On the Instantiation of KBs in Abstract AFs

Adam Wyner
University of Aberdeen

Trevor Bench-Capon and Paul Dunne
University of Liverpool

September 16, 2013
CLIMA XIV
La Coruna, Spain

Overview

- Motivating issues in instantiated AFs:
  - What are ''arguments'' and their relationships in AF terms?
  - How to address the Rationality Postulates?
  - How to simplify?
- Main idea:
  - translate KBs directly into AFs; no mediating 'argument' constructions.
  - 2 steps in analysis rather than 3.
- Some details of proposal and worked example.
- Three senses of ''argument'' (auxiliary issue).
- An ADF approach by Strass (2013) (comments).

Bit of Context

- Dungian arguments are abstract (*nodes*) in attack relations (*arcs*)
- Proposals to construct abstract argument from a KB - instantiated argumentation. ASPIC+, Logic-based, etc.
- Use 3 steps.
- Allow arguments with subarguments; unclarity about attacking subarguments; overgeneration (arguments and attacks); unexplanatory.
- Issues about Rationality Postulates (ASPIC+):
  - Closure; Direct consistency; Indirect consistency.
  - Auxiliary definitions; restricted rebut (rebuttals only on conclusions of defeasible rules, not strict rules).
- Unclear about *senses* of argument (Wyner et al. (2008, 2009)):
  - Argument: a single reasoning step from premises to claim.
  - Case: a train of arguments reasoning to a claim.
  - Debate: cases for and against a claim.

AFs that Wear the Logic on the Sleeves

- Conservative Dungian analysis: arcs and nodes; standard semantics; no auxiliary contributions to success of attacks (preferences).
- Address *Rationality Postulates* (consistency built in; closure by integrity constraint).
- Simplify steps: no need to pack/unpack 'arguments'; facilitates addition/removal/change in theories and AFs.
- Relate argument extensions to models of a theory.
- Account for partiality, i.e. get extensions of theories where there are premises that are not asserted.
- Clarify *senses* of argument (Wyner et al. (2008)).

Three Steps v. Two Steps

Knowledge Base → Argumentation Framework → Extensions of Arguments → Extensions of Conclusions

Step 1: construct arguments and attacks

Step 2: identify sets of accepted arguments

Step 3: identify sets of accepted conclusions

Caminada and Wu (2011)
'arguments' as complex premise-claim constructions

Knowledge Base → Argumentation Framework → Extensions of Arguments

Step 1: construct arguments and attacks

Step 2: identify sets of accepted arguments

Wyner, Bench-Capon, and Dunne (2013)
'arguments' as nodes in graph correlating to KB.

Different notions of 'argument' at work

Given a Theory Base

Definition 3. *A* Theory Base, $\mathcal{T}$, *comprises a pair* $(\mathcal{L}, \mathcal{R})$ *in which*

$$\mathcal{L} = \{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\}$$

is a set of literals over a set of propositional variables $\{x_1, \ldots, x_n\}$*. We use* y_i *to denote an arbitrary literal from* $\{x_i, \neg x_i\}$*.*

We have a set of proper names *of rules* $\{r_1, r_2, \ldots, r_n\}$*. Rules are either strict* ($r \in \mathcal{R}_{str}$) *or defeasible* ($r \in \mathcal{R}_{dfs}$)*, and* $\mathcal{R}_{str} \cap \mathcal{R}_{dfs} = \emptyset$*.* $\mathcal{R} = \mathcal{R}_{str} \cup \mathcal{R}_{dfs}$ *where*

$$\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$$

in which $r \in \mathcal{R}$ *has a body,* $bd(r) \subseteq \mathcal{L}$*, and a head,* $hd(r) \in \mathcal{L}$*.*

Classical negation.
Heads can be positive or negative (not *Horn* clauses).

Derive Nodes (L) for an AF

Definition 5. *Let $\mathcal{T} = (\mathcal{L}, \mathcal{R})$ be a Theory Base with*

$$\begin{aligned} \mathcal{L} &= \{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\} \\ \mathcal{R} &= \mathcal{R}_{str} \cup \mathcal{R}_{dfs} \end{aligned}$$

The derived framework *from $\mathcal{T}$, is the* AF, $\langle \mathcal{L}^A_\mathcal{T}, \mathcal{R}^A_\mathcal{T} \rangle$ *in which,*

$$\begin{aligned} \mathcal{L}^A_\mathcal{T} = \; & \{\, \mathsf{x}, \neg \mathsf{x} \; : x, \neg x \in \mathcal{L}\} \\ & \cup \{\, r : bd(r) \rightarrow hd(r) \; : \; r \in \mathcal{R}_{str}\} \\ & \cup \{\, r : bd(r) \Rightarrow hd(r) \; : \; r \in \mathcal{R}_{dfs}\} \end{aligned}$$

Furthermore,

$$\begin{aligned} & \forall \mathsf{x} \in \mathcal{L}^A_\mathcal{T}, x \in \mathcal{L}, \; and \\ & \forall \mathsf{r} \in \mathcal{L}^A_\mathcal{T}, r \in \mathcal{R} \end{aligned}$$

Have positive and negative literals.
As in the Theory Base, literals occur once in the AF.
Use elements of Theory Base as labels on AF nodes

Derive Attacks (R) for an AF

Definition 6. *In the* AF $\langle \mathcal{L}^A_T, \mathcal{R}^A_T \rangle$, $\mathcal{R}^A_T = \mathcal{R}^A_{ll} \cup \mathcal{R}^A_{lr} \cup \mathcal{R}^A_{rl}$ *where:*

$$\begin{aligned}
\mathcal{R}^A_{ll} &= \{\langle \mathbf{y}_i, \neg \mathbf{y}_i \rangle,\ \langle \neg \mathbf{y}_i, \mathbf{y}_i \rangle\ :\ 1 \leq i \leq n \\
&\quad \textit{and } \mathbf{y}_i, \neg \mathbf{y}_i \in \mathcal{L}^A_T\} \\
\mathcal{R}^A_{lr} &= \{\langle \neg \mathbf{y}_i, \mathbf{r}_j \rangle\ :\ y_i \in bd(r_j) \textit{ and } \neg \mathbf{y}_i, \mathbf{r}_j \in \mathcal{L}^A_T\} \\
&\quad \cup \{\langle \neg \mathbf{y}_i, \mathbf{r}_j \rangle\ :\ r_j \in \mathcal{R}_{dfs} \textit{ and } hd(r_j) = y_i \\
&\quad \textit{and } \neg \mathbf{y}_i \in \mathcal{L}^A_T\} \\
\mathcal{R}^A_{rl} &= \{\langle \mathbf{r}_j, \neg \mathbf{y}_i \rangle\ :\ hd(r_j) = y_i \textit{ and } \neg \mathbf{y}_i, \mathbf{r}_j \in \mathcal{L}^A_T\}
\end{aligned}$$

Positive and negative literals attack one another.
Negative literal of body (negation of premise) attacks rule.
Negative literal of head of defeasible rule (negation of conclusion) attacks rule.
Rule attacks negative literal of head (negation of conclusion).

Dungian Semantics with an Integrity Constraint

Standard Dungian Semantics.
As positive and negative literals attack one another,
no admissible extension can have both.

-- *Direct and Indirect Consistency Built In* --

Constraint 5 *Consider: a, an admissible set of the derived* AF $\langle \mathcal{L}^A_T, \mathcal{R}^A_T \rangle$; $\mathcal{A}$, *the set of admissible sets a; and* $\mathcal{R}_{str} \subseteq \mathcal{L}^A_T$. *For every* $r \in \mathcal{R}_{str}$ *and every* $a \in \mathcal{A}$, *if* $r \in a$ *and every* $bd(r) \in a$, *then* $hd(r) \in a$.

Definition 7. *An admissible set of the derived* AF $\langle \mathcal{L}^A_T, \mathcal{R}^A_T \rangle$ *is a* Well-formed Admissible Set *(WFAS) iff it satisfies Constraint 5.*

Extensions are not homogeneous – contain literals and rules.

-- *Closure of Strict Rules* --

A Strict Rule – Theory Base and AF

Example 1. Let $\mathcal{T}_1$ be the pair with $(\mathcal{L}_1, \mathcal{R}_1)$, where

$$\begin{aligned} \mathcal{L}_1 &= \{x_1, x_2\} \cup \{\neg x_1, \neg x_2\} \\ \mathcal{R}_1 &= \{r_1\},\ \textit{where}\ r_1\ \textit{has rule name}\ r_1 : x_1 \rightarrow x_2 \end{aligned}$$

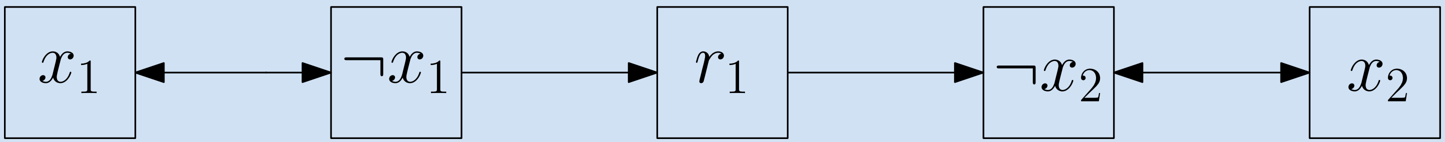
The *derived framework* from $\mathcal{T}_1$ is $\langle \mathcal{L}^A_{\mathcal{T}_1}, \mathcal{R}^A_{\mathcal{T}_1} \rangle$ in which,

$$\mathcal{L}^A_{\mathcal{T}_1} = \{x_1, x_2\} \cup \{\neg x_1, \neg x_2\} \cup \{r_1\}$$

and in which $\mathcal{R}^A_{\mathcal{T}_1}$ comprises the union of three disjoint sets:

$$\begin{aligned} \mathcal{R}^A_{ll} &= \{\langle x_1, \neg x_1 \rangle, \langle \neg x_1, x_1 \rangle, \langle x_2, \neg x_2 \rangle, \langle \neg x_2, x_2 \rangle\} \\ \mathcal{R}^A_{lr} &= \{\langle \neg x_1, r_1 \rangle\} \\ \mathcal{R}^A_{rl} &= \{\langle r_1, \neg x_2 \rangle\} \end{aligned}$$

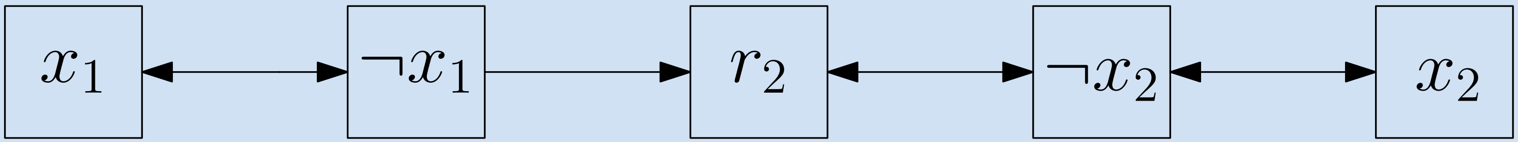
A Strict Rule – Graph and Extensions



$$\{x_1, r_1, x_2\}, \{\neg x_1, x_2\}, \{\neg x_1, \neg x_2\}$$

- Get extensions even where *we just have the rule* (unlike other approaches).
- Presence of rule in extensions is understood to mean that the rule has *applied*.
- With respect to literals, extensions correlate with models of the Theory Base.
- The 'absent' extension is also not a model, that is, {x1, -x2}.

A Defeasible Rule – Graph and Extensions



$$\{x_1, r_2, x_2\}, \{\neg x_1, x_2\}, \{\neg x_1, \neg x_2\}, \{x_1, \neg x_2\}$$

Where one has a Theory Base/AF with several strict and defeasible rules, may use the presence of rules in the extensions to decide which extension is chosen, e.g., choose the extension with the most defeasible rules. This may be construed as the most 'normative' context.

Extensions with Partial Theories

- What to do with a theory that has no assertion for a premise of a rule?
- Cannot construct an argument in such a case in ASPIC:
  - base of definition of argument: body-less rules are arguments; a rule with premises is an argument where the premises are arguments.

Definition 8. *(Argument) Suppose a* Theory Base, $\mathcal{T}$, *with strict and defeasible rules. An argument* A *is:*
$A_1, \ldots, A_n \longrightarrow \psi$ *if* $A_1, \ldots, A_n$, *with* $n \geq 0$, *are arguments such that there exists a strict rule* $\texttt{Conc}(A_1), \ldots, \texttt{Conc}(A_n) \rightarrow \psi$.
$\texttt{Prem}(A) = \texttt{Prem}(A_1) \cup \ldots \cup \texttt{Prem}(A_n)$,
$\texttt{Conc}(A) = \psi$,
$\texttt{Sub}(A) = \texttt{Sub}(A_1) \cup \ldots \cup \texttt{Sub}(A_n) \cup \{A\}$,
$\texttt{DefRules}(A) = \texttt{DefRules}(A_1) \cup \ldots \cup \texttt{DefRules}(A_n)$
$\texttt{TopRule}(A) = \texttt{Conc}(A_1), \ldots, \texttt{Conc}(A_n) \rightarrow \psi$

Extensions with Partial Theories

- Not a problem here.
- Theory Base: $r_{9:}$ -> $x_{1,}$ $r_{1:}$ x_1, x_3 -> x_2
- Extensions: $\{x_1, x_3, r_1, x_2\}, \{x_1, \neg x_3, x_2\}, \{x_1, \neg x_3, \neg x_2\}$

  (typo - should include r_9 in the sets).
- Useful in reasoning with Dungian Semantics in partial theories.

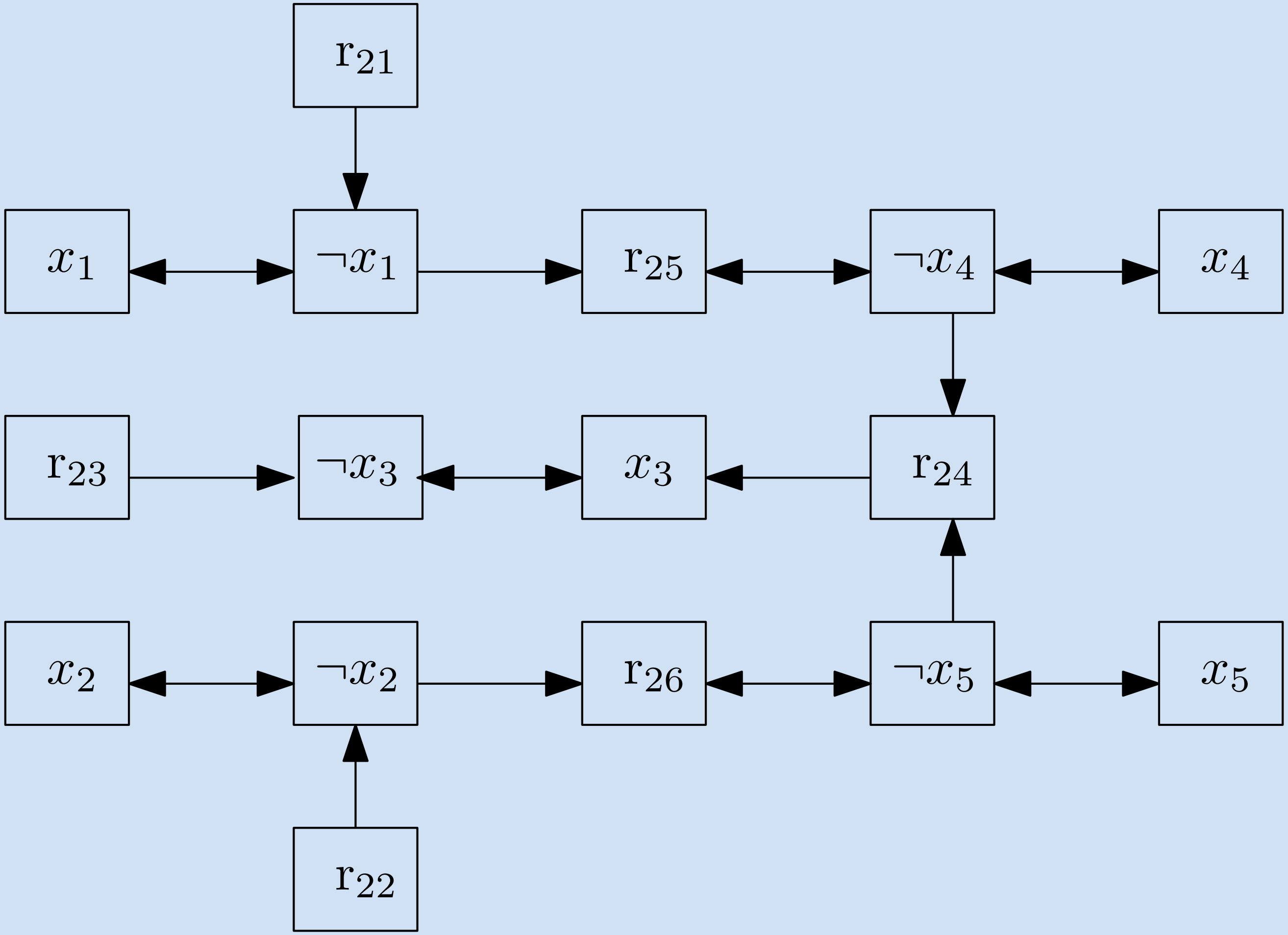
Problematic Example

Example 4. Let $\mathcal{T}_4$ be a Theory Base with the following rules:
$r_{21}: \rightarrow x_1$; $r_{22}: \rightarrow x_2$; $r_{23}: \rightarrow x_3$; $r_{24}: x_4, x_5 \rightarrow \neg x_3$; $r_{25}: x_1 \Rightarrow x_4$; $r_{26}: x_2 \Rightarrow x_5$.
We construct the following arguments:
A_1: $[[\rightarrow x_1] \Rightarrow x_4]$; A_2: $[[\rightarrow x_2] \Rightarrow x_5]$; A_3: $[\rightarrow x_3]$;
A_4: $[\rightarrow x_1]$; A_5: $[\rightarrow x_2]$;
A_6: $[[\rightarrow x_1] \Rightarrow x_4], [[\rightarrow x_2] \Rightarrow x_5] \rightarrow \neg x_3$.

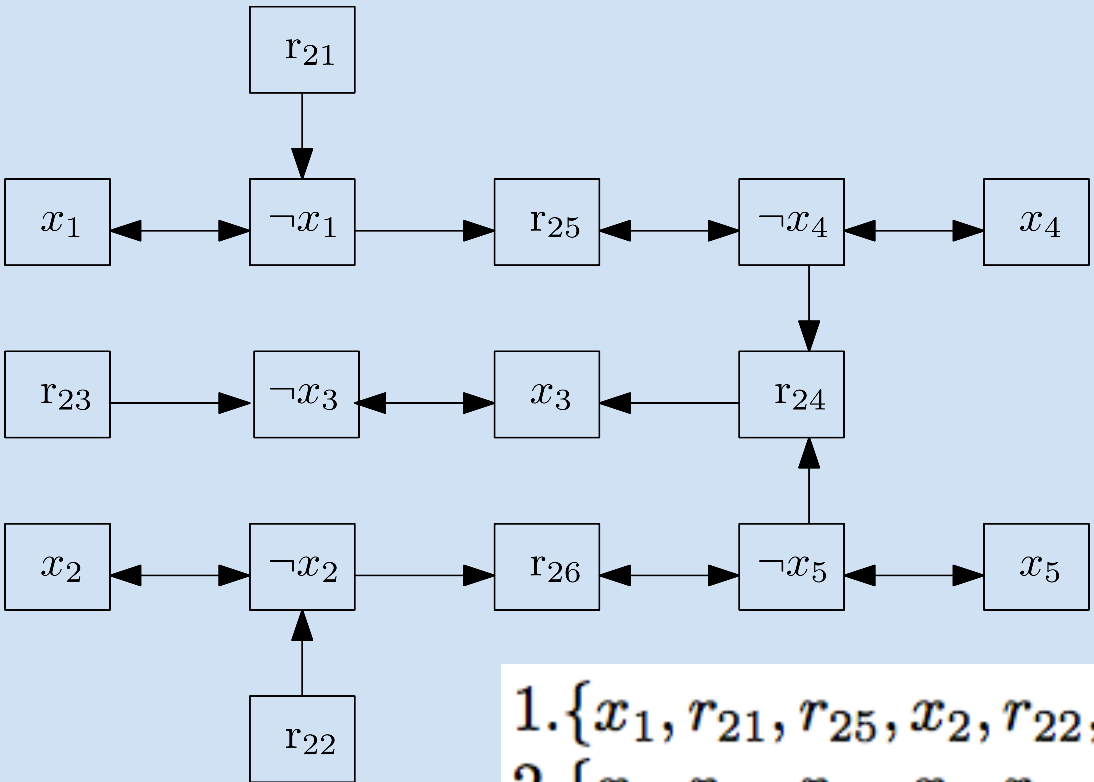
- An argument is strict if it has no defeasible subarguments.
- A strict argument can defeat a defeasible argument, but not vice versa.
- Defeat of a subargument is 'inherited' as defeat of the argument.

Problematic claim: $-x_3$ should be a justified conclusion as the conclusion of a strict rule where all the premises are justified conclusions. Yet, A_6 is defeasible, while A_3 is strict, so A_3 defeats A_6, and we do not yield $–x_3$. Does not satisfy an intuition that there should be closure of strict rules. Problematic because we have a strict conclusion from defeasible premises.

AF Derived from Theory Base



Extensions



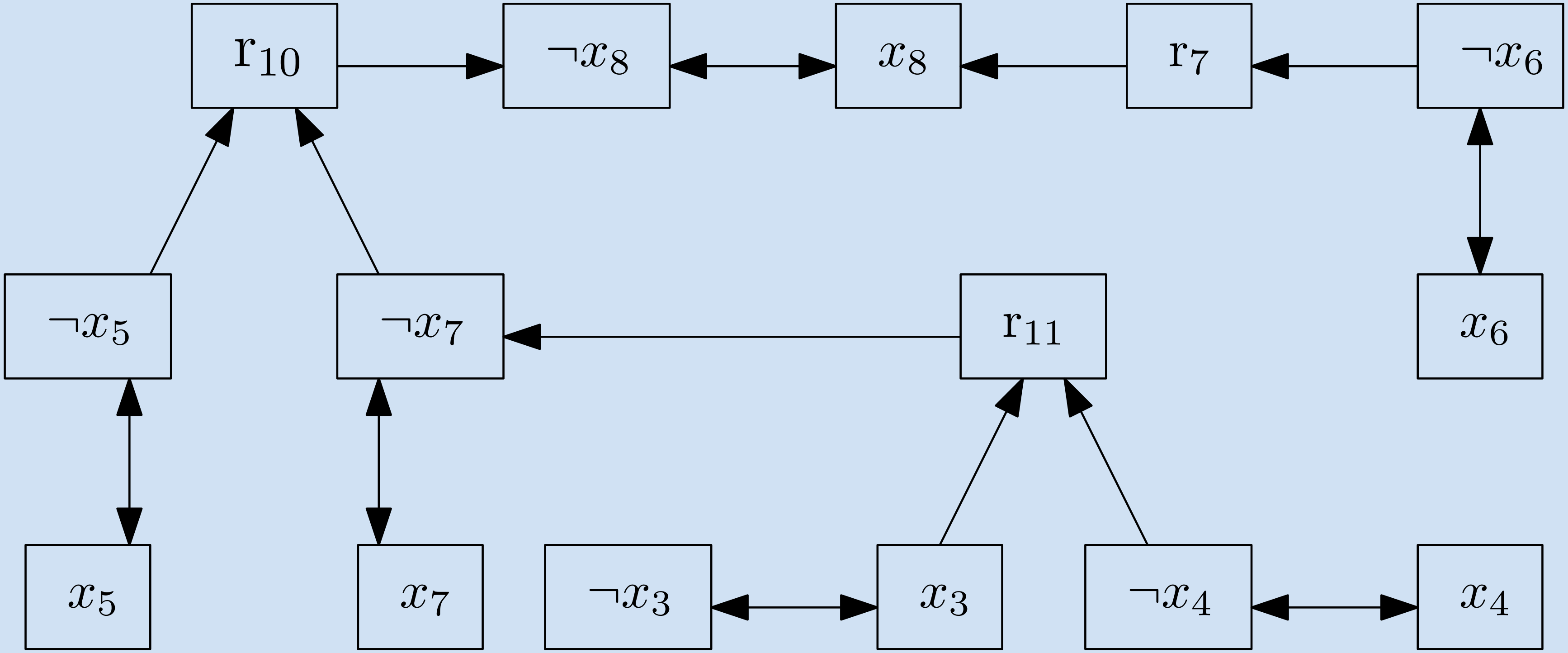
Reason from what must be in sets. Note $-x_4$ or $-x_5$.

r_{24} does not apply

1. $\{x_1, r_{21}, r_{25}, x_2, r_{22}, x_3, x_4, r_{25}, \neg x_5\}$
2. $\{x_1, r_{21}, r_{23}, x_2, r_{22}, x_3, \neg x_4, x_5, r_{26}\}$
3. $\{x_1, r_{21}, r_{23}, x_2, r_{22}, x_3, \neg x_4, \neg x_5\}$
4. $\{x_1, r_{21}, r_{23}, r_{25}, x_2, r_{22}, r_{24}, r_{26}, x_4, x_5\}$

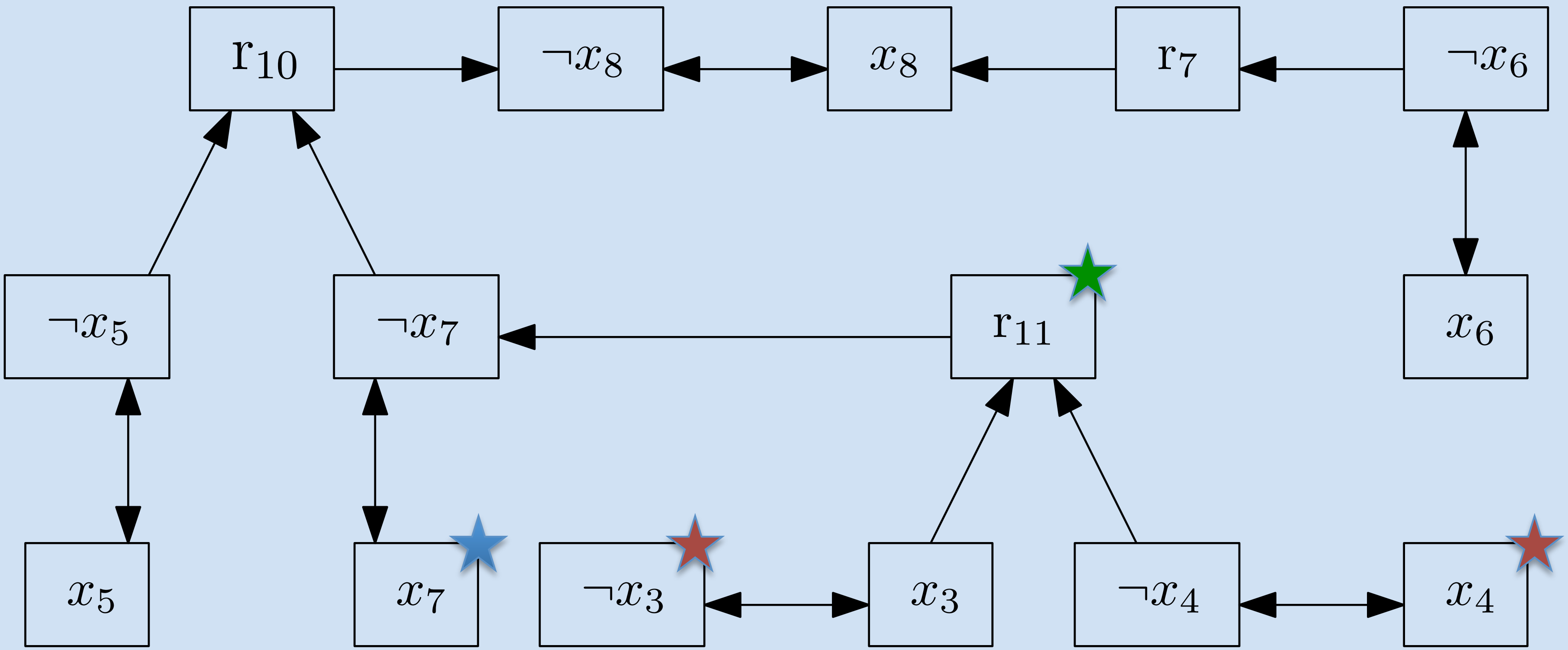
Not a Well-formed Admissible Set.

Arguments, Cases, Debates

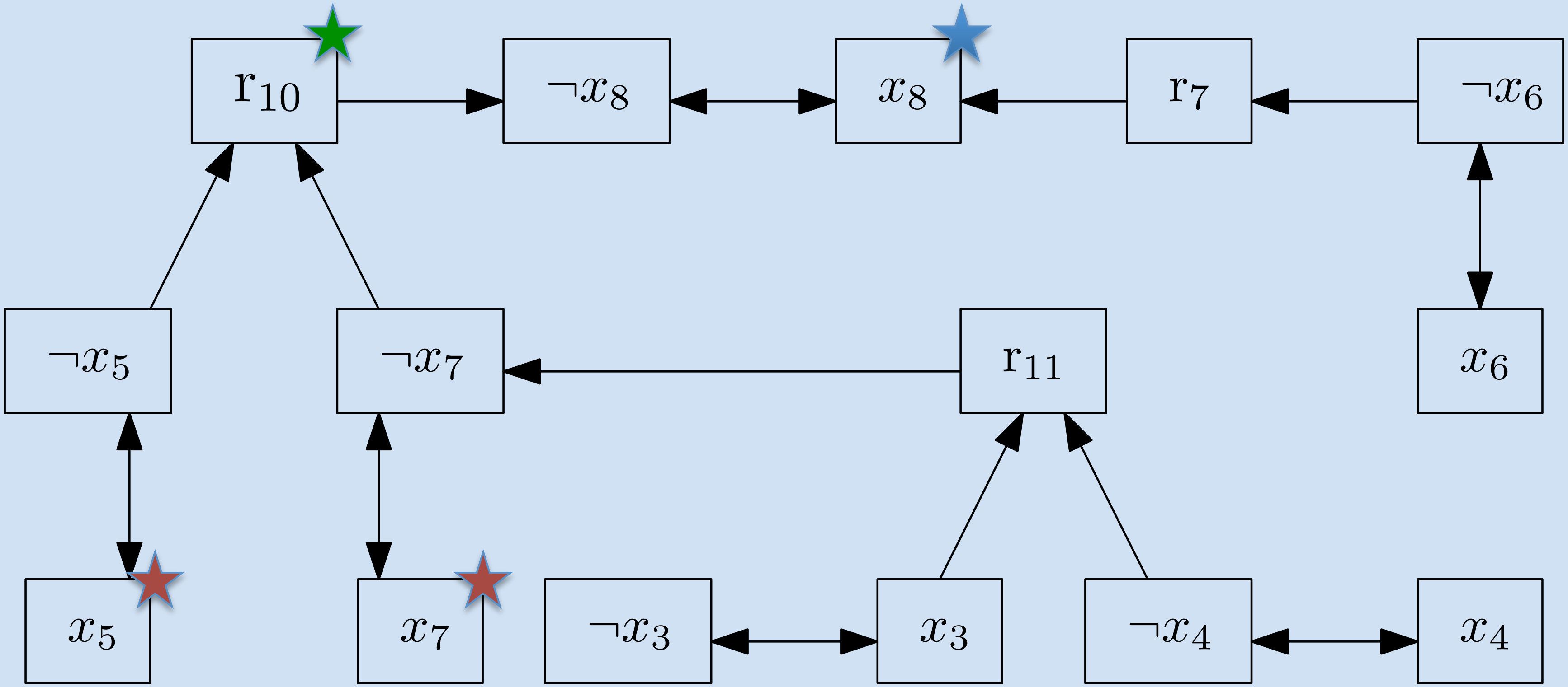


Superstructures on the graph

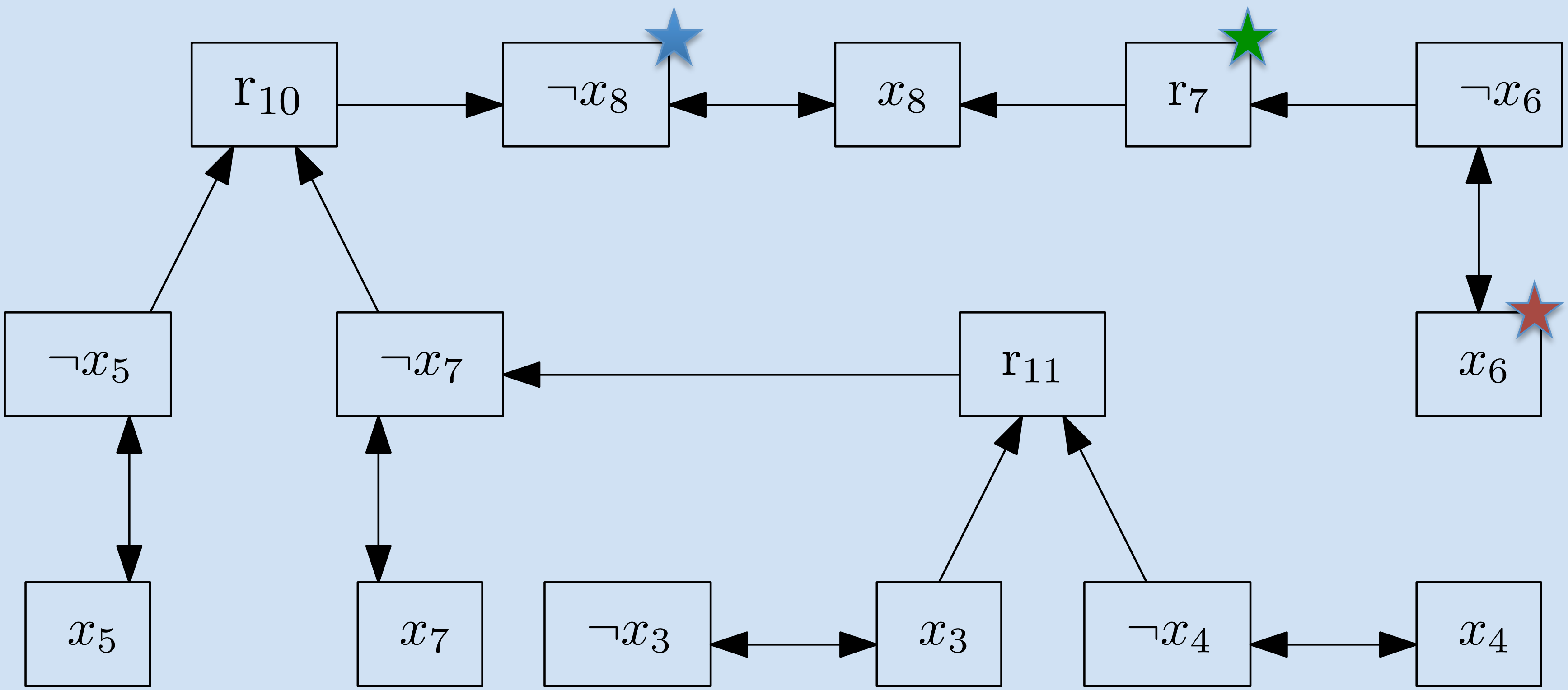
Argument 1 for x_7



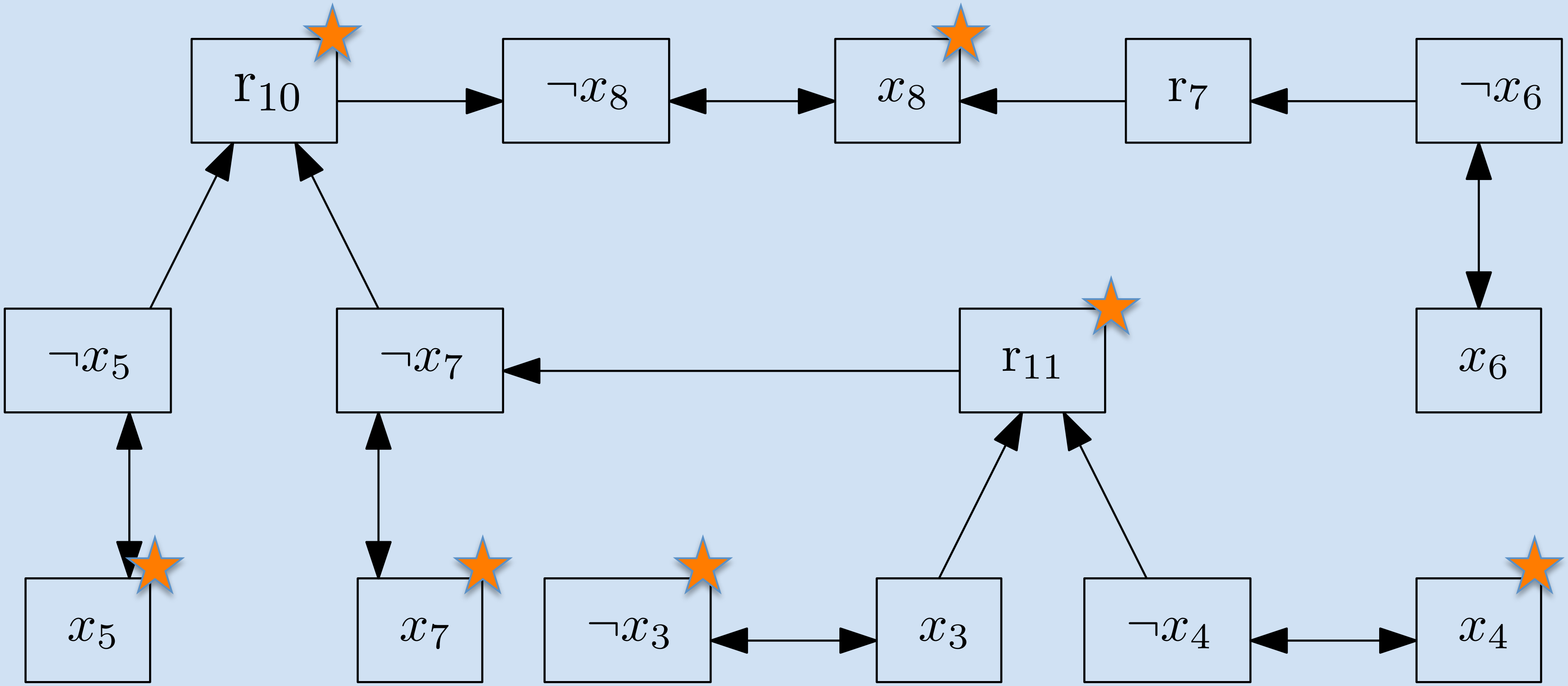
Argument 2 for x_8



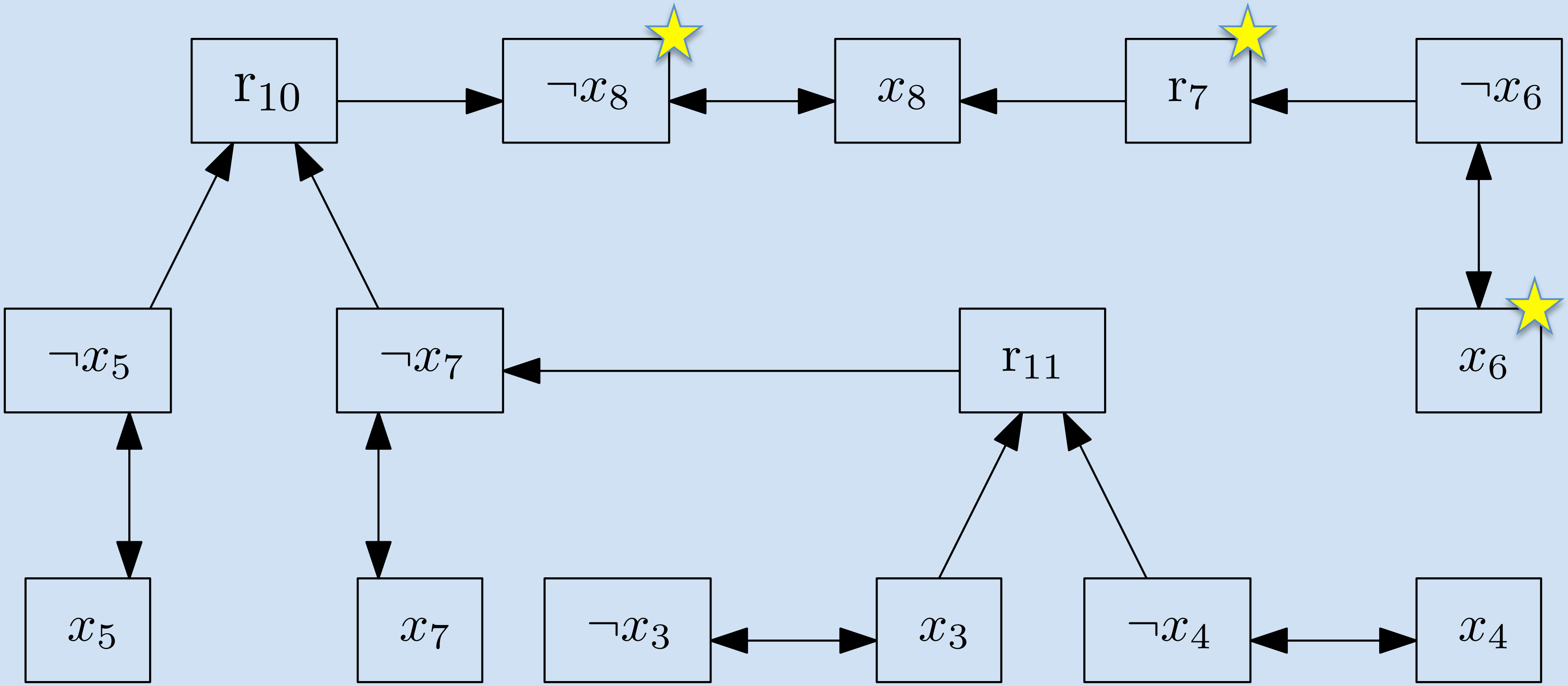
Argument 3 against x_8



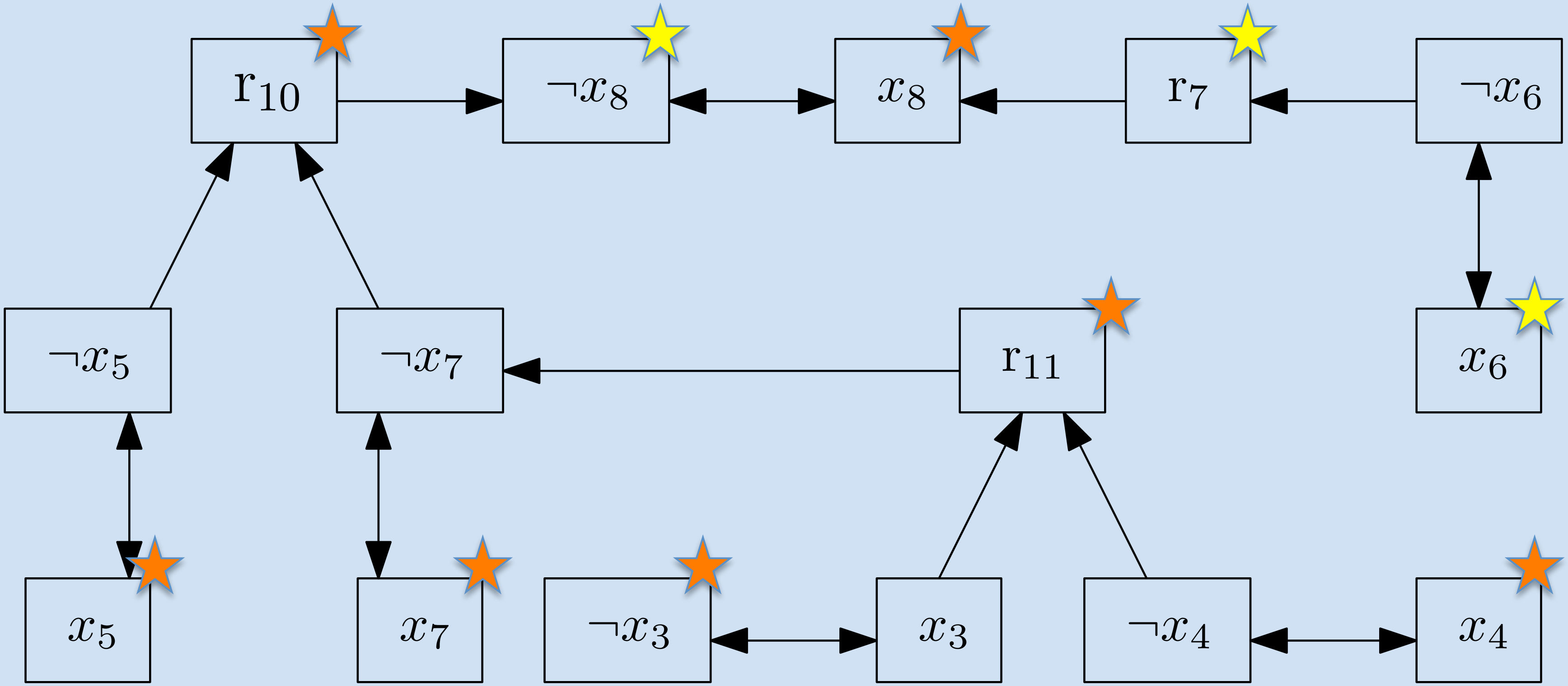
Case 1 for x_8



Case 2 against x_8



Debate for and against x_8

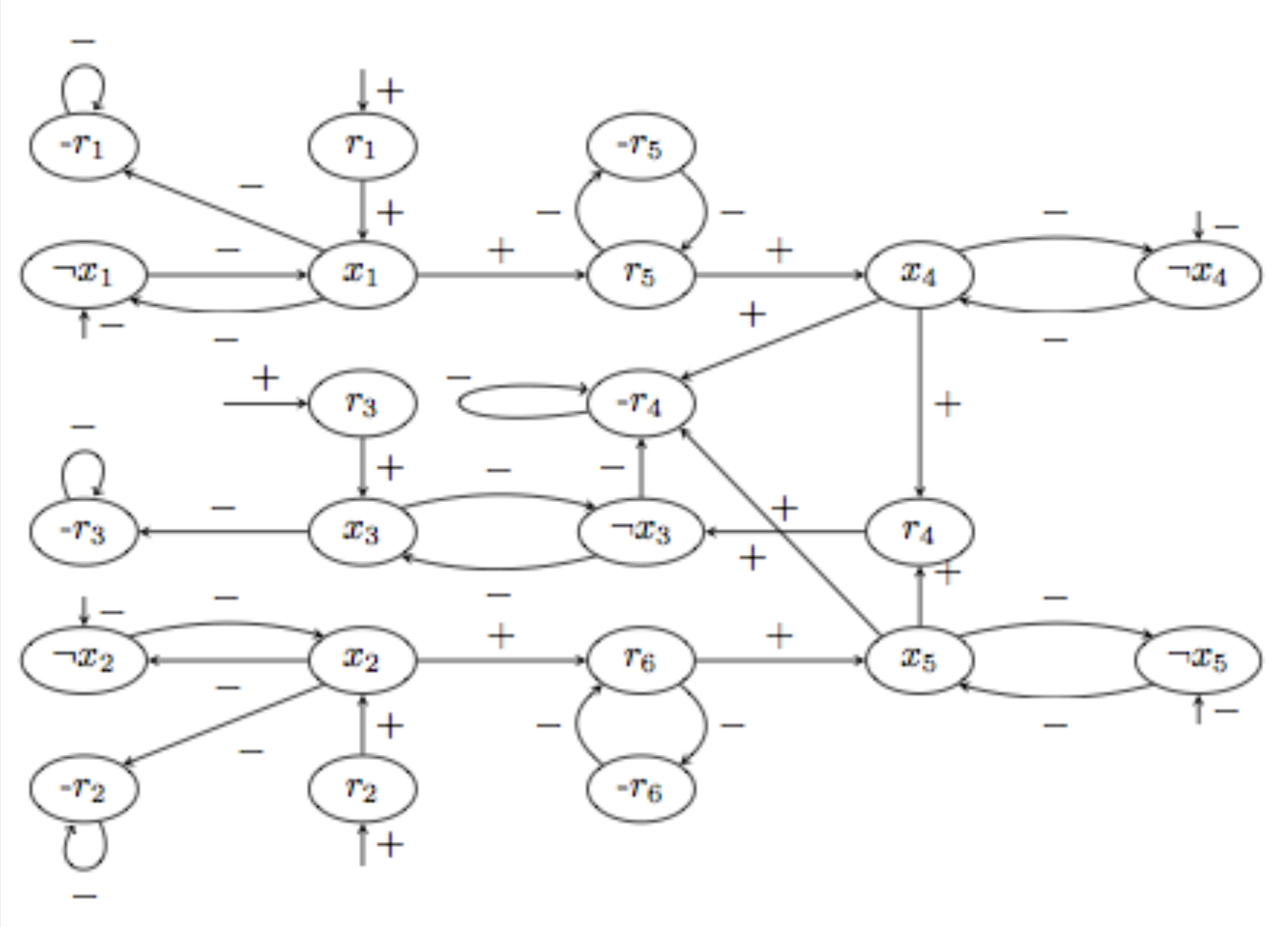


Abstract Dialectical Frameworks – Strass (2013)

- Points out that Wyner et al. (2009) allows the undesireable extension of worked example. Fixed in Wyner et al. (2013).
- Adapts the proposal to an ADF approach.
- ADF more *general* than AFs.
- ADF definitions – statements, links, and acceptance functions (determines when child is acceptable relative to parents).
- Has *attack* and *support*.
- Integrity constraint.
- Negation of literals (closed world) and of rules (inapplicable?).
- Same worked example.

Example in ADF

+ is support;
- is attack.



Acceptance Condition Functions, Then Evaluation

Example 3 (Continued). Definition 4 yields the following acceptance formulas.

$$
\begin{array}{lll}
\varphi_{x_1} = \neg[\neg x_1] \wedge [r_1] & \varphi_{x_2} = \neg[\neg x_2] \wedge [r_2] & \varphi_{x_3} = \neg[\neg x_3] \wedge [r_3] \\
\varphi_{x_4} = \neg[\neg x_4] \wedge [r_5] & \varphi_{x_5} = \neg[\neg x_5] \wedge [r_6] & \\
\varphi_{\neg x_1} = \bot \quad \varphi_{\neg x_2} = \bot & \varphi_{\neg x_3} = \neg[x_3] \wedge [r_4] & \varphi_{\neg x_4} = \bot \quad \varphi_{\neg x_5} = \bot \\
\varphi_{r_1} = \top \quad \varphi_{r_2} = \top & \varphi_{r_3} = \top & \varphi_{r_4} = [x_4] \wedge [x_5] \\
\varphi_{r_5} = [x_1] \wedge \neg[\neg x_4] \wedge \neg[-r_5] & \varphi_{r_6} = [x_2] \wedge \neg[\neg x_5] \wedge \neg[-r_6] & \\
\varphi_{-r_1} = \neg[x_1] \wedge \neg[-r_1] & \varphi_{-r_2} = \neg[x_2] \wedge \neg[-r_2] & \varphi_{-r_3} = \neg[x_3] \wedge \neg[-r_3] \\
\varphi_{-r_4} = [x_4] \wedge [x_5] \wedge \neg[\neg x_3] \wedge \neg[-r_4] & & \varphi_{-r_5} = \neg[r_5] \quad \varphi_{-r_6} = \neg[r_6]
\end{array}
$$

An *outie approach* in which successful 'attack' or 'support' is mediated by a separate component. Others – preferences and values. Very powerful and (too?) expressive.

V.

An *innie approach* in which Dungian attack is all.

Questions on ADF Approach

- While a more *general* can be an advantage, an approach should be *best fit for purpose* (explanation?)
- Supports (difference between premise and 'other reasons'?).
- Self-support, where a literal only has itself in its acceptance function (means what in classical logic?).
- Rule negation (Two negations. Could be 'ab' as in Wyner et al. (2009)).
- Represent 'senses of argument'?
- Main difference is in acceptance conditions, which have to be generated and could be used in a variety of ways to augment the graph.

Future Work

- Back to basics to reconstruct theory from ground up more systematically – models, proofs, complexity....
- Further examples.
- Add auxiliary reasoning, e.g. preferences and values.
- Compare further to ADFs.
- Link to paper:
  - http://wyner.info/LanguageLogicLawSoftware/?p=1858

Thanks for your attention!

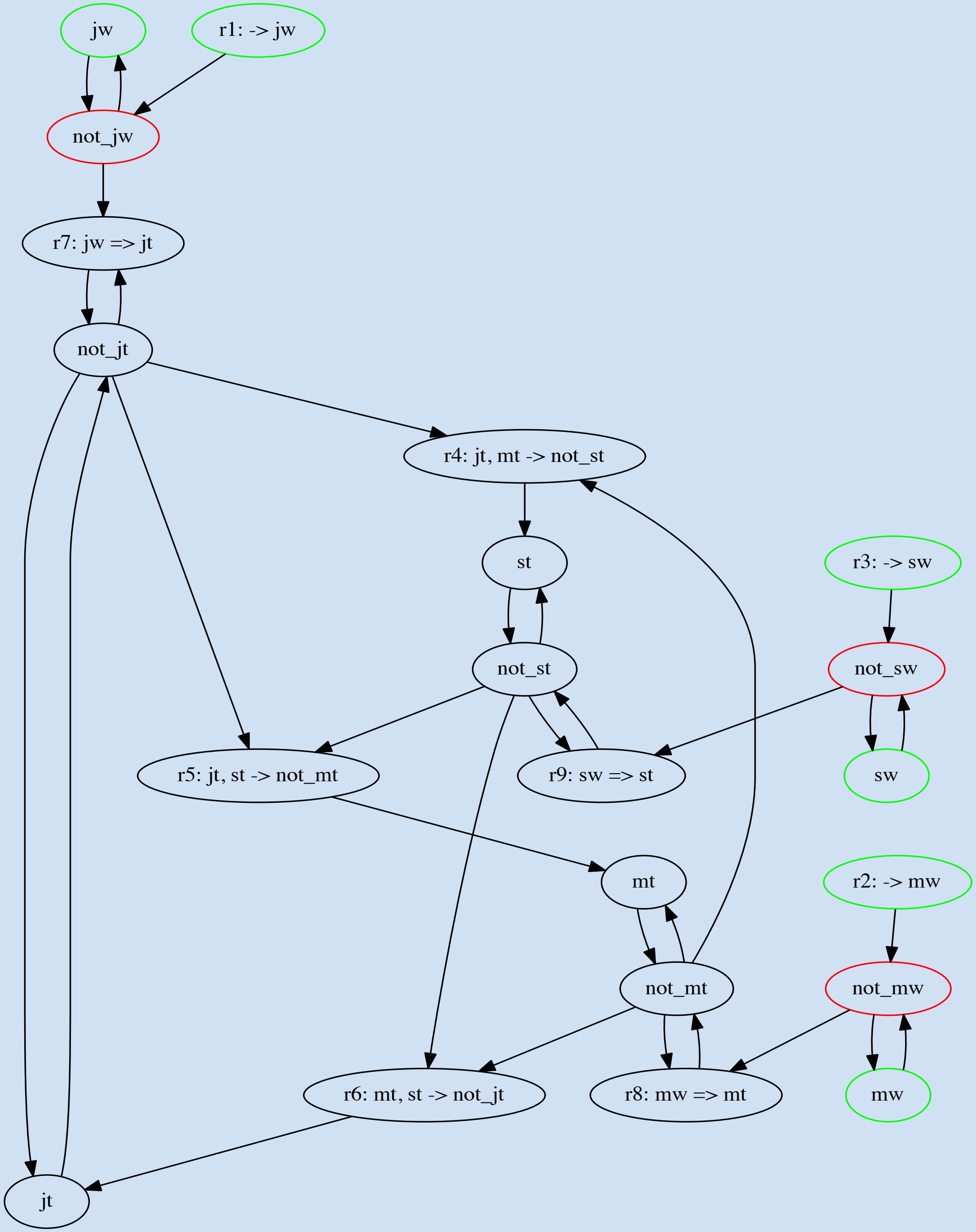
- Appreciation to Federico Cerutti for discussions.
- Questions?
- Contacts:

  – Adam Wyner

    azwyner@abdn.ac.uk

PS – Tandem Example

- S ={→jw; →mw; →sw; mt,st→¬jt; jt,st→¬mt; jt,mt→¬st} and D={jw⇒jt; mw⇒mt; sw⇒st}.
- John, Mary and Suzy want to go cycling in the coutryside (→ jw; → mw; → sw). They have a tandem bicycle, on which they all want to grab a seat (jw ⇒ jt, mw ⇒ mt; sw ⇒ st). However, since the tandem only has two seats, they cannot all three be on it (mt,st → ¬jt; jt,st → ¬mt; jt,mt → ¬st).
- With restricted rebut, we seem to get OK extensions, but have to buy restricted rebut. With unrestricted rebut, we get contradictions except under the grounded extension, which amounts to saying that we don't really know anything particular other than what John, Mary, and Suzy want. It seems intuitive that the unrestricted rebut interpretation is right. But, then how do we get it to work correctly?


jw
r1: -> jw
not_jw
r7: jw => jt
not_jt
r4: jt, mt -> not_st
st
r3: -> sw
not_st
not_sw
r5: jt, st -> not_mt
r9: sw => st
sw
mt
r2: -> mw
not_mt
not_mw
r6: mt, st -> not_jt
r8: mw => mt
mw
jt

Tandem Results

- The grounded comes out the same as with unrestricted rebut. The preferred extensions are below. Everyone always wants to be on the bike (sceptical); we have three for one person being on the tandem (credulous); we have three pairwise being on the tandem (credulous); we have no contradictions:

- {jw, sw, mw, not_jt, not_st, not_mt, r1, r2, r3}
- {jw, sw, mw, r1, r2, r3, r4, r5, r6, r7, r8, r9}
- {jw, sw, mw, st, not_jt, not_mt, r1, r2, r3, r9}
- {jw, sw, mw, mt, not_jt, not_st, r1, r2, r3, r8}
- {jw, sw, mw, jt, not_st, not_mt, r1, r2, r3, r7}
- {jw, sw, mw, st, mt, not_jt, r1, r2, r3, r6, r8, r9}
- {jw, sw, mw, jt, st, not_mt, r1, r2, r3, r5, r7, r9}
- {jw, sw, mw, jt, mt, not_st, r1, r2, r3, r4, r7, r8}