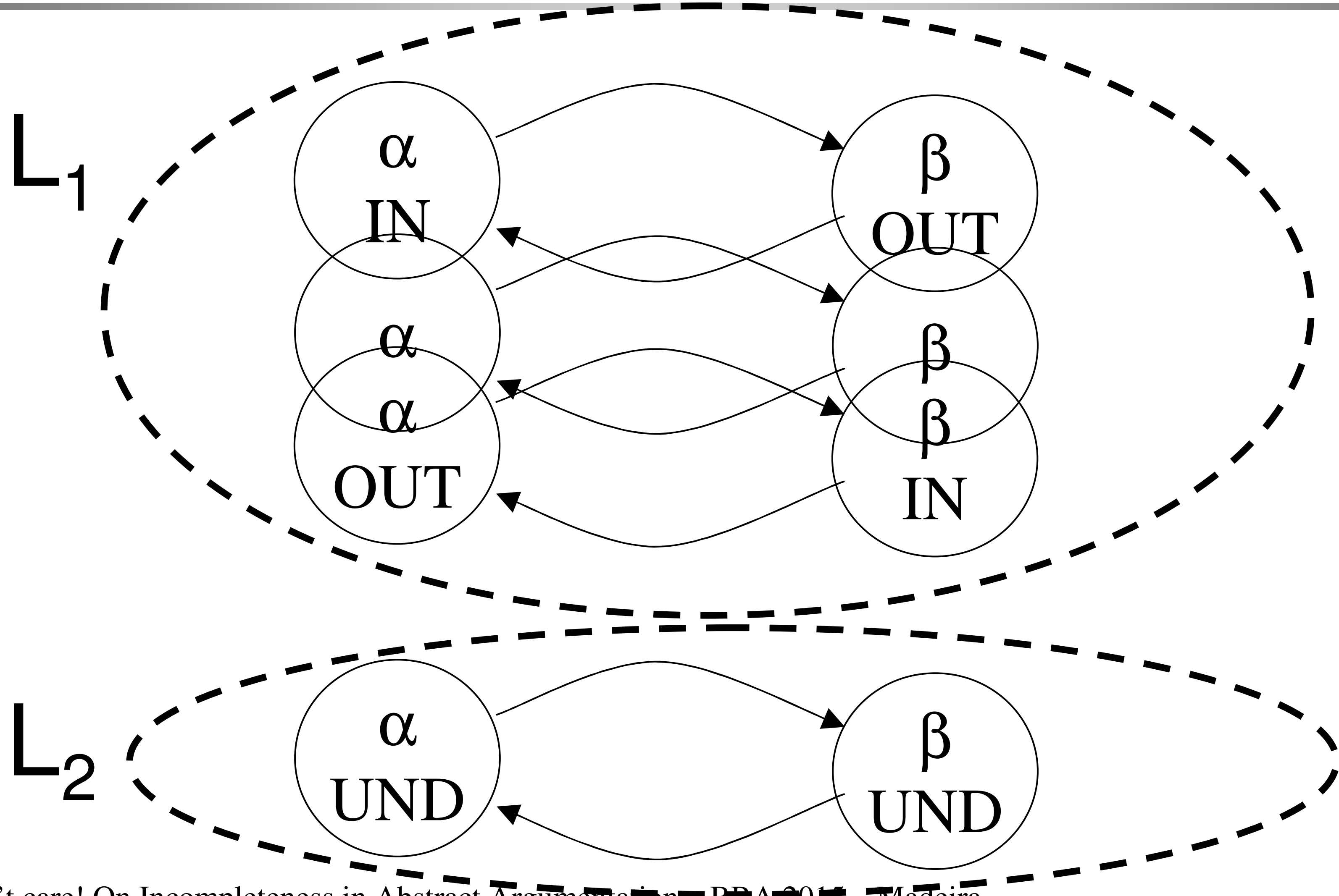


$$E_1 = \{\{\alpha\},\{\beta\},\{\gamma\}\}$$
$$E_2 = \{\varnothing\}$$

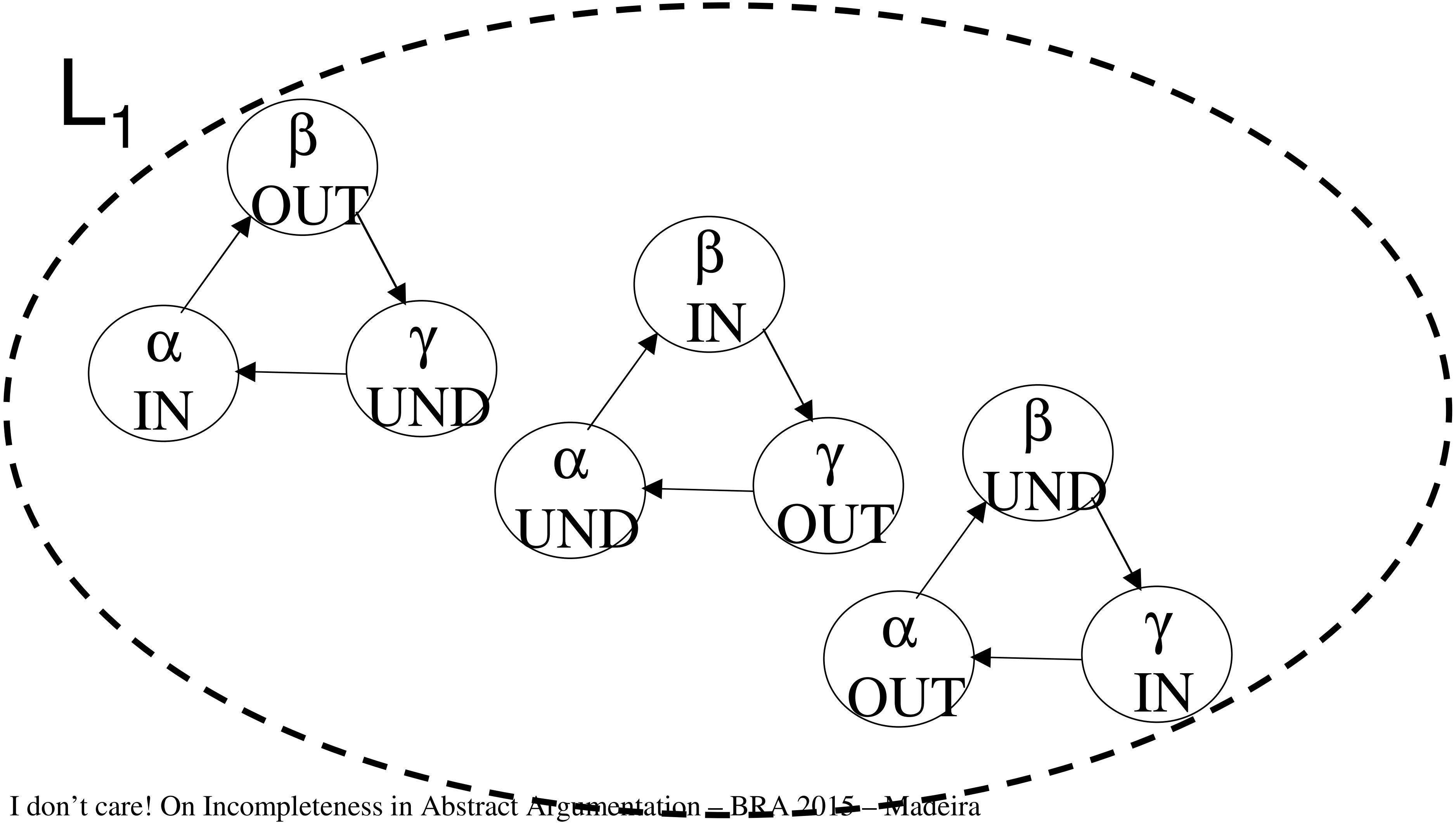
Labelling-based semantics

- A set of labels is defined (e.g. IN, OUT, UNDECIDED) and criteria for assigning labels to arguments are given
- Several alternative labellings are possible
- The justification status of each argument can be defined on the basis of its labels

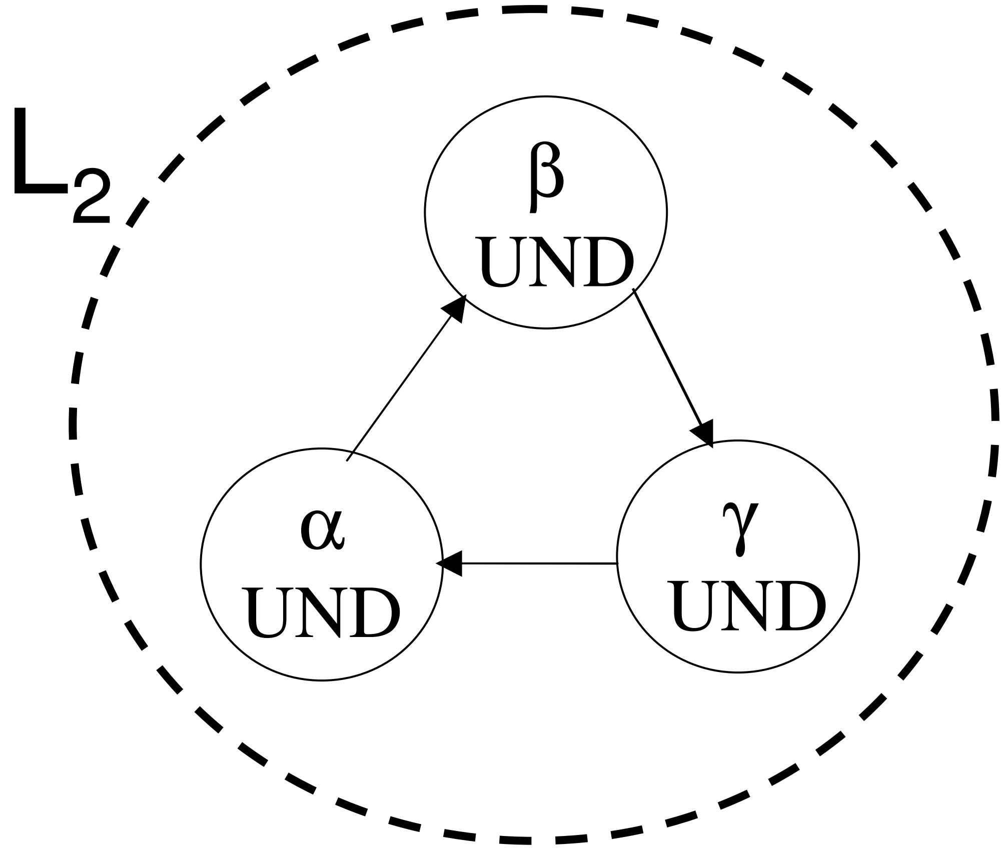
Labelling-based semantics



Labelling-based semantics



Labelling-based semantics



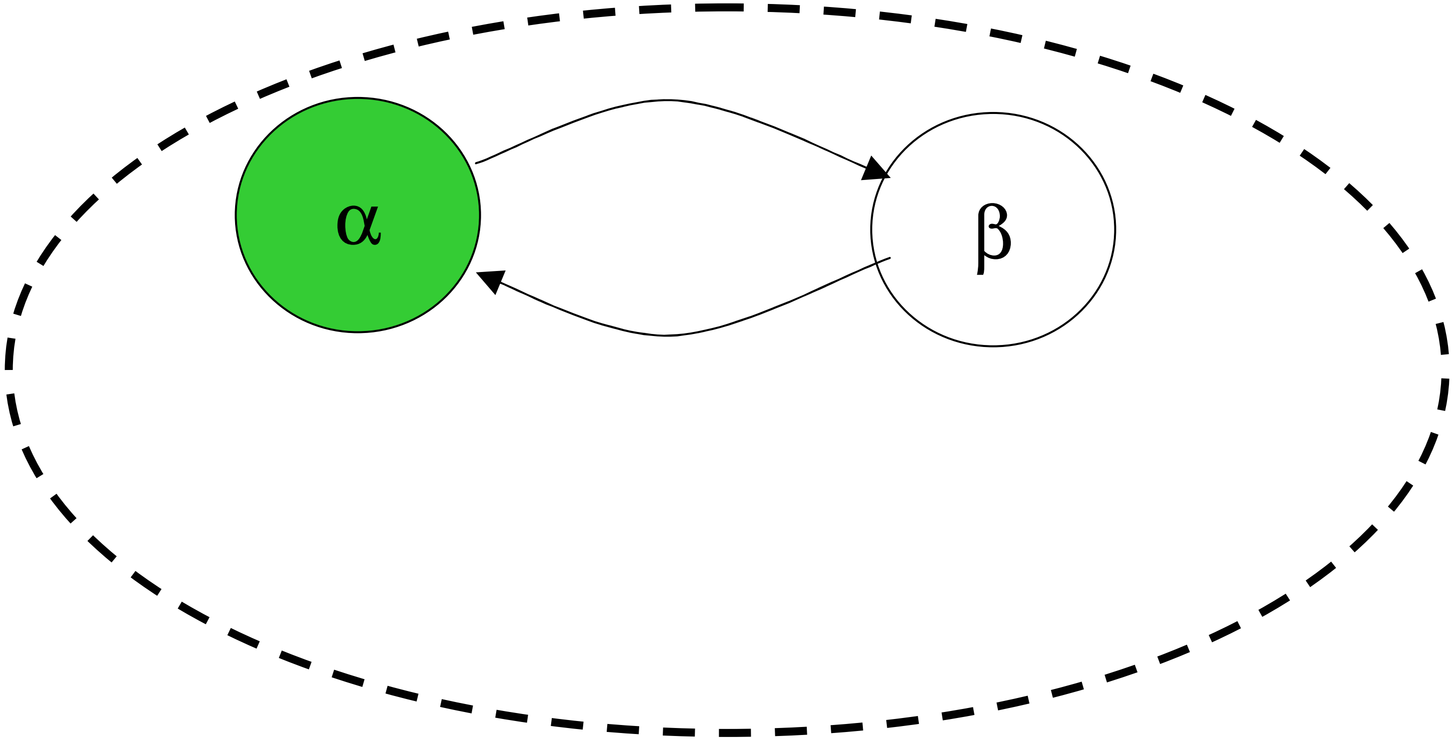
Labellings vs. extensions

- Labellings based on {IN, OUT, UNDEC} and extensions can be put in direct correspondence

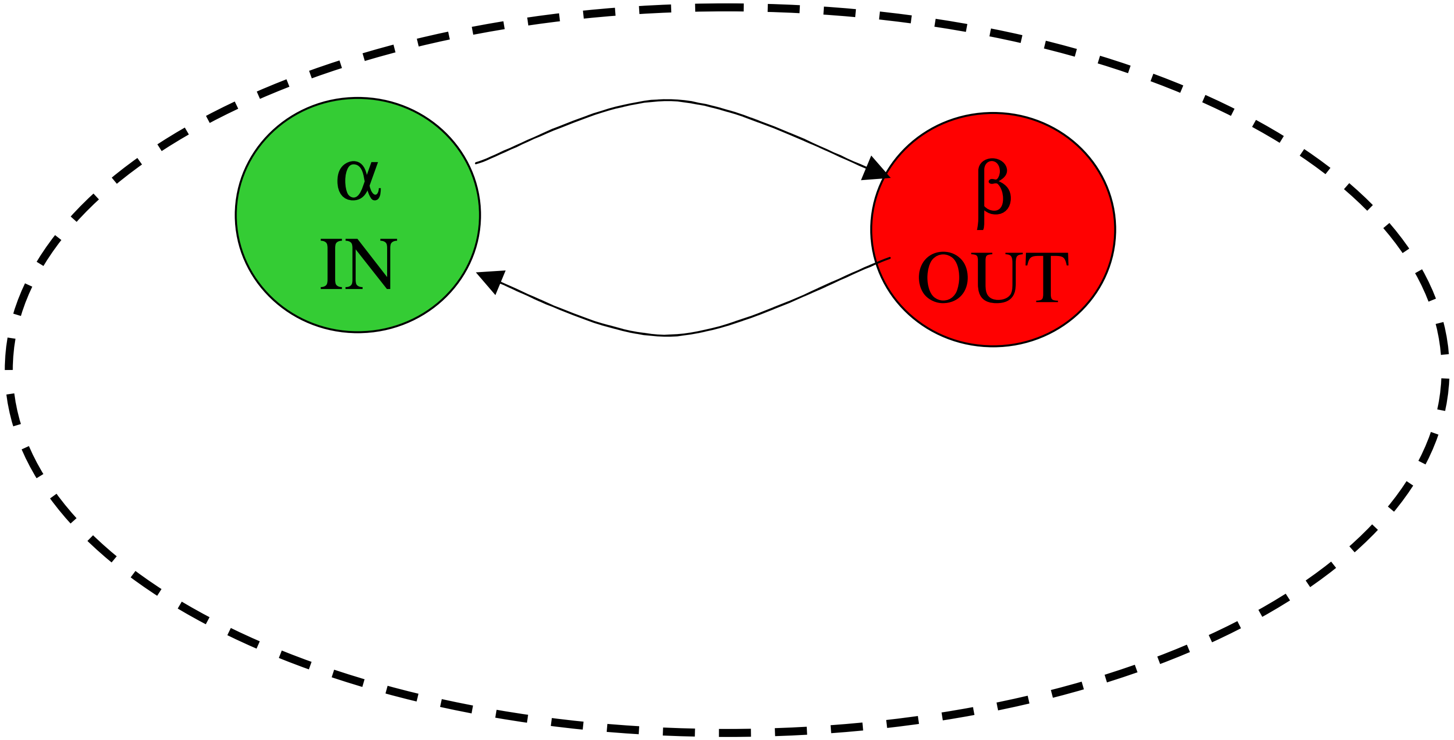
- Given a labelling L, LabToExt(L) = in(L)

- Given an extension E, a labelling L=ExtToLab(E) can be defined as follows:
  in(L)=E
  out(L)=attacked(E)
  undec(L)=all other arguments

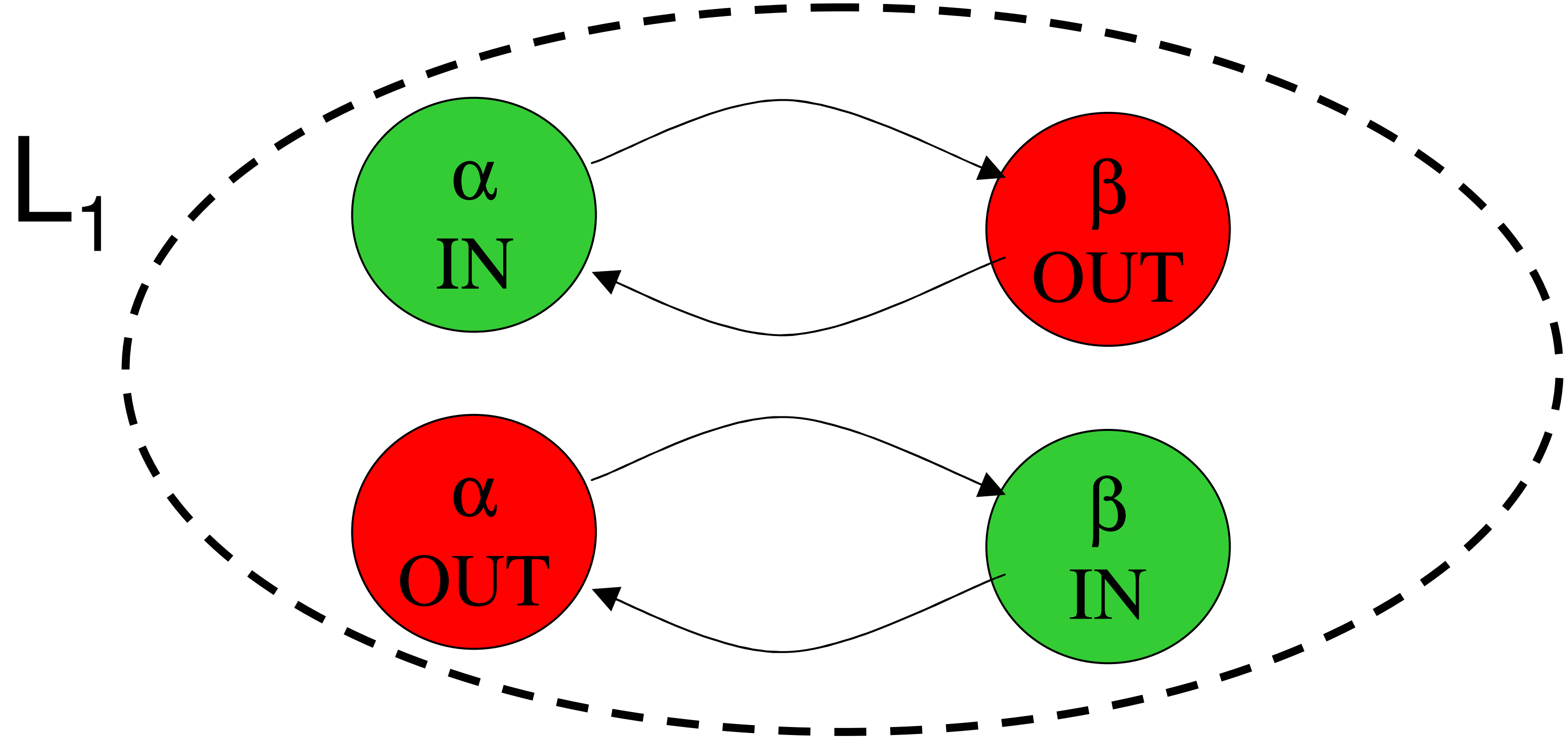
Labellings vs. extensions



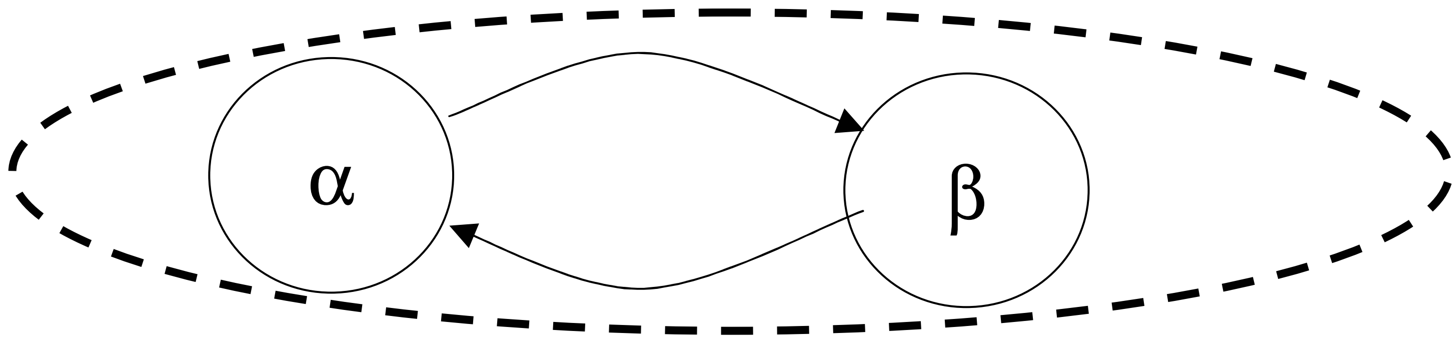
Labellings vs. extensions



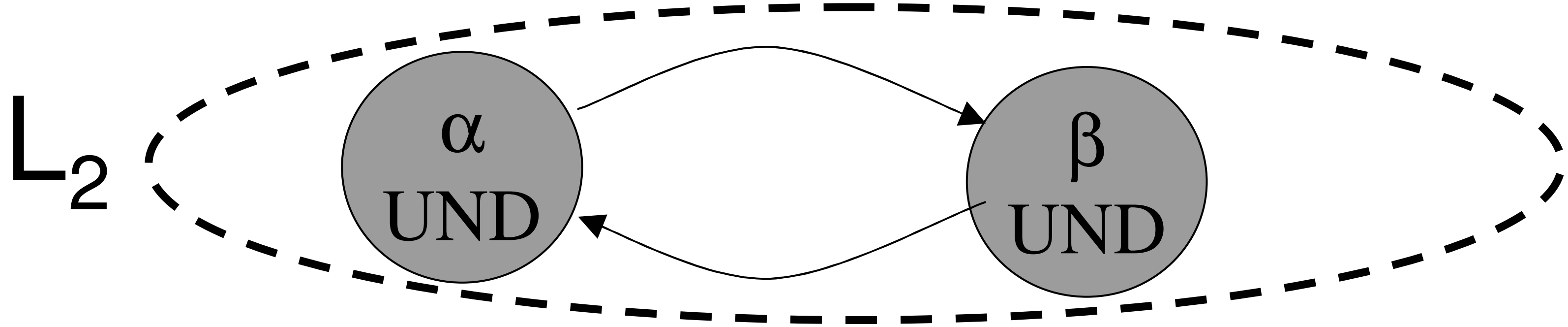
Labellings vs. extensions



Labellings vs. extensions



Labellings vs. extensions



Basic legality constraints on labels

- An argument is IN iff all its attackers are OUT
- An argument is OUT iff it has an attacker IN
- An argument is UND iff it has an attacker UND and no attackers IN

Outline

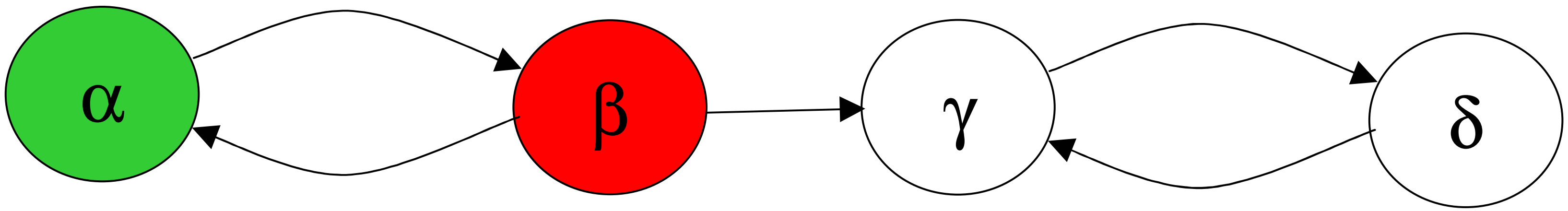
- Abstract Argumentation (AA)
- **Incompleteness in AA: a don't care label**

I don’t care! Allowing incompleteness

- One may want to evaluate the acceptance of some arguments only, leaving the others unspecified
- It’s like having the option “no color” (or a fourth special color) in the labelling approach

I don't care! Allowing incompleteness

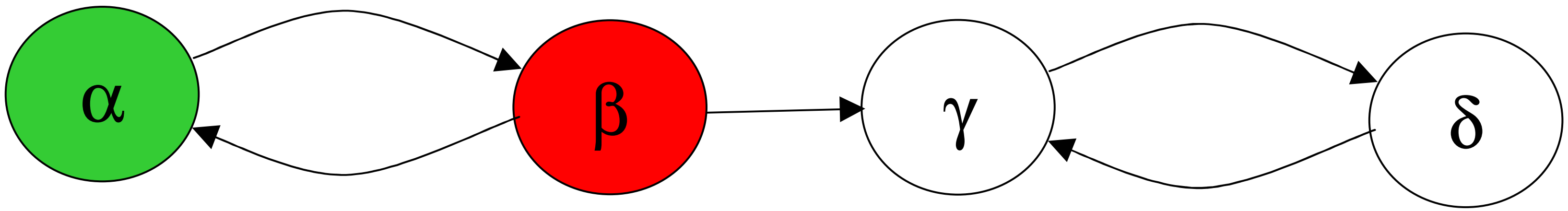
- One may want to evaluate the acceptance of some arguments only, leaving the others unspecified
- It's like having the option "no color" (or a fourth special color) in the labelling approach



I don't care about γ and δ

I don't care! Allowing incompleteness

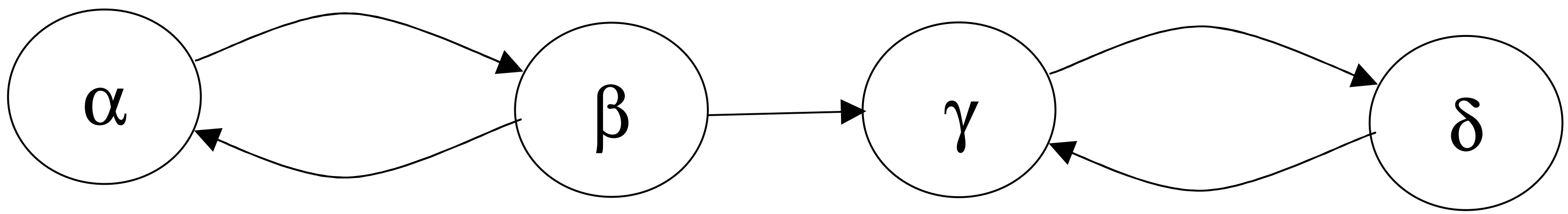
- One may want to evaluate the acceptance of some arguments only, leaving the others unspecified
- It's like having the option "no color" (or a fourth special color) in the labelling approach



These arguments intentionally left blank

I don't care! Allowing incompleteness

- One may want to evaluate the acceptance of some arguments only, leaving the others unspecified
- It's like having the option "no color" (or a fourth special color) in the labelling approach



I don't care at all!

Why not to color all?

- To save paint (i.e. computational resources): you don’t spend resources for evaluations you are not going to use (uninteresting, redundant, ephemeral)

- To save reputation (minimal commitment): you cautiously avoid to take a position when it is not strictly required (and maybe could change very soon)

The "don't care" label of JV99

- Jakobovits and Vermeir proposed in 1999 a set of four labels: +, -, ±, ø.
- +, -, ± correspond to IN, OUT, UND,
  ø means "don't care"
- A labeling including some ø is called partial
- The ø label is reserved to "arguments that are irrelevant or that do not interest the observer"
- This suggests discretionality in its assignment but…

JV legality constraints

$$\text{if } L(\alpha) \in \{-, \pm\} \text{ then } \exists \beta \in \{\alpha\}^{\leftarrow} \text{ such that } L(\beta) \in \{+, \pm\};$$

The presence of a minus must be justified by the presence of a plus in some attacker

$$\text{if } L(\alpha) \in \{+, \pm\} \text{ then } \forall \beta \in \{\alpha\}^{\leftarrow}\ L(\beta) \in \{-, \pm\};$$

The presence of a plus must be justified by the presence of a minus in all attackers

$$\text{if } L(\alpha) \in \{+, \pm\} \text{ then } \forall \beta \in \{\alpha\}^{\rightarrow}\ L(\beta) \in \{-, \pm\}$$

The presence of a plus causes the presence of a minus in all attackees

Implied legality constraints on ø

- The ø label is only possible for an argument α if all the following conditions hold:

$$\forall \beta \in \{\alpha\}^{\leftarrow}\ L(\beta) \in \{\otimes, -\}$$

No attacker has a plus

$$\nexists \beta \in \{\alpha\}^{\rightarrow} \text{ such that } L(\beta) \in \{+, \pm\};$$

No attackee has a plus

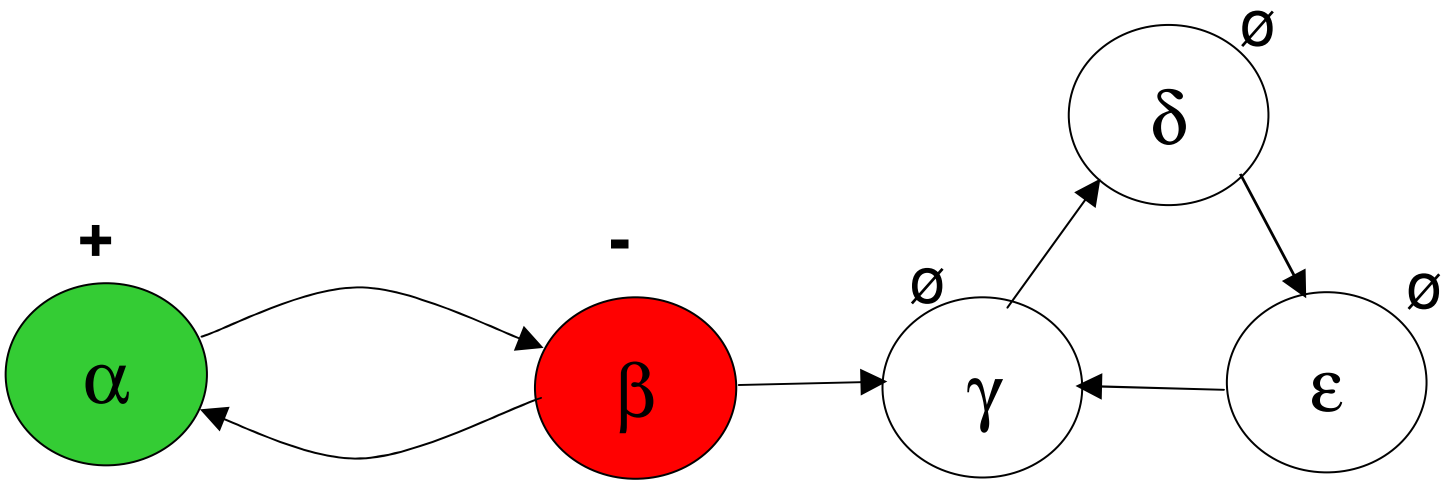
$$\forall \beta \in \{\alpha\}^{\rightarrow} \text{ if } L(\beta) = - \text{ then } \exists \gamma \in \{\beta\}^{\leftarrow} \setminus \{\alpha\} \text{ such that } L(\gamma) \in \{+, \pm\}$$

The attackees labelled - are justified by some other argument

Careless constraints

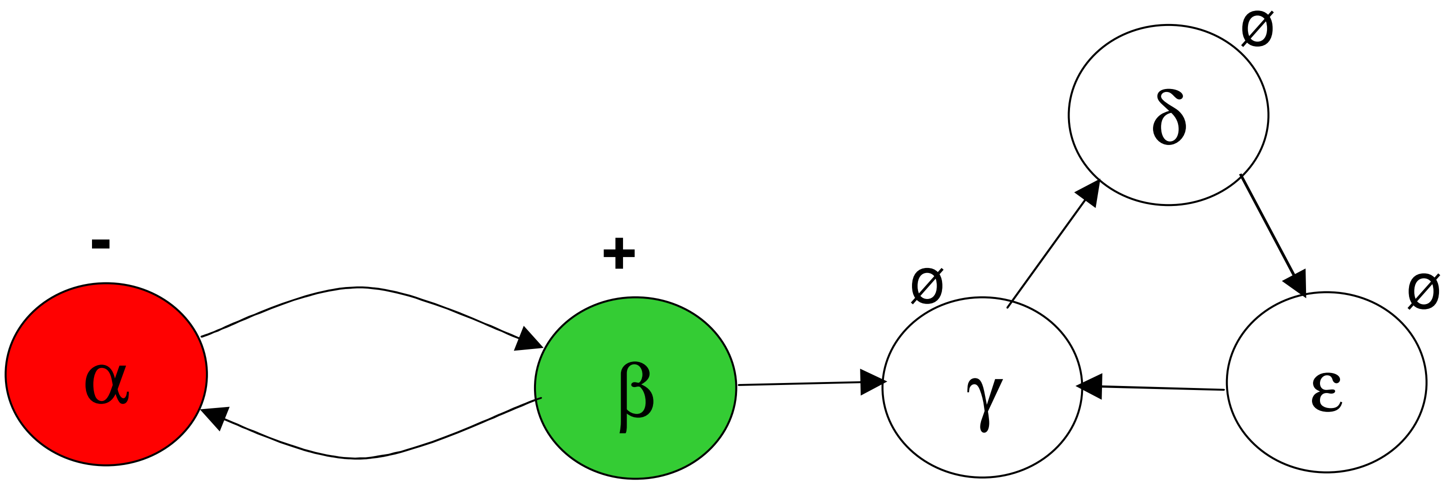
- Full carelessness is always legal
- Partial carelessness may not
- Intuitive as they seem, these constraints are asymmetric:
  - » one may label ø an argument otherwise labelled +
  - » one may not label ø an argument otherwise labelled - or ±
- In some cases, one may assign the ø label to an argument whose label is anyway uniquely determined by the other ones
- Carelessness is not undecidedness

Careless constraints



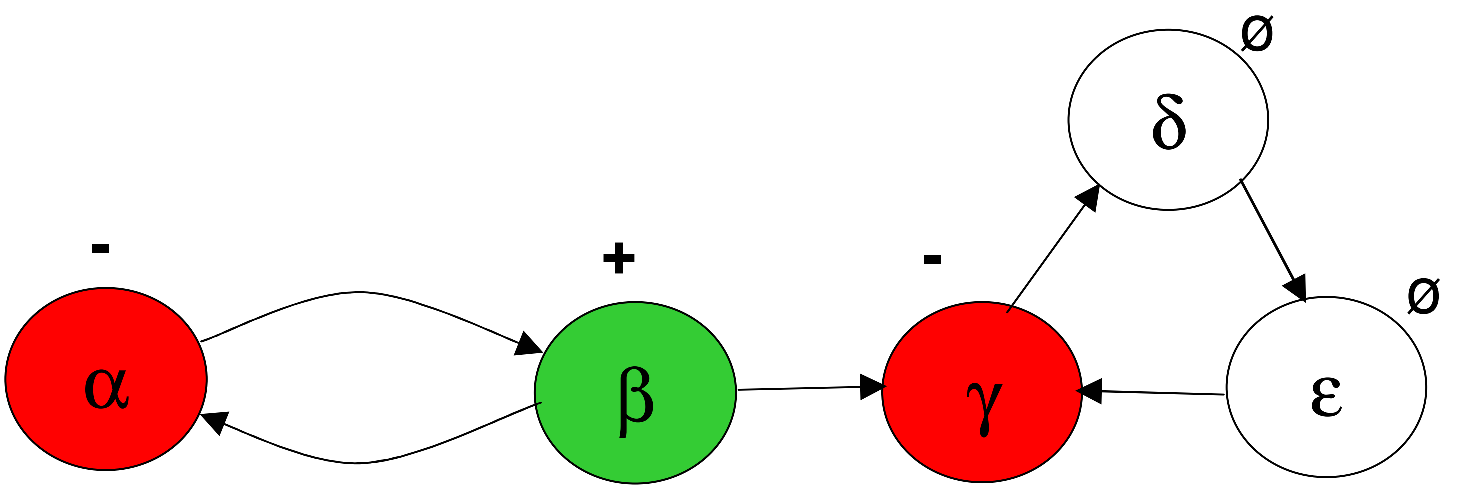
Legal labeling (according to JV99) with 3 "don't care"
Note that there is only one possible label for γ,δ,ε

Careless constraints



Illegal labeling (according to JV99) with 3 "don't care":
γ must be -

Careless constraints



Legal labeling (according to JV99) with 2 don't care
Note that there is only one possible label for δ,ε

Outline

- Abstract Argumentation (AA)
- Incompleteness in AA: a don’t care label
- **Incompleteness in AA: partial semantics and decomposability**

Partial argumentation semantics

- JV99 uses an explicit label for “don’t care”: the labelling still covers all arguments, but some labels express partiality
- One may instead restrict the semantics definition to a strict subset of the arguments, excluding some of them from the evaluation
- Some arguments are ignored “by definition” rather than being explicitly labelled as “uninteresting”

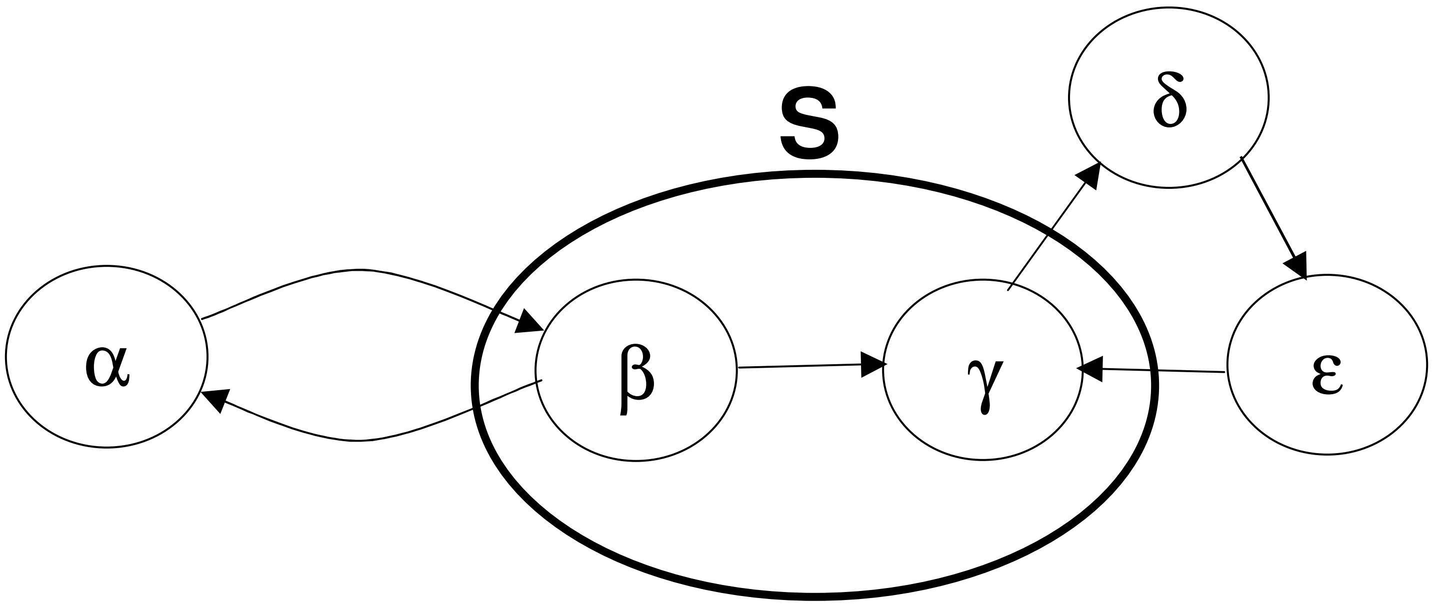
Partial argumentation semantics

- Two interplaying ingredients:

1. a way to define the scope of the partial evaluation: suitable restriction of the framework
2. properties ensuring coherence between partial and global semantics

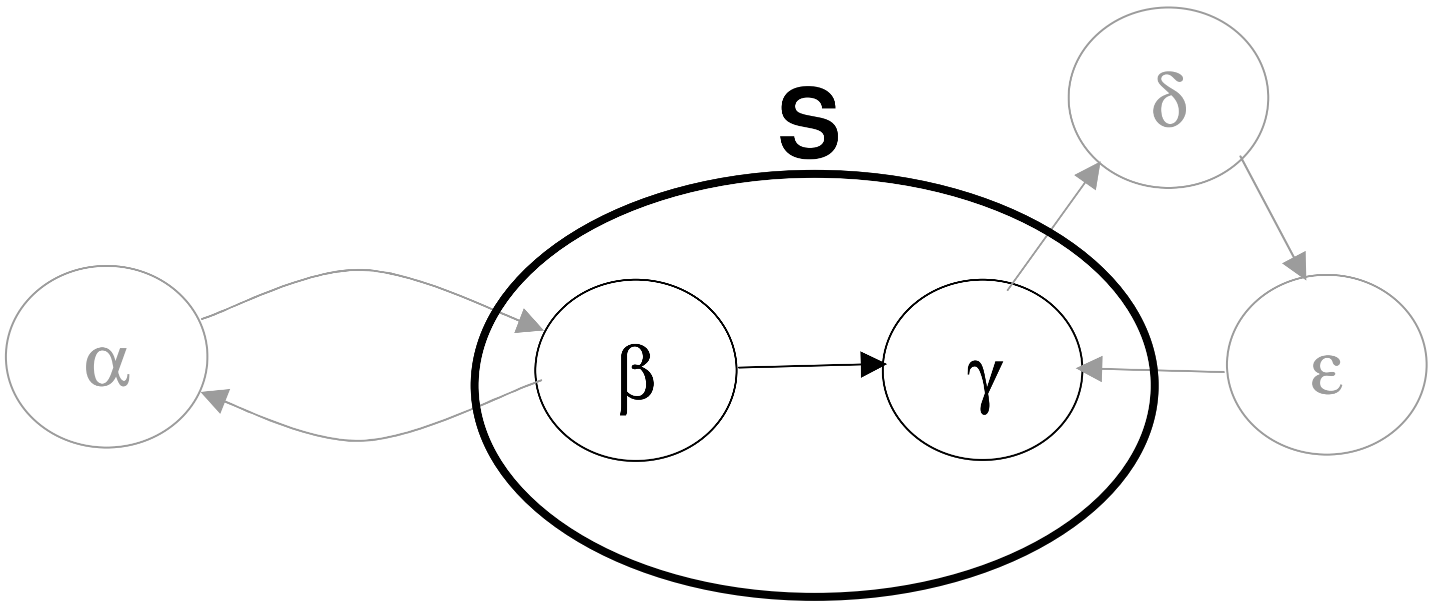
Plain cuts + directionality

- Given a set of arguments S the simplest way to cut is to ignore all the rest: $\mathcal{F}\downarrow_S = \langle S, \rightarrow \cap (S \times S)\rangle$



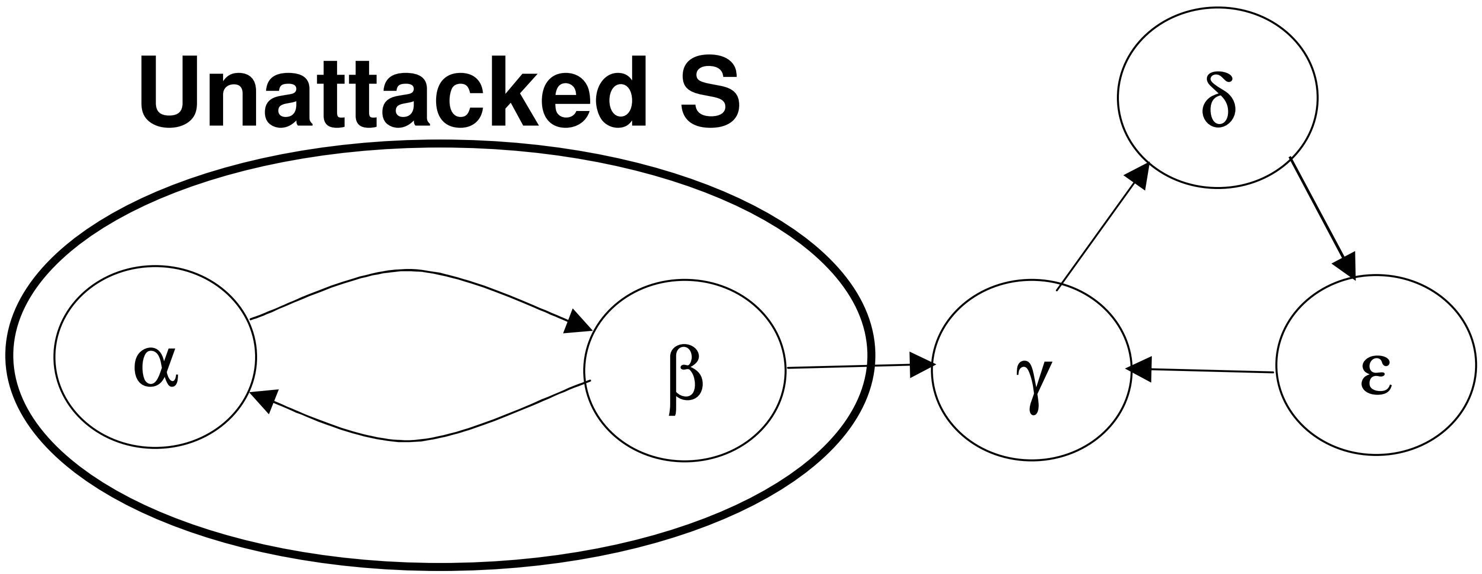
Plain cuts + directionality

- Given a set of arguments S the simplest way to cut is to ignore all the rest: $\mathcal{F}\downarrow_S = \langle S, \rightarrow \cap (S \times S) \rangle$



Plain cuts + directionality

- The plain cut strategy becomes more reasonable if the set S is unattacked and the considered semantics is directional



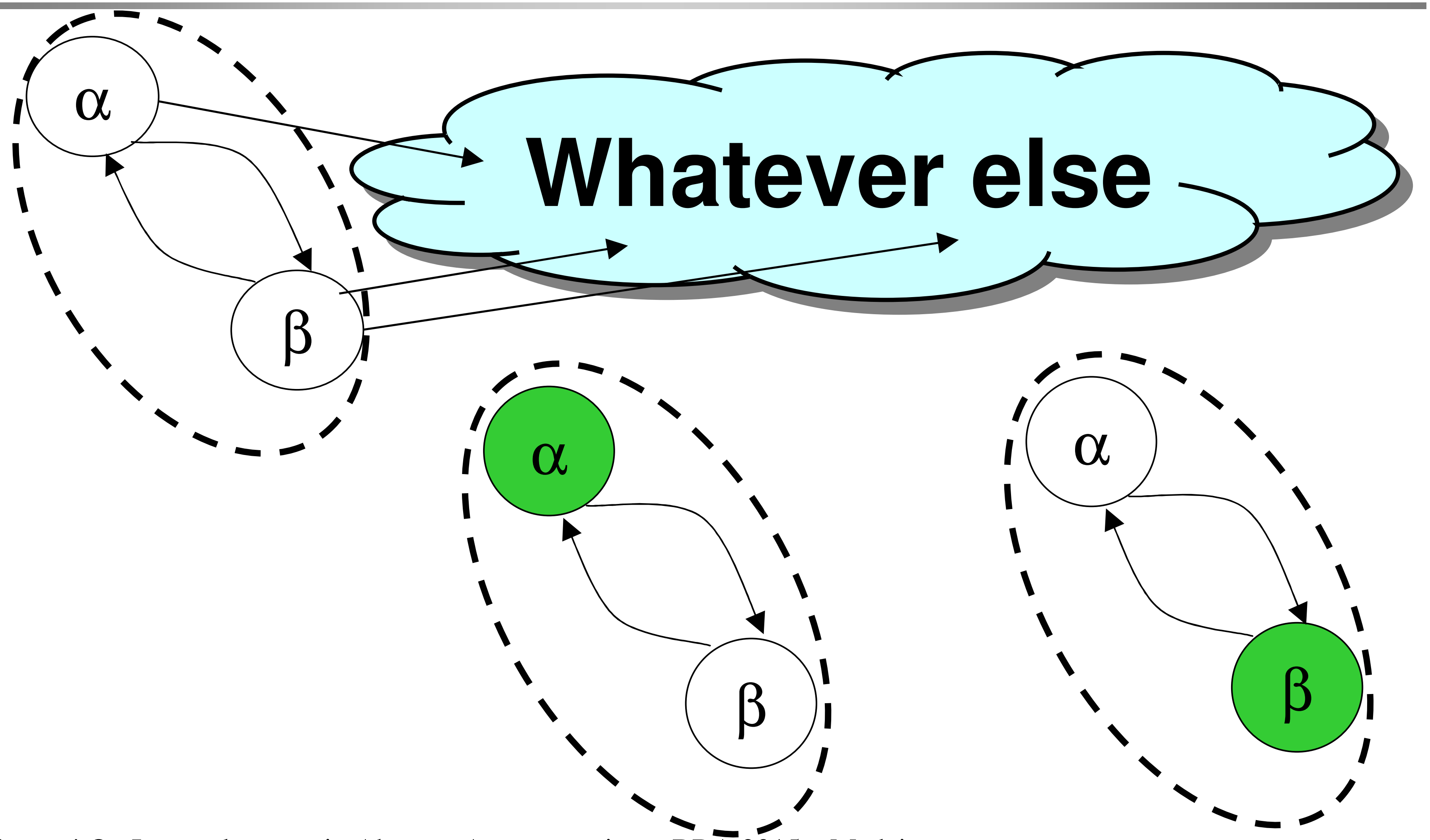
Directionality

Definition 18. Given an argumentation framework $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$, a set $U \subseteq \mathcal{A}$ is *unattacked* if and only if $\nexists \alpha \in (\mathcal{A} \setminus U)\colon \alpha \rightarrow U$. The set of unattacked sets of AF is denoted as $\mathcal{US}(\mathrm{AF})$.

Definition 19. A semantics $\mathcal{S}$ satisfies the directionality criterion if and only if $\forall\, \mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$, $\forall U \in \mathcal{US}(\mathrm{AF})$, $\mathcal{AE}_{\mathcal{S}}(\mathrm{AF}, U) = \mathcal{E}_{\mathcal{S}}(\mathrm{AF}{\downarrow_U})$, where $\mathcal{AE}_{\mathcal{S}}(\mathrm{AF}, U) \triangleq \{(E \cap U) \mid E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})\} \subseteq 2^U$.

- The intersections of the extensions/labelings with an unattacked part of the AF are the same whatever is the remaining part of the AF and coincide with the extensions/labelings of the restricted AF

Directionality



Directionality and partial semantics

- Directionality corresponds to an “indifference to change” of a part of the AF (an unattacked set)
- The underlying intuition is closer to suppression (of the rest of the AF) rather than to expansion
- Potentially very useful for partial semantics and local computation: we can totally ignore part of the AF if what we need is within an unattacked set …
- and we have the guarantee that what we compute locally will be preserved at the global level

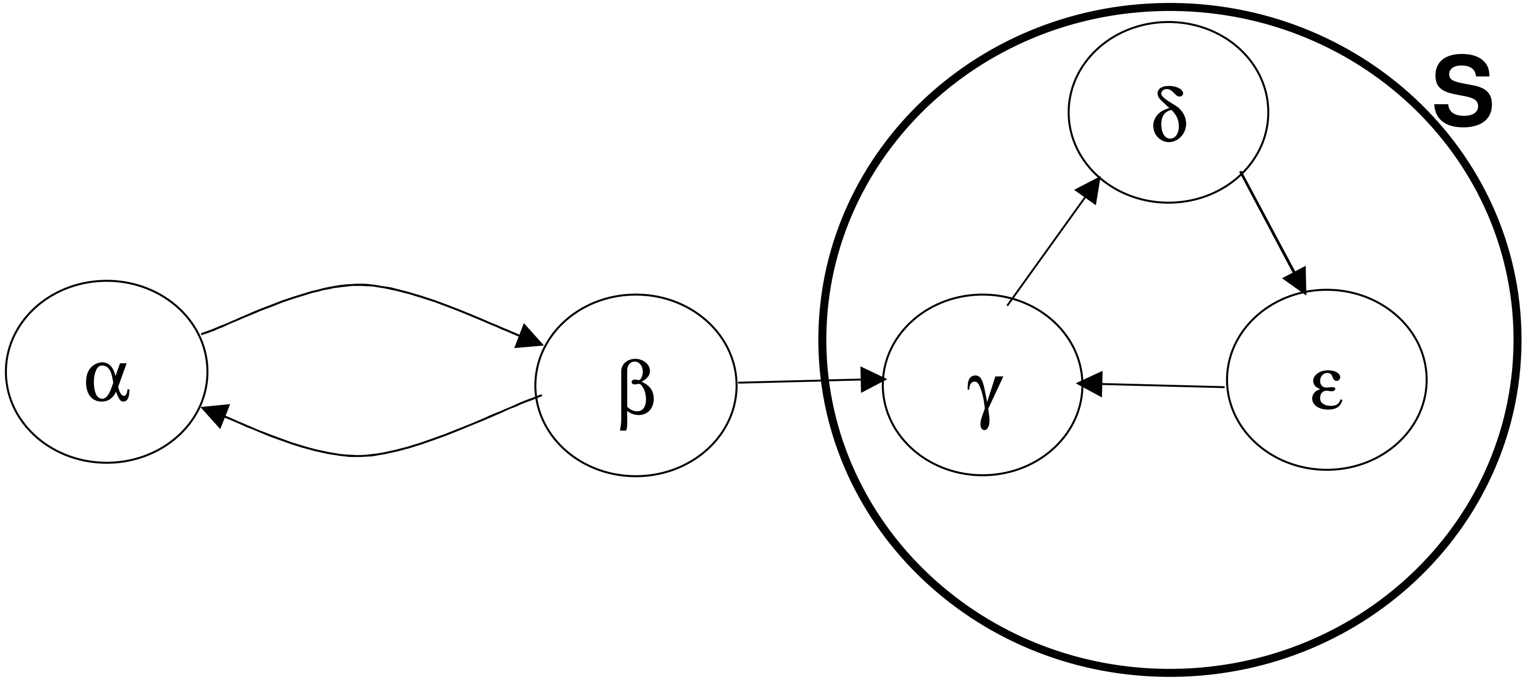
Directionality and partial semantics

Definition 10. *Given an AF* $\mathcal{F} = \langle \mathcal{A}, \rightarrow \rangle$ *and a set of arguments* $S \subseteq \mathcal{A}$*, define* $rlvt_{\mathcal{F}}(S) = \min_{\subseteq}\{U \mid S \subseteq U \wedge U \in \mathcal{US}(\mathcal{F})\}$*. Given an extension-based (labelling-based) semantics* σ *satisfying the directionality criterion the partial semantics of* $\mathcal{F}$ *with respect to* S *is defined as* $\mathcal{E}_\sigma(\mathcal{F}\downarrow_{rlvt_{\mathcal{F}}(S)})$ $(\mathcal{L}_\sigma(\mathcal{F}\downarrow_{rlvt_{\mathcal{F}}(S)}))$*.*

- Apply the usual semantics on the (unattacked) restriction and ignore the rest
- If the rest changes, the partial results remain the same: very useful for argumentation dynamics and incremental computation
- Idea used in several works: splitting AFs, division-based method.

Partial conditional semantics: adding a fixed influence from outside

- The set S is not unattacked, but receives a fixed influence from outside (evaluation within S does not affect backwards the received influence)



Conditioned AF in the division-based method

- The notion of conditioned framework formalizes a situation where a (conditioned) framework is evaluated subject to a prior evaluation of another framework through some conditioning arguments

Definition 11. Given an argumentation framework $AF_1 = \langle A_1, R_1 \rangle$, a conditioned argumentation framework w.r.t. AF_1 is a tuple $CAF = (\langle A_2, R_2 \rangle, (C(A_1), \mathcal{I}_{(C(A_1),A_2)}))$, in which

- $\langle A_2, R_2 \rangle$ is an argumentation framework that is conditioned by $C(A_1)$, in which $A_2 \cap A_1 = \emptyset$;
- $C(A_1) \subseteq A_1$ is a nonempty set of arguments (called conditioning arguments) that have interactions with arguments in A_2, i.e., $\forall \alpha \in C(A_1), \exists \beta \in A_2$, s.t. $(\alpha, \beta) \in \mathcal{I}_{(C(A_1),A_2)}$; and
- $\mathcal{I}_{(C(A_1),A_2)} \subseteq C(A_1) \times A_2$ is the set of interactions from the arguments in $C(A_1)$ to the arguments in A_2, and of the form (α, β), in which $\alpha \in C(A_1)$ and $\beta \in A_2$.

Conditional semantics

- Semantics can be defined for the conditioned part by adapting the usual definitions to take into account the effect coming from outside
- Crucial for dynamics/incremental computation
- Directionality of semantics is still necessary but no more sufficient
- Some additional property ensuring that the construction can proceed is required
- Works well with SCC-recursiveness, but full theoretical analysis of necessary/sufficient properties still to be developed

Incremental computation with the division-based method

- The division-based method works directly with complete, grounded, preferred semantics
- Incremental computation can also be applied to stable and ideal semantics with some adjustment to the basic division-based method

Partial semantics with arbitrary partitions

- The next step is to consider arbitrary partitions of an AF
- An arbitrary partition of an AF induces a set of interacting subframeworks, where each subframework may:
  - » receive some attacks from some external (belonging to another subframework) arguments
  - » launch some attacks against some external arguments

Generic partial evaluation

- Can we carry out partial evaluations in these arbitrary subframeworks?
- We can define a local function which takes into account the input coming from outside and computes the labellings/extensions inside
- The input coming from outside is represented by a labelling of the external attacking arguments

AF with input

Definition 11. *An* argumentation framework with input *is a tuple* (AF, I, L_I, R_I), *including an argumentation framework* $AF = (Ar, att)$, *a set of arguments* I *such that* $I \cap Ar = \emptyset$, *a labelling* $L_I \in \mathfrak{L}_I$ *and a relation* $R_I \subseteq I \times Ar$. *A* local function *assigns to any argumentation framework with input a set of labellings of* AF, *i.e.* $F(AF, I, L_I, R_I) \in 2^{\mathfrak{L}(AF)}$.

- The idea is similar to the conditioned AF, considering only conditioning arguments and a given labelling for them
- The local function corresponds to a local and “context aware” notion of semantics
- The local function can be easily recovered for semantics satisfying very mild properties

Semantics decomposability

- With a local function at hand we can wonder whether, for a given semantics, global labellings can be obtained from local labellings (and vice versa)
- Since the different subframeworks interact, local labellings can be combined together only if they are "compatible", i.e. for each local labelling L_i the input used by the local function to produce L_i is equal to the labels of the input arguments determined in other subframeworks taking into account L_i

Full decomposability

Definition 15. *A semantics* **S** *is* fully decomposable *(or simply* decomposable*) iff there is a local function F such that for any argumentation framework $AF = (Ar, att)$ and any partition $\{P_1, \ldots P_n\}$ of Ar,* $\mathbf{L}_{\mathbf{S}}(AF) = \{L_{P_1} \cup \ldots \cup L_{P_n} \mid L_{P_i} \in F(AF{\downarrow_{P_i}}, P_i^{in}, (\bigcup_{j=1\cdots n, j\neq i} L_{P_j}){\downarrow_{P_i^{in}}}, R_{P_i})\}$

- For **any partition** of **any AF**
- Any combination of compatible local labellings gives rise to a global labelling
- Any global labelling gives rise to a set of compatible local labellings

Top-down decomposability

Definition 16. *A complete-compatible semantics* **S** *is* top-down decomposable *iff for any argumentation framework* $AF = (Ar, att)$ *and any partition* $\{P_1, \ldots P_n\}$ *of* Ar*, it holds that* $\mathbf{L_S}(AF) \subseteq \{L_{P_1} \cup \ldots \cup L_{P_n} \mid L_{P_i} \in F_{\mathbf{S}}(AF{\downarrow_{P_i}}, {P_i}^{in}, (\bigcup_{j=1 \cdots n, j \neq i} L_{P_j}){\downarrow_{{P_i}^{in}}}, R_{P_i})\}$.

- Completeness of the combination procedure
- From the combinations of compatible local labellings you get all global labelings (and possibly something more)

Bottom-up decomposability

Definition 17. *A complete-compatible semantics* **S** *is* bottom-up decomposable *iff for any argumentation framework* $AF = (Ar, att)$ *and any partition* $\{P_1, \ldots P_n\}$ *of* Ar*, it holds that* $\mathbf{L_S}(AF) \supseteq \{L_{P_1} \cup \ldots \cup L_{P_n} \mid L_{P_i} \in F_{\mathbf{S}}(AF{\downarrow_{P_i}}, {P_i}^{in}, (\bigcup_{j=1 \cdots n, j \neq i} L_{P_j}){\downarrow_{{P_i}^{in}}}, R_{P_i})\}$.

- Soundness of the combination procedure
- All combinations of compatible local labellings give rise to global labelings (possibly not to all of them)

Semantics decomposability

- Admissible, complete and stable semantics are fully decomposable (note that stable semantics is not directional)
- Grounded, preferred, semistable and ideal semantics are not fully decomposable
- Grounded and preferred semantics are top-down decomposable

Restricting the set of partitions

- One could consider a restricted decomposability focusing on families of partitions with certain properties
- In particular one can focus on partitions whose elements are sets of strongly connected components
- Under this restriction also grounded and preferred semantics are fully decomposable, while ideal and semistable semantics are still not

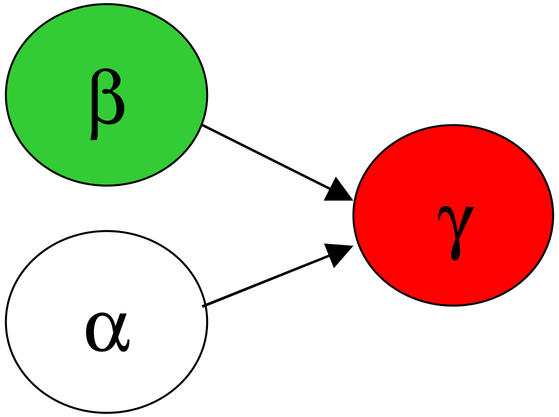
Outline

- Abstract Argumentation (AA)
- Incompleteness in AA: a don't care label
- Incompleteness in AA: partial semantics and decomposability
- **Perspectives and conclusions**

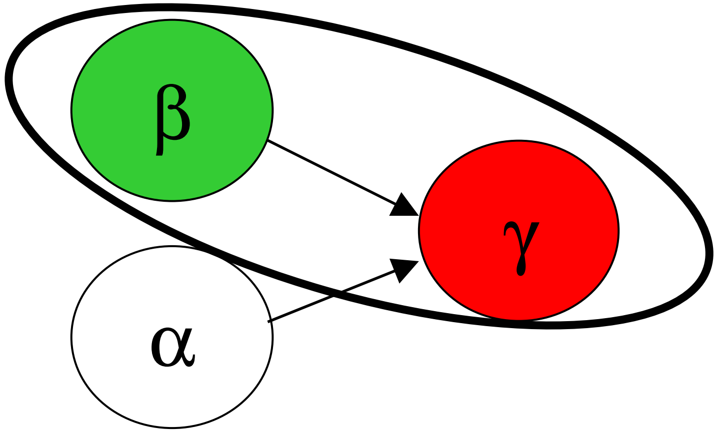
Discussion on incompleteness within argumentation

- The JV approach (explicit don't care labels) and the partial semantics approach (framework restriction) are technically very different but conceptually similar
- Both involve some constraints but their relations have not been investigated yet
- They are not the same and could be fruitfully combined

Different constraints



Ignoring α is legal in the JV approach



But β,γ is not unattacked
Ignoring α would not be allowed in the restriction-based approaches

Incompleteness in belief revision?

- I am not aware of any approach involving incompleteness in belief revision
- A naive Google search of "partial belief revision" did not give me further information
- Revision of a belief base rather than of a belief set potentially encompasses some form of incompleteness
- Partial evaluation is very important for argumentation dynamics, so it can be for iterated belief revision

Incompleteness in practical reasoning

- Incompleteness is not a bug but a feature of most practical reasoning activities in real life
- Yet, most theoretical models tend to be omni-* (omniscient, omnicomprehensive, omnicomputing) and to consider incompleteness as an accident in the end, rather than a feature in the beginning
- Some specialised treatments of incompleteness are available in argumentation and might be available also in other reasoning models, including belief revision

Incompleteness in practical reasoning

- Comparing treatments of incompleteness in different areas both conceptually and technically
- Cross fertilization and reuse/exchange of ideas
- General theory of incompleteness in dynamic practical reasoning
- BR and ARG communities could start this process

Some references/related works (in implicit order of appearance)

- 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):321-357, 1995.
- P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. Knowledge Engineering Review, 26(4):365{410, 2011.
- P. Baroni, M. Giacomin, B. Liao, I don't care, I don't know … I know too much! On incompleteness and undecidedness in abstract argumentation, in T. Eiter, H. Strass, M. Truszczyński, S. Woltran (Eds.), Advances in Knowledge Representation, Logic Programming and Abstract Argumentation., LNCS, Vol. 9060, Springer, 2015, 265-280
- H. Jakobovits and D. Vermeir. Robust semantics for argumentation frameworks J. of Logic and Computation, 9(2):215-261, 1999.
- P. Baroni and M. Giacomin. On principle-based evaluation of extension-based argumentation semantics. Artif. Intell., 171(10/15):675-700, 2007.
- B. Liao and H. Huang. Partial semantics of argumentation: basic properties and empirical results. J. of Logic and Computation, 23(3):541-562, 2013.
- R. Baumann. Splitting an argumentation framework. In Proc. of LPNMR 2011, 11th Int. Conf. on Logic Programming and Nonmonotonic Reasoning, pages 40-53,2011.
- B. Liao, L. Jin, and R. C. Koons. Dynamics of argumentation systems: A division-based method. Artif. Intell., 175(11):1790-1814, 2011.

Some references/related works (in implicit order of appearance)

- P. Baroni, M. Giacomin, and G. Guida. SCC-recursiveness: a general schema for argumentation semantics. Artif. Intell., 168(1-2):165{210, 2005.
- P. Baroni, M. Giacomin, and B. Liao. On topology-related properties of abstract argumentation semantics. A correction and extension to Dynamics of argumentation systems: A division-based method. Artif. Intell., 212:104-115, 2014.
- P. Baroni, G. Boella, F. Cerutti, M. Giacomin, L. W. N. van der Torre, and S. Villata. On input/output argumentation frameworks. In Proc. of the 4th Int. Conf. on Computational Models of Argument (COMMA 2012), pages 358-365, 2012.
- P. Baroni, G. Boella, F. Cerutti, M. Giacomin, L. W. N. van der Torre, and S. Villata. On the input/output behavior of argumentation frameworks. Artif. Intell.,217:144-197, 2014.
- B. Liao. Toward incremental computation of argumentation semantics: A decomposition-based approach. Ann. Math. Artif. Intell., 67(3-4):319{358, 2013.
- R. Baumann, G. Brewka, W. Dvorak, and S. Woltran. Parameterized splitting: A simple modification-based approach. In E. Erdem, J. Lee, Y. Lierler, and D. Pearce, editors, Correct Reasoning - Essays on Logic-Based AI in Honour of Vladimir Lifschitz, pages 57-71. Springer, 2012.
- T. Rienstra, A. Perotti, S. Villata, D. Gabbay, L. van Der Torre, and G. Boella. Multi-sorted argumentation frameworks. In Theory and Applications of Formal Argumentation - First Int. Workshop (TAFA 2011). Revised Selected Papers, pages 231-245. Springer, 2011.


I JUST
DON'T
CARE