Intertranslatability of Abstract Argumentation Frameworks

Polberg Sylwia
University College London

Cardiff Argumentation Meeting
July 2016

Roadmap

Scenarios I

So let us assume you want to use argumentation in your project and know what sort of features you will need...

Scenario 1

- You find two argumentation frameworks that are "almost" good, but each one would have to be extended with a missing feature that is present in the other structure
- What do you do?
  - Create another framework joining the two?
  - Perhaps find an easy way to simulate the missing feature?

Scenarios II

Scenario 2

- You find two argumentation frameworks that have what you want, but you have problem choosing between them...
- ...and what you keep finding are interesting observations about their differences, but no hard facts that you can really use to defend your choice
- What do you do?
  - Go with the general opinion?
  - Throw a dice?
  - Or see what it would take for one framework to emulate the behaviour of the other?

Scenarios III

Scenario 3

- You find an argumentation framework that is just right.
- ...but it's computational complexity is not analyzed and it does not have an implementation
- What do you do?
  - Find a different framework?
  - Fill in the research gaps yourself?
  - ...or just use a translation?

Scenarios IV

Scenario 4

- You find an argumentation framework that is just right and work with it
- ...but then a reviewer complains your choice was unnecessary and that "with a bit of effort the Dung's framework could have handled it"
- What do you do?
  - Talk about your preferences? How the framework is easier to use in your application than Dung's? Hope he/she will get that?
  - ...or, if it is helpful, show him actual translations, their computational complexity, the impossibility proofs, straight facts that he or she cannot deny?

Scenarios V

Scenario 5

- You had to create a new framework to handle what you want...
- ...and now need to explain how it is related to other works in the field
- It would be awesome if you could create:
  - Scenarios handled differently between the frameworks
  - A way for your framework to handle the existing ones
  - The effort it would take for other structures to emulate yours
- How can you come up with such things?

Our Work I

Our motivation

Intertranslatibility research can be used in:

- Designing argumentation–based software
- Widening the application and instantiation range of a given framework
- Research of framework dedicated solvers
- Comparing expressive power of given frameworks
- Studying the meaning and the “added value” of framework components
- Establishing the connections between different framework components

Our Work II

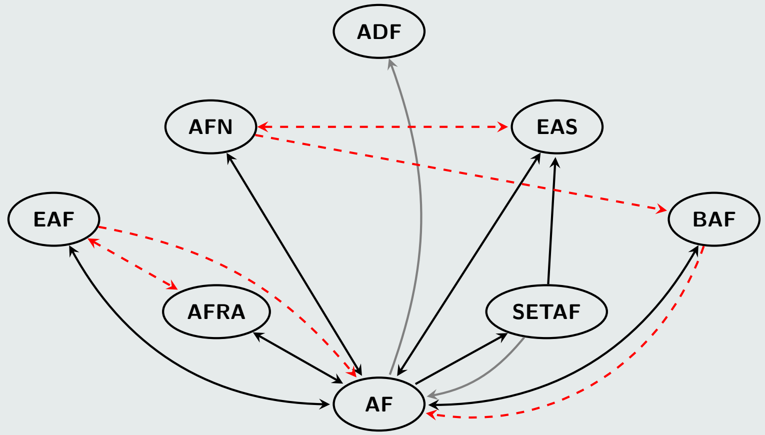
Argumentation Frameworks

Abstract argumentation is more than Dung's framework. There exist many different types (BPW14), including:

- Attack frameworks:
  - Dung's Frameworks (AF) (Dun95)
  - Framework with Sets of Attacking Arguments (SETAF) (NP07)
  - Framework with Recursive Attack (AFRA) (BCGG11)
  - Extended Argumentation Framework (EAF) (Mod09)
- Support frameworks:
  - Bipolar Argumentation Framework (BAF) (CLS09; CLS13)
  - Argumentation Framework with Necessities (AFN) (Nou13)
  - Evidential Argumentation System (EAS) (ORL10; PO14)
  - Abstract Dialectical Framework (ADF) (BW10; BES+13; Pol15)

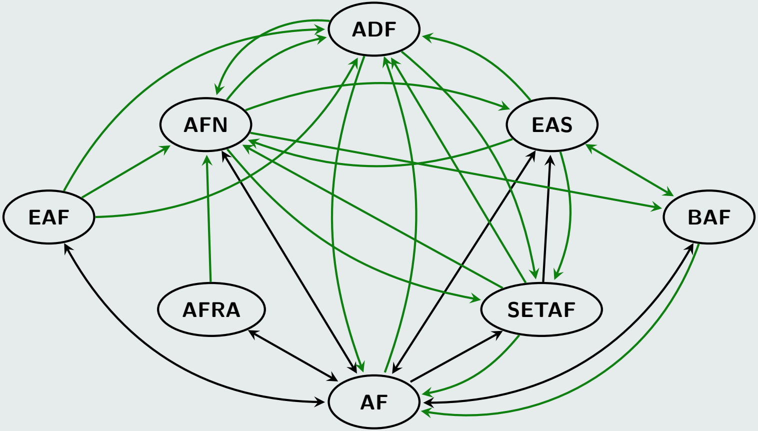
Contributions

Existing translations



Contributions

New translations

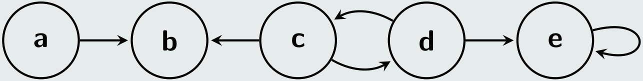


Dung's Framework (Dun95)

Dung's framework

A **Dung's abstract argumentation framework** (AF) is a pair (A, R), where A is a set of **arguments** and $R \subseteq A \times A$ represents the **attack** relation.

Example

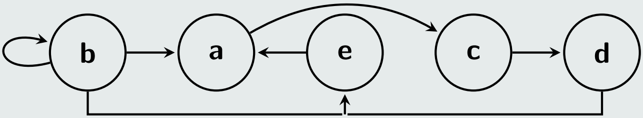


Framework with Sets of Attacking Arguments (NP07) I

Framework with Sets of Attacking Arguments

A **Framework with Sets of Attacking Arguments** (SETAF) is a pair (A, R), where A is a set of **arguments** and $R \subseteq (2^A \setminus \emptyset) \times A$ represents the **attack** relation.

Example

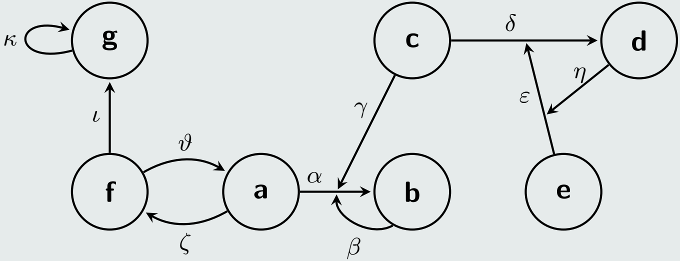


Argumentation Framework with Recursive Attacks (BCGG11)

Argumentation Framework with Recursive Attacks

An **argumentation framework with recursive attacks** (AFRA) is a pair (A, R) where A is a set of **arguments** and R is a set of **attacks**, namely pairs (a, X) s.t. $a \in A$ and $X \in A \cup R$.

Example

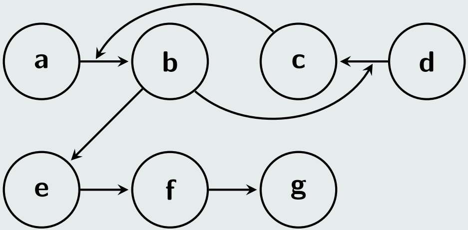


Extended Argumentation Framework (MP10)

Extended Argumentation Framework

The **extended argumentation framework** (EAF) is a tuple (A, R, D), where A is a set of **arguments**, $R \subseteq A \times A$ is the **attack** relation, $D \subseteq A \times R$ is the **defense attack** relation.

Example

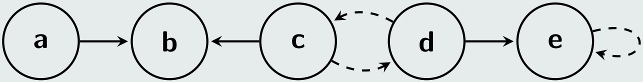


Bipolar Argumentation Frameworks (CLS13)

Bipolar Argumentation Framework

The **bipolar argumentation framework** (BAF) is a tuple (A, R, S), where A is a set of **arguments**, $R \subseteq A \times A$ represents the **attack** relation and $S \subseteq A \times A$ the **support** relation.

Example

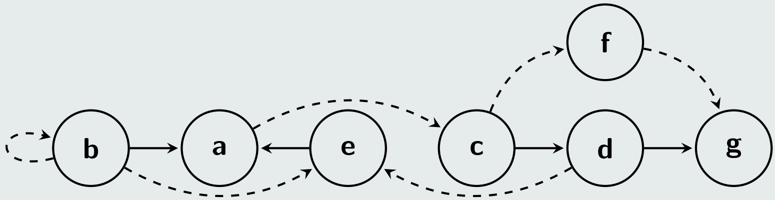


Argumentation Framework with Necessities (Nou13) I

Argumentation Framework with Necessities

An **argumentation framework with necessities** is a tuple (A, R, N), where A is the set of **arguments**, $R \subseteq A \times A$ represents (binary) **attacks**, and $N \subseteq (2^A \setminus \emptyset) \times A$ is the **necessity relation**.

Example

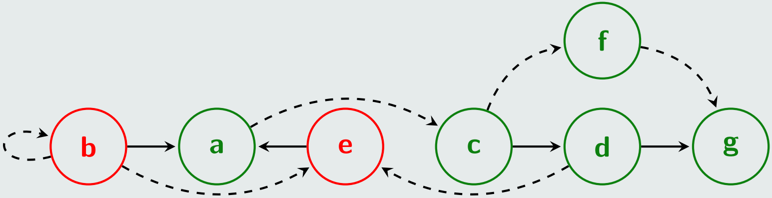


Argumentation Framework with Necessities (Nou13) II

Argumentation Framework with Necessities

An **argumentation framework with necessities** is a tuple (A, R, N), where A is the set of **arguments**, $R \subseteq A \times A$ represents (binary) **attacks**, and $N \subseteq (2^A \setminus \emptyset) \times A$ is the **necessity relation**.

Example

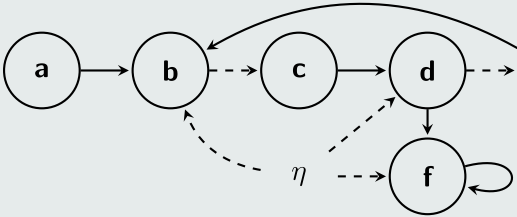


Evidential Argumentation System (ON08; ORL10; PO14)

Evidential Argumentation System

An **evidential argumentation system** (EAS) is a tuple (A, R, E) where A is a set of **arguments**, $R \subseteq (2^A \setminus \emptyset) \times A$ is the **attack** relation, and $E \subseteq (2^A \setminus \emptyset) \times A$ is the **support** relation. We distinguish a special argument $\eta \in A$ s.t. $\nexists (X, y) \in R$ where $\eta \in X$; and $\nexists X$ where $(X, \eta) \in R$ or $(X, \eta) \in E$.

Example

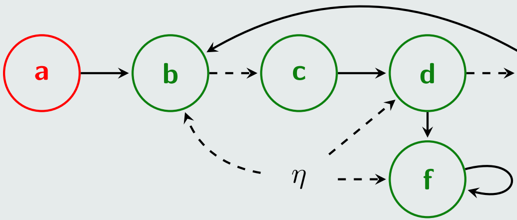


Evidential Argumentation System (ON08; ORL10; PO14)

Evidential Argumentation System

An **evidential argumentation system** (EAS) is a tuple (A, R, E) where A is a set of **arguments**, $R \subseteq (2^A \setminus \emptyset) \times A$ is the **attack** relation, and $E \subseteq (2^A \setminus \emptyset) \times A$ is the **support** relation. We distinguish a special argument $\eta \in A$ s.t. $\nexists (X, y) \in R$ where $\eta \in X$; and $\nexists X$ where $(X, \eta) \in R$ or $(X, \eta) \in E$.

Example



Abstract Dialectical Framework (BES+13; Pol15) I

Definition

An **abstract dialectical framework** (ADF) is a tuple (S, L, C), where:

- S is a set of abstract **arguments** (nodes, statements),
- $L \subseteq S \times S$ is a set of **links** (edges) and
- $C = \{C_s\}_{s \in S}$ is a set of **acceptance conditions**, one condition per each argument.

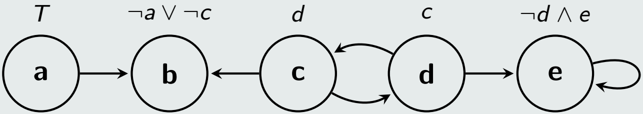
Important: links now do not represent relations anymore; the precise nature of the interaction between arguments is specified by the acceptance conditions.

Acceptance conditions

- They represent the relation of an argument to its parents
- Can be represented as functions $C_s : 2^{par(s)} \to \{in, out\}$
- More commonly defined as propositional formulas

Abstract Dialectical Framework (BES+13; Pol15) II

Example



Semantics

- Labeling–based (BES+13): implemented in DIAMOND (ES13)
- Extension–based (Pol15): four families (AA, CC, AC, CA_1 and CA_2), to be implemented

Translation

Intuition

" *...translation can be understood as a* **function** *Tr which maps theories from one formalism into another such that intended models of a theory* Δ *from the source formalism are in a* **certain relation** *to the intended models of* $Tr(\Delta)$. " ((DW11, 1))

Abstract Argumentation Translations

Let S, T be two framework types between which we want to translate and σ, δ source and target semantics. We distinguish:

- Semantics translations – same framework type, different semantics (DW11)
- Framework translations – different framework types, same semantics
- Normal form translations – same framework type, same semantics (CK14)

Properties of Translations I

Functional Properties

Look at a translation as a function:

- Can it handle any source framework, or just some subclass?
- Can any target framework be produced, or just some subclass?
- Does it produce same target framework for more than one source framework?
- If yes, what is the relation between the source frameworks?

Properties of Translations II

Complexity Properties

Look at how difficult the translation is:

- Is it purely structural, or does it require computing some basic semantics?
- Is it modular?
- What is the computation time?
- Does it cause any blow up in size of the target framework?

Syntactical Properties

Look at what it does to framework components:

- Does it change the type of arguments or not?
- Does it introduce auxiliary arguments and relations or not?
- Does it remove certain arguments and relations or not?

Properties of Translations III

Semantical Properties

Look at how the semantics of the frameworks behave:

- Is the translations specialized for a particular semantics, or is it generic?
- Is the semantics' domain the same?
- How strong is the translation?
- Is the translation bijective?
- Does the translation introduce auxiliary arguments in the extensions?

Existing Notions

Typical translation properties include (Got95; Lib14; Jan99):

- Modularity
- Efficiency, polynomiality
- Exactness, faithfulness

Properties of Translations IV

Exactness and Faithfulness

- Strong translation – every target extension corresponds to a source one and vice versa
- Semantics bijective translation – it's strong and there is a one to one relation between target and source extensions
- Faithful translation – it's semantics bijective and the original extensions are retrieved by removing auxiliary arguments
- Exact translation – it's semantics bijective and the target extensions are exactly the same as source ones

Translation Approaches I

Possible Approaches

Some translations are **easy** and our target framework can handle everything that the source one does. Some however, **are not**. When one structure possesses a feature the other does not, we can:

- Hide it
- Simulate it
- Remove it
- Limit ourselves to cases in which it does not occur

Translation Approaches II

Translation Types

We can distinguish four main types of translations (BGvdTV09; MBC11; CLS13; CL15; ORL10; PO14; BGvdTV10):

- **Basic** – when going from less to more complex frameworks, usually target framework can handle all elements of the source one
- **Coalition** – from more to less complex structures, not handled elements are hidden away in argument structure
- **Attack Propagation** – from more to less complex structures, effect of not handled elements is simulated by handled ones
- **Defender** – from more to less complex structures, not handled elements are translated into handled ones with the use of auxiliary arguments

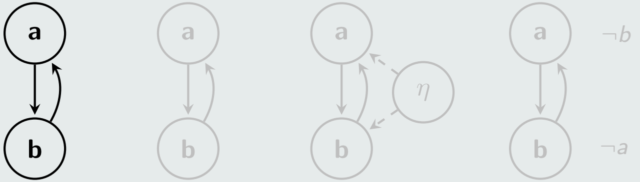
Basic Translation I

Basic translation

- A simple translation, often a generalization
- Never semantical
- On average, it does not require too many auxiliary arguments
- Preserves the structure of the source framework
- Generic, usually preserves all standard semantics in at least faithful manner

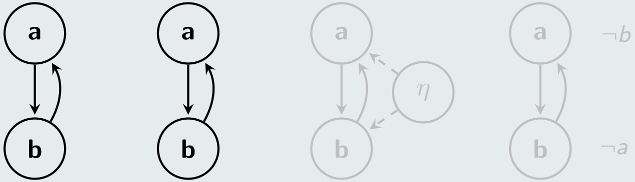
Basic Translation II

Example



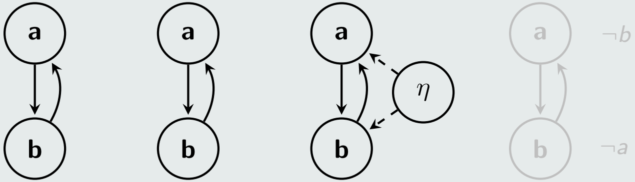
Basic Translation III

Example



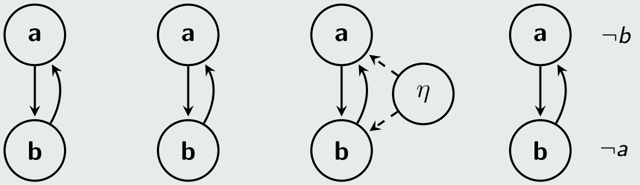
Basic Translation IV

Example



Basic Translation V

Example



Basic Translation: Summary

	AF	AFRA	SETAF	EAF	BAF	AFN	EAS	ADF
AF	x	✓ [green]	✓ [green]	✓ [green]	✓ [green]	✓ [green]	✓ [green]	✓ [green]
AFRA	✓ [red], ✓ [red]	x			✓ [green]	✓ [green]		
SETAF	✓ [red], ✓ [red]		x			✓ [red]	✓ [green]	✓ [green]
EAF	✓ [red]	✓ [blue]	✓ [blue]	x		✓ [blue], ✓ [blue]		✓ [blue], ✓ [green]
BAF	✓ [red], ✓ [red]				x	✓ [blue]	✓ [blue]	
AFN					✓ [blue]	x	✓ [green]	✓ [blue], ✓ [green]
EAS					✓ [blue]	✓ [blue], ✓ [red]	x	✓ [blue], ✓ [green]
ADF								x

✓ [green] – translation
✓ [red] – hybrid translation
✓ [blue] – subclass translation

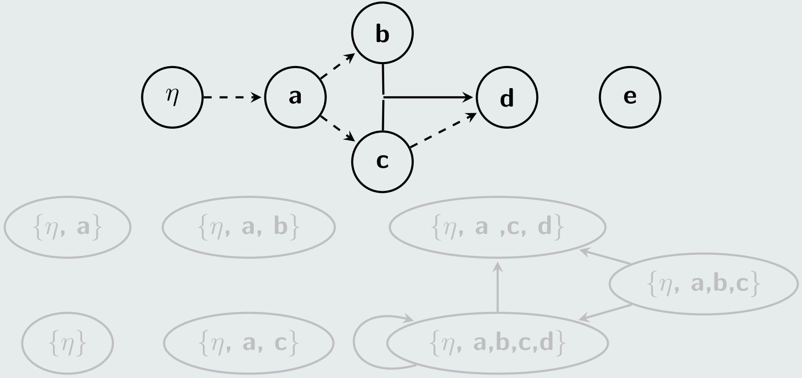
Coalition Translation I

Coalition translation

- Arguments in the target framework are collections of source arguments that are tied by support or ability to carry out a group attack
- Almost always semantical
- Exponential increase in amount of required auxiliary arguments
- Translation is lossy, it removes arguments from the source framework
- Usually preserves most of the standard semantics in a strong to semantics bijective manner

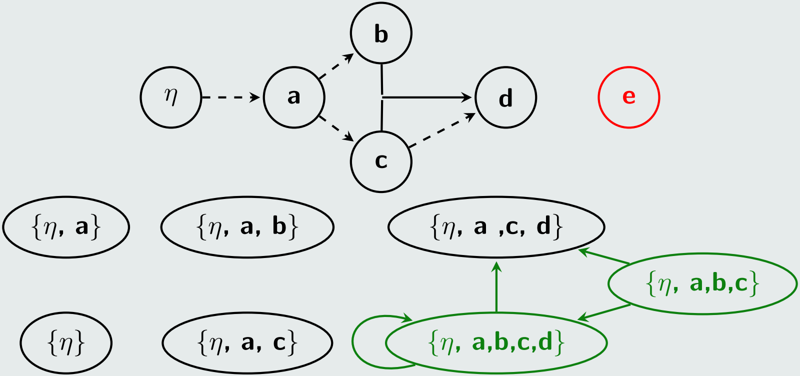
Coalition Translation II

Example



Coalition Translation III

Example



Coalition Translation: Summary

	AF	AFRA	SETAF	EAF	BAF	AFN	EAS	ADF
AF	x							
AFRA		x						
SETAF	✓		x			✓		
EAF				x				
BAF	✓				x			
AFN	✓					x		
EAS	✓		✓			✓	x	
ADF	✓, ✓, ✓, ✓		✓					x

✓ – translation
✓ – hybrid translation

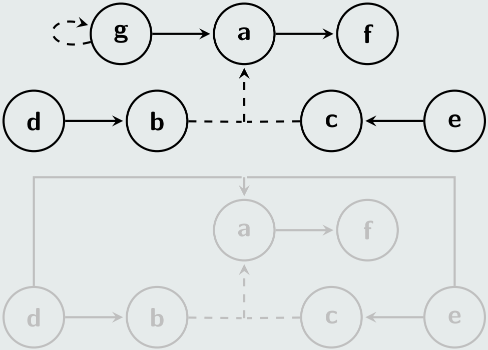
Attack Propagation Translation I

Attack propagation translation

- Completes the source framework with various types of indirect attacks
- Does not require auxiliary arguments
- Removes some of the arguments in the source framework
- In principle, the translation is semantical; can be structural only for particular normal forms
- Preserves completeness–based semantics in an exact manner

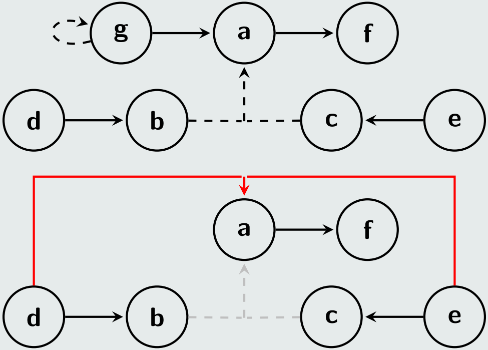
Attack Propagation Translation II

Example



Attack Propagation Translation III

Example



Attack Propagation: Summary

	AF	AFRA	SETAF	EAF	BAF	AFN	EAS	ADF
AF	x							
AFRA	✓ (red)	x						
SETAF			x					
EAF				x				
BAF	✓ (red)				x			
AFN			✓			x		
EAS			✓				x	
ADF			✓					x

✓ – translation

✓ (red) – hybrid translation

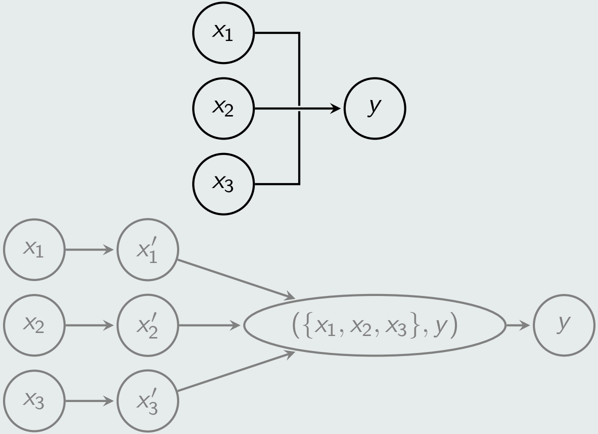
Defender Translation

Defender translation

- Exploit defense to e.g. simulate support or to connect arguments
- Does not remove arguments from the source framework
- For attack–based frameworks, the translation:
  - is structural
  - can require exponentially may auxiliary arguments
- For support–based frameworks:
  - it is semantical and can be structural only for particular normal forms
  - requires polynomially many auxiliary arguments
- Usually preserves semantics that are at least admissible in a strong to faithful manner

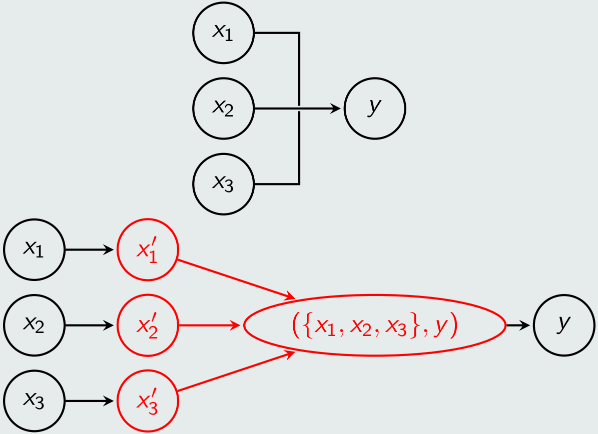
Defender Translation

Example



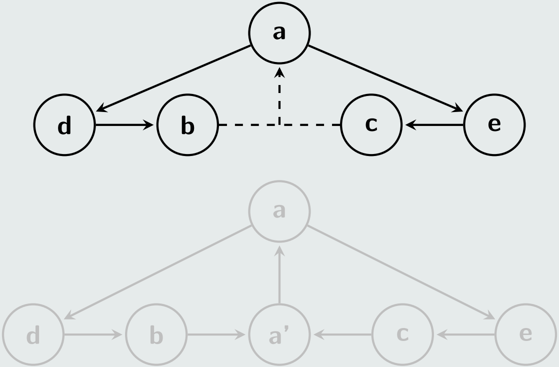
Defender Translation

Example



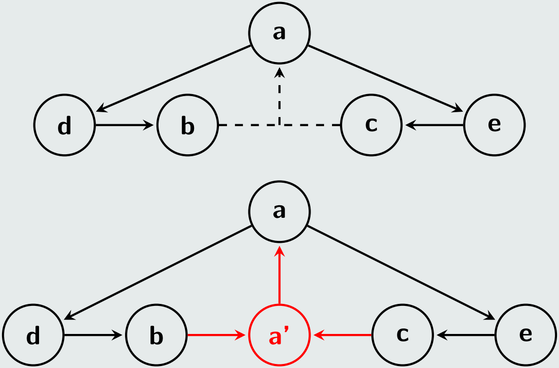
Defender Translation

Example



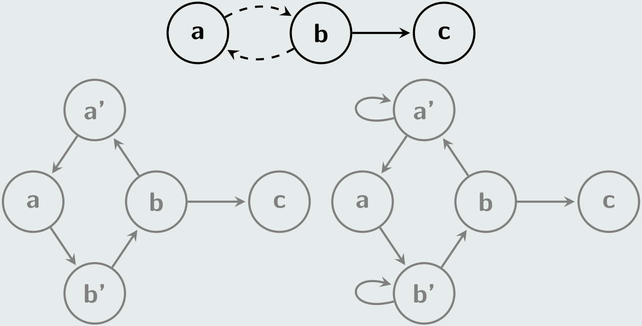
Defender Translation

Example



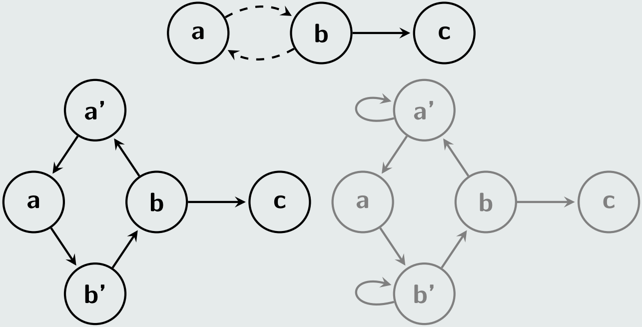
Defender Translation

Example



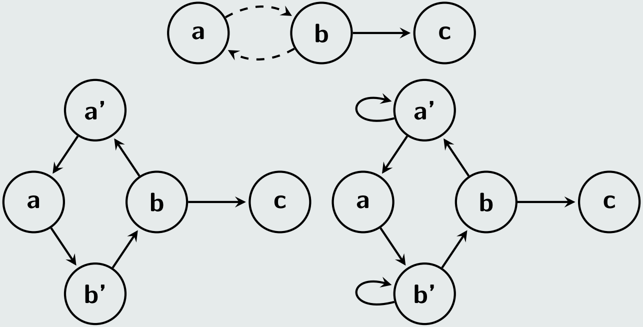
Defender Translation

Example



Defender Translation

Example



Defender Translation: Summary

	AF	AFRA	SETAF	EAF	BAF	AFN	EAS	ADF
AF	x							
AFRA	✓	x						
SETAF	✓		x					
EAF	✓			x				
BAF					x			
AFN			✓			x		
EAS			✓				x	
ADF	✓, ✓		✓					x

✓ – translation

✓ – hybrid translation

Improving Translations

	AF	AFRA	SETAF	EAF	BAF	AFN	EAS	ADF
AF	x	✓	✓	✓	✓	✓	X	✓
AFRA	✓	x			✓	✓	X	
SETAF	X		x			X	✓	✓
EAF	X	X	X	x	X	X	X	✓
BAF					x			
AFN	✓ (green)		✓			x	X	✓
EAS	X		✓			X	x	✓
ADF	X	X	X	X?	X	X	X	x

✓ – is exact

✓ (green) – exact might exist

X – exact most likely does not exist


this is how I finish a presentation:
happymonsters.tumblr.com
Soo...
uhh
...yeah.

References I

Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida.
AFRA: Argumentation framework with recursive attacks.
Int. J. Approx. Reasoning, 52(1):19–37, 2011.

Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes Peter Wallner, and Stefan Woltran.
Abstract dialectical frameworks revisited.
In *Proc. IJCAI'13*, pages 803–809. AAAI Press, 2013.

Guido Boella, Dov M. Gabbay, Leendert van der Torre, and Serena Villata.
Meta-argumentation modelling I: Methodology and techniques.
Studia Logica, 93(2-3):297–355, 2009.

References II

Guido Boella, Dov Gabbay, Leendert van der Torre, and Serena Villata.
Support in abstract argumentation.
In *Proc. of COMMA 2010*, pages 111–122, Amsterdam, The Netherlands, The Netherlands, 2010. IOS Press.

Gerhard Brewka, Sylwia Polberg, and Stefan Woltran.
Generalizations of Dung frameworks and their role in formal argumentation.
Intelligent Systems, IEEE, 29(1):30–38, Jan 2014.

Gerhard Brewka and Stefan Woltran.
Abstract dialectical frameworks.
In *Proc. KR '10*, pages 102–111. AAAI Press, 2010.

References III

Cosmina Croitoru and Timo Kötzing.
A normal form for argumentation frameworks.
In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Theory and Applications of Formal Argumentation*, volume 8306 of *Lecture Notes in Computer Science*, pages 32–45. Springer Berlin Heidelberg, 2014.

Claudette Cayrol and Marie-Christine Lagasquie-Schiex.
An axiomatic approach to support in argumentation.
In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Theory and Applications of Formal Argumentation - Third International Workshop, TAFA 2015, Buenos Aires, Argentina, July 25-26, 2015, Revised Selected Papers*, volume 9524 of *Lecture Notes in Computer Science*, pages 74–91. Springer, 2015.

References IV

Claudette Cayrol and Marie-Christine Lagasquie-Schiex.
Bipolar abstract argumentation systems.
In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 65–84. 2009.

Claudette Cayrol and Marie-Christine Lagasquie-Schiex.
Bipolarity in argumentation graphs: Towards a better understanding.
Int. J. Approx. Reasoning, 54(7):876–899, 2013.

Phan Minh Dung.
On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games.
Artif. Intell., 77:321–357, 1995.

Wolfgang Dvořák and Stefan Woltran.
On the intertranslatability of argumentation semantics.
J. Artif. Int. Res., 41(2):445–475, 2011.

References V

Stefan Ellmauthaler and Hannes Strass.
The DIAMOND system for argumentation: Preliminary report.
In Michael Fink and Yuliya Lierler, editors, *Proc. ASPOCP*, 2013.

Georg Gottlob.
Translating default logic into standard autoepistemic logic.
J. ACM, 42(4):711–740, July 1995.

Tomi Janhunen.
On the intertranslatability of non–monotonic logics.
Annals of Mathematics and Artificial Intelligence, 27(1-4):79–128, 1999.

Paolo Liberatore.
Bijective faithful translations among default logics.
Journal of Logic and Computation, 2014.

References VI

Sanjay Modgil and Trevor J. M. Bench-Capon.
Metalevel argumentation.
J. Log. Comput., 21(6):959–1003, 2011.

Sanjay Modgil.
Reasoning about preferences in argumentation frameworks.
Artif. Intell., 173(9-10):901–934, 2009.

Sanjay Modgil and Henry Prakken.
Reasoning about preferences in structured extended argumentation frameworks.
In *Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010.*, pages 347–358, 2010.

References VII

Farid Nouioua.
AFs with necessities: Further semantics and labelling characterization.
In Weiru Liu, V.S. Subrahmanian, and Jef Wijsen, editors, *Proc. SUM '13*, volume 8078 of *LNCS*, pages 120–133. Springer Berlin Heidelberg, 2013.

Søren Nielsen and Simon Parsons.
A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments.
In *Proc. ArgMAS*, volume 4766 of *LNCS*, pages 54–73. Springer, 2007.

Nir Oren and Timothy J. Norman.
Semantics for evidence-based argumentation.
In *Proc. COMMA '08*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 276–284. IOS Press, 2008.

References VIII

Nir Oren, Chris Reed, and Michael Luck.
Moving between argumentation frameworks.
In *Proceedings of the 2010 conference on Computational Models of Argument: Proceedings of COMMA 2010*, pages 379–390, Amsterdam, The Netherlands, The Netherlands, 2010. IOS Press.

Sylwia Polberg and Nir Oren.
Revisiting support in abstract argumentation systems.
In Simon Parsons, Nir Oren, Chris Reed, and Federico Cerutti, editors, *Computational Models of Argument - Proceedings of COMMA 2014*, volume 266 of *Frontiers in Artificial Intelligence and Applications*, pages 369–376. IOS Press, 2014.

Sylwia Polberg.
Revisiting extension–based semantics of abstract dialectical frameworks.
Technical Report DBAI-TR-2014-85, Institute for Information Systems, Technical University of Vienna, 2015.