Hannes Strass

(based on slides by Stefan Woltran and Sarah Gaggl)

Faculty of Computer Science, Institute of Artificial Intelligence, Computational Logic Group

Abstract Argumentation

Lecture 10, 13th Jan 2025 // Foundations of Knowledge Representation, WS 2024/25

Introduction

Argumentation:

The study of processes "concerned with how assertions are proposed, discussed, and resolved in the context of issues upon which several diverging opinions may be held".

[Bench-Capon and Dunne, *Argumentation in AI*, AIJ 171:619-641, 2007]

Formal Models of Argumentation are concerned with

- representation of an argument (i.e. an expression of opinion)
- representation of the relationship between arguments
- resolving conflicts between the arguments ("acceptability")

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 2 of 37

Computational
Logic ∴ Group

Overview

Overall Process

Argumentation Frameworks

AFs – Semantics

AFs – Outlook

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 3 of 37

TECHNISCHE
UNIVERSITÄT
DRESDEN

Computational
Logic ∴ Group

Overall Process

Overall Process

The overall process of using argumentation frameworks consists of the steps listed below.

Starting point: Knowledge base

1. Form arguments
2. Identify conflicts
3. Abstract from internal structure of arguments
4. Resolve conflicts
5. Draw conclusions

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 5 of 37

Computational
Logic ∴ Group

Overall Process – Form Arguments

Consider the following knowledge base:

Knowledge Base

$$\Delta = \{s, r, w, s \rightarrow \neg r, r \rightarrow \neg w, w \rightarrow \neg s\}$$

From this, form arguments:
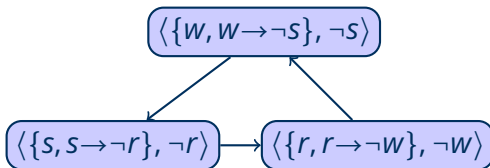
$$\langle \{w, w \rightarrow \neg s\}, \neg s \rangle$$

$$\langle \{s, s \rightarrow \neg r\}, \neg r \rangle \qquad \langle \{r, r \rightarrow \neg w\}, \neg w \rangle$$

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 6 of 37

Computational Logic ∴ Group

Overall Process – Identify Conflicts

Knowledge Base

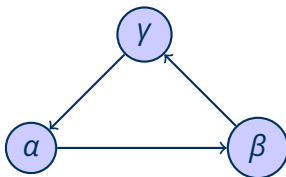$$\Delta = \{s, r, w, s \rightarrow \neg r, r \rightarrow \neg w, w \rightarrow \neg s\}$$

Overall Process – Abstract from Internal Structure of Arguments

Knowledge Base
$\Delta = \{s, r, w, s \rightarrow \neg r, r \rightarrow \neg w, w \rightarrow \neg s\}$

F_Δ:

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide   8 of 37

TECHNISCHE
UNIVERSITÄT
DRESDEN
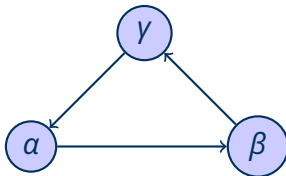
Computational
Logic ∴ Group

Overall Process – Resolve Conflicts

Knowledge Base

$$\Delta = \{s, r, w, s \rightarrow \neg r, r \rightarrow \neg w, w \rightarrow \neg s\}$$
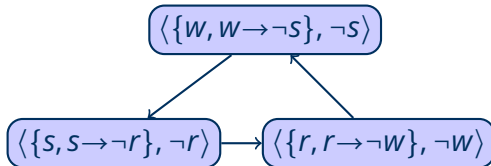
F_Δ:

$pref(F_\Delta) = \{\emptyset\}$

$stage(F_\Delta) = \{\{\alpha\}, \{\beta\}, \{\gamma\}\}$

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 9 of 37

TECHNISCHE
UNIVERSITÄT
DRESDEN

Computational
Logic ∴ Group

Overall Process – Draw Conclusions

Knowledge Base

$$\Delta = \{s, r, w, s \rightarrow \neg r, r \rightarrow \neg w, w \rightarrow \neg s\}$$

$$Cn_{pref}(F_\Delta) = Cn(\top)$$
$$Cn_{stage}(F_\Delta) = Cn(\neg r \vee \neg w \vee \neg s)$$

The Overall Process (ctd.)

Some Remarks

- Main idea dates back to the seminal work of Phan Minh Dung [1995]; has then been refined by several authors (Prakken, Gordon, Caminada, …)
- Separation between logical (forming arguments) and non-monotonic reasoning ("abstract argumentation frameworks")
- Abstraction allows to compare several KR formalisms on a conceptual level ("calculus of conflict")

Main Challenge

- All steps in the argumentation process are, in general, intractable.
- This calls for:
  - careful complexity analysis (identification of tractable fragments)
  - re-use of established tools for implementations (reduction method)

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 11 of 37

Computational
Logic ∴ Group

Approaches to Form Arguments

Classical Arguments [Besnard & Hunter, 2001]

- Given is a KB (a set of propositional formulas) Δ
- An **argument** is a pair (Φ, α), such that $\Phi \subseteq \Delta$ is satisfiable, $\Phi \models \alpha$ and for no $\Psi \subsetneq \Phi$, $\Psi \models \alpha$
- An argument (Φ, α) **attacks** argument (Φ', α') iff $\Phi' \cup \{\alpha\}$ is unsatisfiable
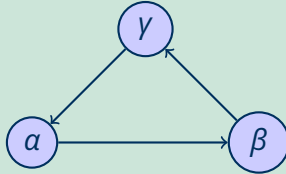
Example

$$\langle \{s, s \to \neg r\}, \neg r \rangle \longrightarrow \langle \{r, r \to \neg w\}, \neg w \rangle$$

Other Approaches:
- arguments are trees (or directed acyclic graphs) of statements
- claims are obtained via strict and defeasible rules
- different notions of conflict: rebuttal, undercut, etc.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 12 of 37

Computational
Logic ∴ Group

Argumentation Frameworks

Dung's Abstract Argumentation Frameworks

Example

Main Properties

- Abstract from the concrete content of arguments; only consider the relation between them
- Semantics select subsets of arguments respecting certain criteria
- Simple, yet powerful formalism
- Most active research area in the field of argumentation

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 14 of 37

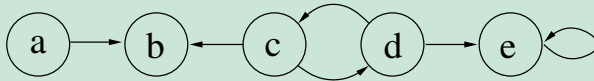Computational Logic ∴ Group

Dung's Abstract Argumentation Frameworks

An **argumentation framework** (AF) is a pair (A, R) where
- A is a set of arguments,
- $R \subseteq A \times A$ is a relation representing the conflicts ("attacks").

Example

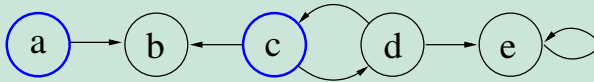$$F = (\{a, b, c, d, e\},\ \{(a, b), (c, b), (c, d), (d, c), (d, e), (e, e)\})$$

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 15 of 37

Computational
Logic ∴ Group

Basic Properties (1)

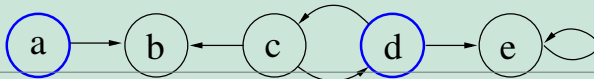**Definition (Conflict-Free Sets)**

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is **conflict-free** in F iff for each $a, b \in S$ we have $(a, b) \notin R$.

Example

$cf(F) = \{\{a, c\},$

Example

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 16 of 37

Computational Logic ∴ Group

$cf(F) = \{\{a, c\}, \{a, d\},$
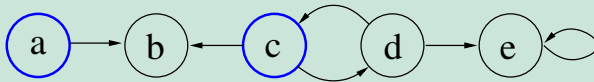
Basic Properties (2)

Definition (Admissible Sets [Dung, 1995])

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is **admissible** in F iff

- S is conflict-free in F,
- each $a \in S$ is defended by S in F, where
  - $a \in A$ is **defended** by S in F iff for each $b \in A$ with $(b, a) \in R$, there exists a $c \in S$ such that $(c, b) \in R$.

Example

$adm(F) = \{\{a, c\},$

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 17 of 37

Computational
Logic ∴ Group

Basic Properties (3)

Dung's Fundamental Lemma

Let S be admissible in an AF F and a, a' arguments in F defended by S in F. Then,

1. $S' = S \cup \{a\}$ is admissible in F.
2. a' is defended by S' in F.

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 18 of 37

TECHNISCHE
UNIVERSITÄT
DRESDEN

Computational
Logic ∴ Group

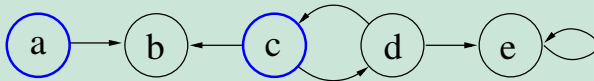# AFs – Semantics

Naive

Definition (Naive Sets/Extensions)

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is **naive** in F iff

- S is conflict-free in F,
- there is no conflict-free $T \subseteq A$ in F such that $S \subsetneq T$.

Naive sets are subset-maximally conflict-free sets.
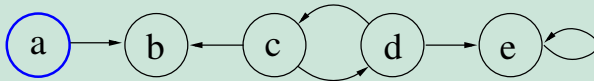
Example

$naive(F) = \{\{a, c\},$

Example

Computational
Logic ∴ Group

TECHNISCHE
UNIVERSITÄT
DRESDEN

Grounded

Definition (Grounded Extension [Dung, 1995])

Let $F = (A, R)$ be an AF. The unique **grounded extension** of F is defined as the outcome S (initially empty) of the following "algorithm":

1. put each argument $a \in A$ that is not attacked in F into S; if no such arguments exist, return S;

2. remove from F all (new) arguments in S and all arguments attacked by them (together with all adjacent attacks); and continue with Step 1.

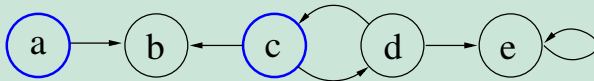**Example**

$ground(F) = \{\{a\}\}$

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 21 of 37

Computational Logic ∴ Group

Complete

Let (A, R) be an AF.

A set $S \subseteq A$ is **complete** in F iff

- S is admissible in F,
- each $a \in A$ defended by S in F is contained in S.
  - Recall: $a \in A$ is defended by S in F iff for each $b \in A$ with $(b, a) \in R$, there exists a $c \in S$, such that $(c, b) \in R$.

Example

$comp(F) = \{\{a, c\},$

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 22 of 37

Computational Logic ∴ Group

Grounded vs. Complete

Properties of the Grounded Extension

For any AF F, the grounded extension of F is the subset-least complete extension of F.

Remark

Since there exists exactly one grounded extension for each AF F, we often write $ground(F) = S$ instead of $ground(F) = \{S\}$.

Preferred
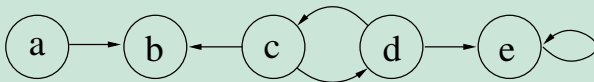
Definition (Preferred Extensions [Dung, 1995])

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is a **preferred extension** of F iff

- S is admissible in F,
- there is no admissible $T \subseteq A$ in F such that $S \subsetneq T$.

Preferred extensions are subset-maximally admissible sets.

Example

$pref(F) = \{\{a, c\}, \{a, d\}, \{a\}, \{c\}, \{d\}, \emptyset\}$

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 24 of 37

TECHNISCHE UNIVERSITÄT DRESDEN
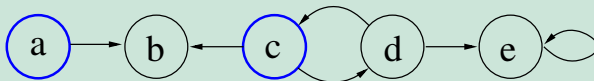
Computational Logic ∴ Group

Stable

Definition (Stable Extensions [Dung, 1995])

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is a **stable extension** of F iff
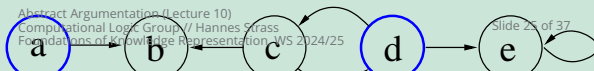
- S is conflict-free in F,
- for each $a \in A \setminus S$, there exists a $b \in S$ such that $(b, a) \in R$.

Example

$stable(F) = \{\{a, c\}$

Example

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group / Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 25 of 37

Computational
Logic ∴ Group

Relationships Between Semantics

Proposition

For any AF F the following relations hold:

1. Each stable extension of F is also a preferred one;
2. Each preferred extension of F is also a complete one;
3. Each complete extension of F is admissible in F.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

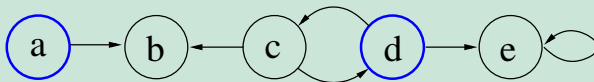Slide 26 of 37

Computational
Logic ∴ Group

Semi-Stable

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is a **semi-stable extension** of F iff

- S is admissible in F,
- there is no admissible $T \subseteq A$ in F such that $S^+ \subsetneq T^+$, where
  - for $S \subseteq A$, define $S^+ = S \cup \{a \in A \mid \exists b \in S \text{ with } (b, a) \in R\}$.

Defined as admissible stages by Verheij [1996].

Example

$semi(F) = \{\{a, c\}, \{a, d\}, \{a\}, \{c\}, \{d\}, \emptyset\}$

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 27 of 37

Computational Logic ∴ Group

Stage

Definition (Stage Extensions [Verheij, 1996])

Let $F = (A, R)$ be an AF.

A set $S \subseteq A$ is a **stage extension** of F iff

- S is conflict-free in F,
- there is no conflict-free $T \subseteq A$ in F such that $S^+ \subsetneq T^+$.
  - Recall: $S^+ = S \cup \{a \in A \mid \exists b \in S \text{ with } (b, a) \in R\}$.

S^+ is also called the range of S. Thus:

- Semi-stable extensions are range-maximally admissible sets.
- Stage extensions are range-maximally conflict-free sets.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 28 of 37

Computational
Logic ∴ Group

Ideal

Definition (Ideal Extension [Dung, Mancarella & Toni 2007])

Let $F = (A, R)$ be an AF.

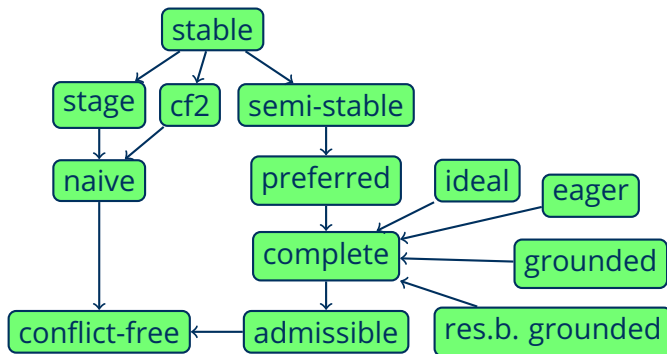A set $S \subseteq A$ is an **ideal extension** of F iff

- S is admissible in F and contained in each preferred extension of F,
- there is no $T \supsetneq S$ admissible in F and contained in each of $pref(F)$.

Properties of Ideal Extensions

For any AF F the following observations hold:

1. There exists exactly one ideal extension of F.
2. The ideal extension of F is also a complete one.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 29 of 37

Computational
Logic ∴ Group

Relations Between Semantics

An arrow from semantics σ to semantics τ means that each σ-extension is also a τ-extension.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 30 of 37

Computational
Logic ∴ Group

AFs – Outlook

Characteristics of Argumentation Semantics

Example

$pref(F) = \{\{a, d, e\}, \{b, c, e\}, \{a, b\}\}$
$naive(F) = \{\{a, d, e\}, \{b, c, e\}, \{a, b, e\}\}$
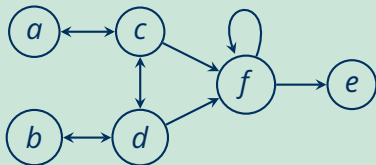
Natural Questions ⤳ Realizability

- How to change the AF if we want $\{a, b, e\}$ instead of $\{a, b\}$ in $pref(F)$?
- How to change the AF if we want $\{a, b, d\}$ instead of $\{a, b\}$ in $pref(F)$?
- Can we have equivalent AFs without argument f?

TECHNISCHE UNIVERSITÄT DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 32 of 37

Computational Logic ∴ Group

Some Properties . . .

Theorem

For any AFs *F* and *G*, we have

- $adm(F) = adm(G)$ implies $\sigma(F) = \sigma(G)$, for $\sigma \in \{pref, ideal\}$;
- $comp(F) = comp(G)$ implies $\theta(F) = \theta(G)$, for $\theta \in \{pref, ideal, ground\}$;
- no other such relation between the different semantics (*adm*, *pref*, *ideal*, *semi*, *ground*, *comp*, *stable*) in terms of standard equivalence holds.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 33 of 37

Computational
Logic ∴ Group

Decision Problems on AFs

Credulous Acceptance

Cred_σ: Given AF $F = (A, R)$ and $a \in A$;
is a contained in at least one σ-extension of F?

Skeptical Acceptance

Skept_σ: Given AF $F = (A, R)$ and $a \in A$;
is a contained in every σ-extension of F?

If no extension exists then all arguments are skeptically accepted and no argument is credulously accepted.[1]

---

[1] This is only relevant for stable semantics.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 34 of 37

Computational
Logic ∴ Group

Decision Problems on AFs

Hence we are also interested in the following problem:

Skeptically and Credulously Accepted

Skept'_σ: Given AF $F = (A, R)$ and $a \in A$;
is a contained in every and at least one σ-extension of F?

TECHNISCHE
UNIVERSITÄT
DRESDEN

Computational
Logic ∴ Group

Further Decision Problems

Verifying an extension

Ver_σ: Given AF $F = (A, R)$ and $S \subseteq A$;
is S a σ-extension of F?

Does there exist an extension?

Exists_σ: Given AF $F = (A, R)$;
Does there exist a σ-extension for F?

Does there exist a nonempty extensions?

$\text{Exists}_\sigma^{\neg\emptyset}$: Given AF $F = (A, R)$;
Does there exist a non-empty σ-extension for F?

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 36 of 37

Computational
Logic ∴ Group

Conclusion

- Abstract argumentation frameworks are constructed from KBs.
- Edges (attacks) between nodes (arguments) express (directed) conflicts.
- A variety of semantics for AFs try to make sense of acceptability:
  - Complete
  - Grounded
  - Preferred
  - Stable
  - …
- Various inclusion relationships between the semantics hold, as they build on similar notions.

TECHNISCHE
UNIVERSITÄT
DRESDEN

Abstract Argumentation (Lecture 10)
Computational Logic Group // Hannes Strass
Foundations of Knowledge Representation, WS 2024/25

Slide 37 of 37

Computational
Logic ∴ Group