Vorlesung “Formale Argumentation”

10. Structured Argumentation – Inkonsistenzbehandlung

Ringo Baumann
Professur für Formale Argumentation
und Logisches Schließen

13. Juni 2024
Leipzig

Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

$$\begin{aligned} r_1 &: \quad \Rightarrow a \\ r_2 &: \quad a \Rightarrow b \\ r_3 &: \quad b \rightarrow not\ a \\ r_4 &: \quad \rightarrow c \\ r_5 &: \quad c \Rightarrow not\ b \end{aligned}$$

Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

$r_1: \quad \Rightarrow a$

$r_2: \quad a \Rightarrow b$

$r_3: \quad b \rightarrow \mathit{not}\ a$

$r_4: \quad \rightarrow c$

$r_5: \quad c \Rightarrow \mathit{not}\ b$

$\Longrightarrow$

2. Arguments

$a_1: \quad [r_1 \mid a]$

$a_2: \quad [a_1, r_2 \mid b]$

$a_3: \quad [a_2, r_3 \mid \mathit{not}\ a]$

$a_4: \quad [r_4 \mid c]$

$a_5: \quad [a_4, r_5 \mid \mathit{not}\ b]$

Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

r_1 :	$\Rightarrow a$
r_2 :	$a \Rightarrow b$
r_3 :	$b \rightarrow \mathit{not}\ a$
r_4 :	$\rightarrow c$
r_5 :	$c \Rightarrow \mathit{not}\ b$

2. Arguments

a_1 :	$[r_1 \mid a]$
a_2 :	$[a_1, r_2 \mid b]$
a_3 :	$[a_2, r_3 \mid \mathit{not}\ a]$
a_4 :	$[r_4 \mid c]$
a_5 :	$[a_4, r_5 \mid \mathit{not}\ b]$

3. Conflicts

c_1 :	a_1 attacks a_3
c_2, c_3 :	a_2 attacks a_3, a_5
c_4, c_5, c_6, c_7 :	a_3 attacks a_1, a_2, a_3, a_5
	a_4 attacks no-one
c_8, c_9 :	a_5 attacks a_2, a_3

Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

$r_1 : \quad \Rightarrow a$

$r_2 : \quad a \Rightarrow b$

$r_3 : \quad b \rightarrow not\ a$

$r_4 : \quad \rightarrow c$

$r_5 : \quad c \Rightarrow not\ b$

2. Arguments

$a_1 : \quad [r_1 \mid a]$

$a_2 : \quad [a_1, r_2 \mid b]$

$a_3 : \quad [a_2, r_3 \mid not\ a]$

$a_4 : \quad [r_4 \mid c]$

$a_5 : \quad [a_4, r_5 \mid not\ b]$

3. Conflicts

$c_1 :$ a_1 attacks a_3

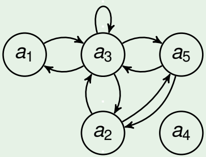
$c_2, c_3 :$ a_2 attacks a_3, a_5

$c_4, c_5, c_6, c_7 :$ a_3 attacks a_1, a_2, a_3, a_5

a_4 attacks no-one

$c_8, c_9 :$ a_5 attacks a_2, a_3

4. Instantiation



Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

$r_1 : \quad \Rightarrow a$
$r_2 : \quad a \Rightarrow b$
$r_3 : \quad b \rightarrow not\ a$
$r_4 : \quad \rightarrow c$
$r_5 : \quad c \Rightarrow not\ b$

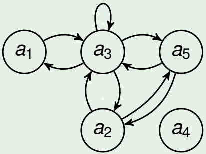
2. Arguments

$a_1 : \quad [r_1 \mid a]$
$a_2 : \quad [a_1, r_2 \mid b]$
$a_3 : \quad [a_2, r_3 \mid not\ a]$
$a_4 : \quad [r_4 \mid c]$
$a_5 : \quad [a_4, r_5 \mid not\ b]$

3. Conflicts

$c_1 :$ a_1 attacks a_3
$c_2, c_3 :$ a_2 attacks a_3, a_5
$c_4, c_5, c_6, c_7 :$ a_3 attacks a_1, a_2, a_3, a_5
a_4 attacks no-one
$c_8, c_9 :$ a_5 attacks a_2, a_3

4. Instantiation



5. Resolving

$$E_1 = \{a_1, a_2, a_4\}$$
$$E_2 = \{a_1, a_4, a_5\}$$

Reconstruction via Argumentation

Example (Rule-based Formalism)

1. Knowledge Base

$r_1 : \quad \Rightarrow a$

$r_2 : \quad a \Rightarrow b$

$r_3 : \quad b \rightarrow \textit{not } a$

$r_4 : \quad \rightarrow c$

$r_5 : \quad c \Rightarrow \textit{not } b$

2. Arguments

$a_1 : \quad [r_1 \mid a]$

$a_2 : \quad [a_1, r_2 \mid b]$

$a_3 : \quad [a_2, r_3 \mid \textit{not } a]$

$a_4 : \quad [r_4 \mid c]$

$a_5 : \quad [a_4, r_5 \mid \textit{not } b]$

3. Conflicts

$c_1 :$ a_1 attacks a_3

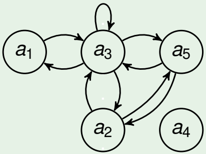
$c_2, c_3 :$ a_2 attacks a_3, a_5

$c_4, c_5, c_6, c_7 :$ a_3 attacks a_1, a_2, a_3, a_5

a_4 attacks no-one

$c_8, c_9 :$ a_5 attacks a_2, a_3

4. Instantiation



5. Resolving

$E_1 = \{a_1, a_2, a_4\}$

$E_2 = \{a_1, a_4, a_5\}$

6. Conclusion

$E_1 = \{a, b, c\}$

$E_2 = \{a, c, \textit{not } b\}$

Conc = {a,c}

Themenüberblick

- Wissensbasis in der Sprache der Aussagenlogik

Themenüberblick

- Wissensbasis in der Sprache der Aussagenlogik
- Inkonsistenzbehandlung

Themenüberblick

- Wissensbasis in der Sprache der Aussagenlogik
- Inkonsistenzbehandlung
- Argumentdefinition

Themenüberblick

- Wissensbasis in der Sprache der Aussagenlogik
- Inkonsistenzbehandlung
- Argumentdefinition
- Attackdefinition

Themenüberblick

- Wissensbasis in der Sprache der Aussagenlogik
- Inkonsistenzbehandlung
- Argumentdefinition
- Attackdefinition
- Rekonstruktion der Inkonsistenzbehandlung

Inkonsistenzbehandlung

- Für eine inkonsistente Menge T gilt $Mod(T) = \emptyset$ und somit $Cn(T) = \mathcal{F}$.

Was können wir tun?

Inkonsistenzbehandlung

- Für eine inkonsistente Menge T gilt $Mod(T) = \emptyset$ und somit $Cn(T) = \mathcal{F}$.

Was können wir tun?

1. Benutze eine Logik $\mathcal{L}'$ mit schwächerem Modellbegriff, d.h. $Mod(T) \subseteq Mod'(T)$. (mehr Modelle)

Inkonsistenzbehandlung

- Für eine inkonsistente Menge T gilt $Mod(T) = \varnothing$ und somit $Cn(T) = \mathcal{F}$.

Was können wir tun?

1. Benutze eine Logik $\mathcal{L}'$ mit schwächerem Modellbegriff, d.h. $Mod(T) \subseteq Mod'(T)$. (mehr Modelle)
2. Einschränkung auf bestimmte konsistente TM von T. Falls $S \subseteq T$, dann $Mod(T) \subseteq Mod(S)$. (auch mehr Modelle)

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \rightarrow b, b \rightarrow \neg a, c, c \rightarrow \neg b\}$. Bestimme MC(T)?

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$. Bestimme MC(T)?

$$MC(T) = \{E_1, E_2, E_3, E_4\}$$

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \rightarrow b, b \rightarrow \neg a, c, c \rightarrow \neg b\}$. Bestimme MC(T)?

$$MC(T) = \{E_1, E_2, E_3, E_4\}$$

- $E_1 = \{a, a \rightarrow b, c\}$

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \rightarrow b, b \rightarrow \neg a, c, c \rightarrow \neg b\}$. Bestimme MC(T)?

$$MC(T) = \{E_1, E_2, E_3, E_4\}$$

- $E_1 = \{a, a \rightarrow b, c\}$
- $E_2 = \{a, a \rightarrow b, c \rightarrow \neg b\}$

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$. Bestimme MC(T)?

$$MC(T) = \{E_1, E_2, E_3, E_4\}$$

- $E_1 = \{a, a \to b, c\}$
- $E_2 = \{a, a \to b, c \to \neg b\}$
- $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$

Maximal Konsistente Teilmengen

Definition

Gegeben eine Menge T. Wir definieren

1. $C(T) = \{S \mid S \subseteq T, Mod(S) \neq \emptyset\}$ (konsistente TM)
2. $MC(T) = \max_{\subseteq} C(T)$ (max. konsistente TM)

Bsp.: $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$. Bestimme MC(T)?

$$MC(T) = \{E_1, E_2, E_3, E_4\}$$

- $E_1 = \{a, a \to b, c\}$
- $E_2 = \{a, a \to b, c \to \neg b\}$
- $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
- $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$

Konsequenzrelationen

Definition

Sei T eine Theorie und ϕ eine Formel. Wir definieren:

1. $T \vDash_{MC} \phi$ gdw. $\bigcap MC(T) \vDash \phi$
2. $T \vDash_{MC'} \phi$ gdw. für alle $S \in MC(T)$: $S \vDash \phi$.

Konsequenzrelationen

Definition

Sei T eine Theorie und ϕ eine Formel. Wir definieren:

1. $T \vDash_{MC} \phi$ gdw. $\bigcap MC(T) \vDash \phi$
2. $T \vDash_{MC'} \phi$ gdw. für alle $S \in MC(T)$: $S \vDash \phi$.

$T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$, $MC(T) = \{E_1, E_2, E_3, E_4\}$

- $E_1 = \{a, a \to b, c\}$
- $E_2 = \{a, a \to b, c \to \neg b\}$
- $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
- $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$

Existieren ϕ, ψ mit $T \vDash_{MC} \phi$ bzw. $T \vDash_{MC'} \psi$?

Konsequenzrelationen

Definition

Sei T eine Theorie und ϕ eine Formel. Wir definieren:

1. $T \vDash_{MC} \phi$ gdw. $\bigcap MC(T) \vDash \phi$
2. $T \vDash_{MC'} \phi$ gdw. für alle $S \in MC(T)$: $S \vDash \phi$.

$T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$, $MC(T) = \{E_1, E_2, E_3, E_4\}$

- $E_1 = \{a, a \to b, c\}$
- $E_2 = \{a, a \to b, c \to \neg b\}$
- $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
- $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$

Existieren ϕ, ψ mit $T \vDash_{MC} \phi$ bzw. $T \vDash_{MC'} \psi$?

Es gilt: $T \vDash_{MC} \phi \;\Rightarrow\; T \vDash_{MC'} \phi \;\Rightarrow\; T \vDash \phi$ (Übung 5)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, I) mit

1. $S \subseteq T$ (Teilmenge)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \perp$ (Konsistenz)
4. $S \models l$ (Folgerung)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Wir setzen $claim(A) = l$ und $support(A) = S$.

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Wir setzen $\mathit{claim}(A) = l$ und $\mathit{support}(A) = S$.

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Wir setzen $claim(A) = l$ und $support(A) = S$.

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$$A_1 = (\{a, a \to b\}, b)$$

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Wir setzen $claim(A) = l$ und $support(A) = S$.

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$$A_1 = (\{a, a \to b\}, b), \;\; A_2 = (\{c, c \to \neg b, a \to b\}, \neg a)$$

Argumentkonstruktion

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Ein T-Argument A ist ein Paar (S, l) mit

1. $S \subseteq T$ (Teilmenge)
2. $l \in \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$ (Literal)
3. $S \not\models \bot$ (Konsistenz)
4. $S \models l$ (Folgerung)
5. Es gibt kein $S' \subset S : S' \models l$ (Minimalität)

Wir setzen $\mathit{claim}(A) = l$ und $\mathit{support}(A) = S$.

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$. Sind nachfolgende Paare Argumente?

- $A_3 = (\{a, c \to \neg b\}, a)$
- $A_4 = (\{a, b \to \neg a\}, \neg b)$
- $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \equiv \neg \bigwedge S$,

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $claim(A) \equiv \neg claim(B)$.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Undercut von B, dann A Gegenargument von B.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Undercut von B, dann A Gegenargument von B.

Proof.

Es gilt $\mathit{claim}(A) \equiv \neg \bigwedge S$ gdw. $\mathit{claim}(A) \models \neg \bigwedge S$ und $\neg \bigwedge S \models \mathit{claim}(A)$. $\square$

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann A Gegenargument von B.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \vDash \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann A Gegenargument von B.

Proof.

Per Definition gilt $\mathit{support}(B) \vDash \mathit{claim}(B)$.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann A Gegenargument von B.

Proof.

Per Definition gilt $\mathit{support}(B) \models \mathit{claim}(B)$. Somit $\neg \mathit{claim}(B) \models \neg \bigwedge \mathit{support}(B)$.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann A Gegenargument von B.

Proof.

Per Definition gilt $\mathit{support}(B) \models \mathit{claim}(B)$. Somit $\neg \mathit{claim}(B) \models \neg \bigwedge \mathit{support}(B)$. Da $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$ angenommen, folgt $\mathit{claim}(A) \models \neg \bigwedge \mathit{support}(B)$. $\square$

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Unabhängigkeit Rebuttal/Undercut: Falls A Rebuttal von B, dann nicht zwangsweise A Undercut von B.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Unabhängigkeit Rebuttal/Undercut: Falls A Rebuttal von B, dann nicht zwangsweise A Undercut von B.

Bsp.: $A = (\{\neg b\}, \neg b)$, $B = (\{a, a \to b\}, b)$.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq support(B)$ existiert mit $claim(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $claim(A) \equiv \neg claim(B)$.

Unabhängigkeit Rebuttal/Undercut: Falls A Undercut von B, dann nicht zwangsweise A Rebuttal von B.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \vDash \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Unabhängigkeit Rebuttal/Undercut: Falls A Undercut von B, dann nicht zwangsweise A Rebuttal von B.

Bsp.: $A = (\{\neg b\}, \neg b)$, $B = (\{b, b \rightarrow a\}, a)$.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann B Rebuttal von A.

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Proposition

Wenn A Rebuttal von B, dann B Rebuttal von A.

Proof.

Es gilt $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$ gdw. $\neg \mathit{claim}(A) \equiv \mathit{claim}(B)$. □

Argumente und Gegenargumente

Definition

Gegeben zwei Argumente A und B. A heißt

- *Gegenargument* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \models \neg \bigwedge S$,
- *Undercut* von B, falls ein $S \subseteq \mathit{support}(B)$ existiert mit $\mathit{claim}(A) \equiv \neg \bigwedge S$, bzw.
- *Rebuttal* von B, falls $\mathit{claim}(A) \equiv \neg \mathit{claim}(B)$.

Ist Undercut auch symmetrisch? (Übung 5)

Der Assozierte Graph

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Das assozierte AF $F_T = (A, R)$ ist gegeben mit

- $A = \{A_i \mid A_i \text{ ist } T\text{-Argument}\}$
- $R = \{(A_i, A_j) \mid A_i \text{ ist Gegenargument von } A_j\}$

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

Der Assozierte Graph

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Das assozierte AF $F_T = (A, R)$ ist gegeben mit

- $A = \{A_i \mid A_i \text{ ist } T\text{-Argument}\}$
- $R = \{(A_i, A_j) \mid A_i \text{ ist Gegenargument von } A_j\}$

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,

Der Assozierte Graph

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Das assozierte AF $F_T = (A, R)$ ist gegeben mit

- $A = \{A_i \mid A_i \text{ ist } T\text{-Argument}\}$
- $R = \{(A_i, A_j) \mid A_i \text{ ist Gegenargument von } A_j\}$

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
$A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$

Der Assozierte Graph

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$.
Das assozierte AF $F_T = (A, R)$ ist gegeben mit

- $A = \{A_i \mid A_i \text{ ist } T\text{-Argument}\}$
- $R = \{(A_i, A_j) \mid A_i \text{ ist Gegenargument von } A_j\}$

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,

$A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,

$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$

Der Assozierte Graph

Definition

Gegeben eine endliche Menge T von Formeln über Alphabet $\mathcal{A}$. Das assozierte AF $F_T = (A, R)$ ist gegeben mit

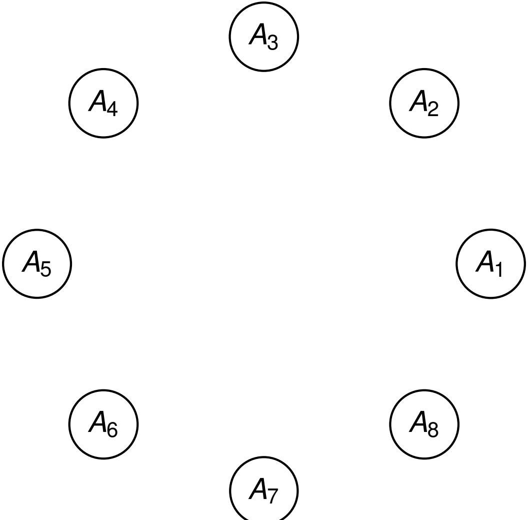
- $A = \{A_i \mid A_i \text{ ist } T\text{-Argument}\}$
- $R = \{(A_i, A_j) \mid A_i \text{ ist Gegenargument von } A_j\}$

Bsp.: Sei $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$.

$$A_1 = (\{a\}, a),\ \ A_2 = (\{c\}, c),$$
$$A_3 = (\{a, a \to b\}, b),\ \ A_4 = (\{a, b \to \neg a\}, \neg b),$$
$$A_5 = (\{a \to b, b \to \neg a\}, \neg a),\ \ A_6 = (\{c, c \to \neg b\}, \neg b),$$
$$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c),\ \ A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$$

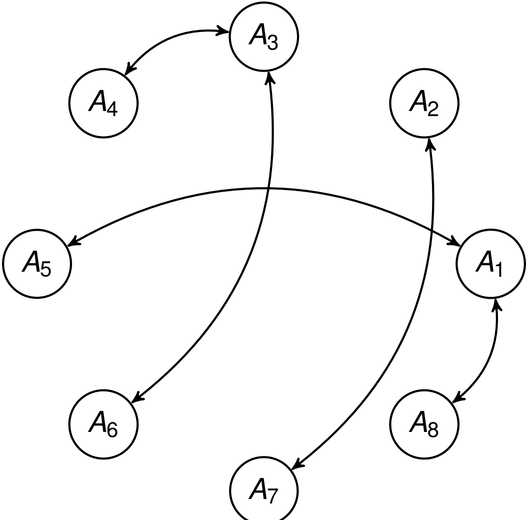
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \rightarrow b\}, b)$, $A_4 = (\{a, b \rightarrow \neg a\}, \neg b)$,
$A_5 = (\{a \rightarrow b, b \rightarrow \neg a\}, \neg a)$, $A_6 = (\{c, c \rightarrow \neg b\}, \neg b)$,
$A_7 = (\{a, a \rightarrow b, c \rightarrow \neg b\}, \neg c)$, $A_8 = (\{a \rightarrow b, c, c \rightarrow \neg b\}, \neg a)$



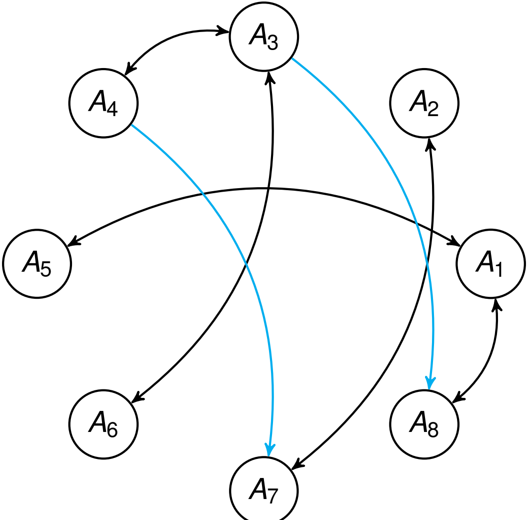
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,
$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$,
$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$



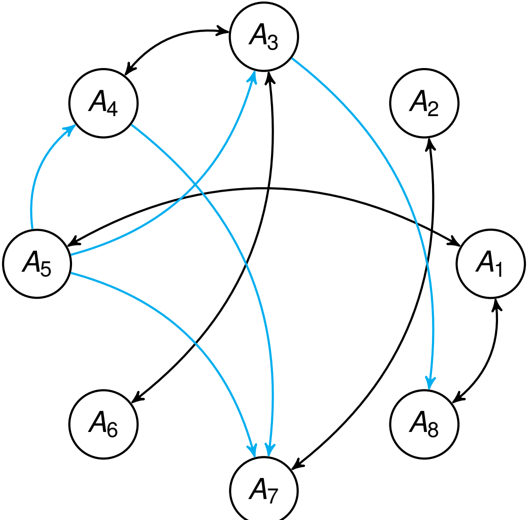
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,
$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$,
$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$



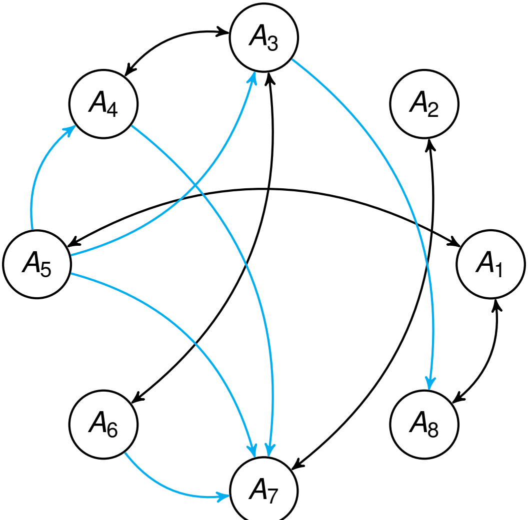
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,
$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$,
$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$



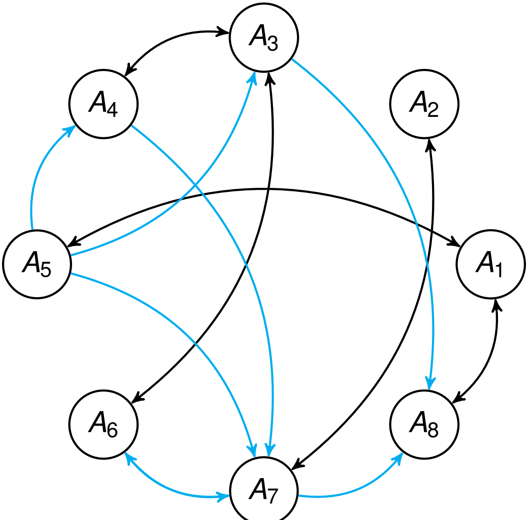
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,
$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$,
$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$



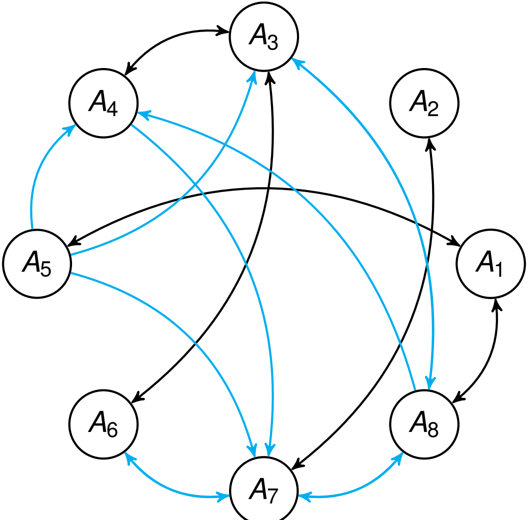
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$,
$A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$,
$A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$



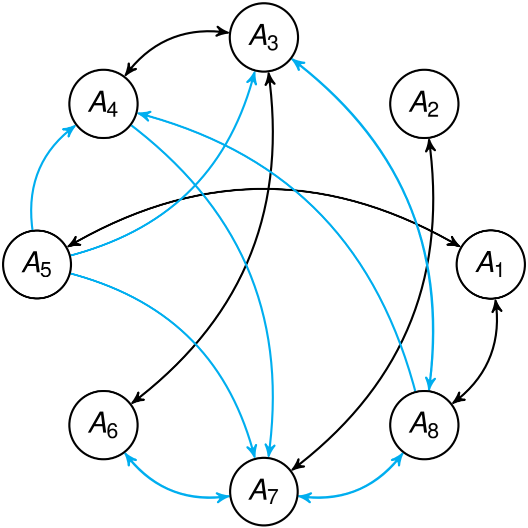
Der Assozierte Graph

$A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$, $A_3 = (\{a, a \rightarrow b\}, b)$, $A_4 = (\{a, b \rightarrow \neg a\}, \neg b)$,
$A_5 = (\{a \rightarrow b, b \rightarrow \neg a\}, \neg a)$, $A_6 = (\{c, c \rightarrow \neg b\}, \neg b)$,
$A_7 = (\{a, a \rightarrow b, c \rightarrow \neg b\}, \neg c)$, $A_8 = (\{a \rightarrow b, c, c \rightarrow \neg b\}, \neg a)$



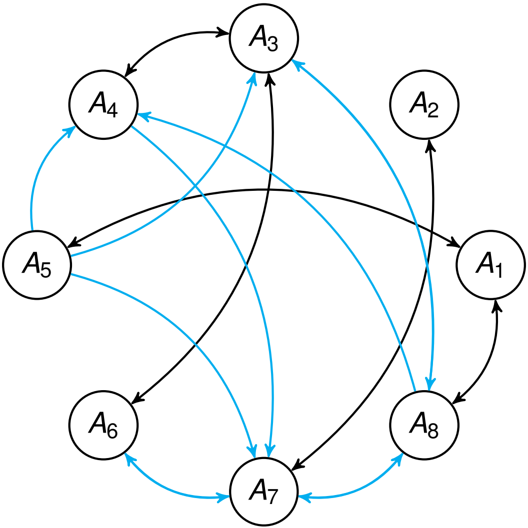
Stabile Extensionen

$$stb(F) = \ ?$$



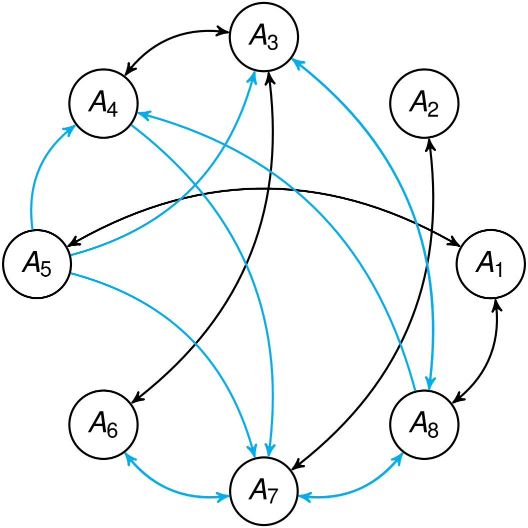
Stabile Extensionen

$$stb(F) = \{\{A_5, A_2, A_6, A_8\}, \ldots\}$$



Stabile Extensionen

$$stb(F) = \{\{A_5, A_2, A_6, A_8\}, \{A_1, A_2, A_3\}, \{A_1, A_2, A_4, A_6\}, \{A_1, A_7, A_3\}\}$$



Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \dots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \dots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Was haben wir alles berechnet?

- $T = \{a, a \to b, b \to \neg a, c, c \to \neg b\}$
- $MC(T) = \{E_1, E_2, E_3, E_4\}$
  - $E_1 = \{a, a \to b, c\}$
  - $E_2 = \{a, a \to b, c \to \neg b\}$
  - $E_3 = \{a, b \to \neg a, c, c \to \neg b\}$
  - $E_4 = \{a \to b, b \to \neg a, c, c \to \neg b\}$
- $stb(F) = \{S_1, S_2, S_3, S_4\}$
  - $S_1 = \{A_5, A_2, A_6, A_8\}$
  - $S_2 = \{A_1, A_2, A_3\}$
  - $S_3 = \{A_1, A_2, A_4, A_6\}$
  - $S_4 = \{A_1, A_7, A_3\}$
- $A(F) = \{A_1, A_2, \ldots, A_8\}$
  - $A_1 = (\{a\}, a)$, $A_2 = (\{c\}, c)$,
  - $A_3 = (\{a, a \to b\}, b)$, $A_4 = (\{a, b \to \neg a\}, \neg b)$
  - $A_5 = (\{a \to b, b \to \neg a\}, \neg a)$, $A_6 = (\{c, c \to \neg b\}, \neg b)$
  - $A_7 = (\{a, a \to b, c \to \neg b\}, \neg c)$, $A_8 = (\{a \to b, c, c \to \neg b\}, \neg a)$

Stellschrauben

- Theorie T: beliebige Formeln, Hornformeln, 2-KNF, . . .

Stellschrauben

- Theorie T: beliebige Formeln, Hornformeln, 2-KNF, . . .
- T-argument: claim Literale, beliebige Formel,. . .
  support Konsistenz, Minimalität . . .

Stellschrauben

- Theorie T: beliebige Formeln, Hornformeln, 2-KNF, ...
- T-argument: claim Literale, beliebige Formel,...
  support Konsistenz, Minimalität ...
- Attackdefinition: Gegenargument, Undercut, Rebuttal, ...

Stellschrauben

- Theorie T: beliebige Formeln, Hornformeln, 2-KNF, ...
- T-argument: claim Literale, beliebige Formel,...
  support Konsistenz, Minimalität ...
- Attackdefinition: Gegenargument, Undercut, Rebuttal, ...
- benutzte Semantik: stable, preferred, complete, ...

Stellschrauben

- Theorie T: beliebige Formeln, Hornformeln, 2-KNF, ...
- T-argument: claim Literale, beliebige Formel,...
  support Konsistenz, Minimalität ...
- Attackdefinition: Gegenargument, Undercut, Rebuttal, ...
- benutzte Semantik: stable, preferred, complete, ...

$\Rightarrow$ Unzählige Resultate bzgl. verschiedener Kombinationen
(... heute nur ein kleiner Einblick)

Vorlesung “Formale Argumentation”

10. Structured Argumentation – Inkonsistenzbehandlung

Ringo Baumann
Professur für Formale Argumentation
und Logisches Schließen

13. Juni 2024
Leipzig