Representational Issues in DL
Example 13.1: Clever Students

Consider the following sentences:

- *Students are usually clever.*
- *Clever people usually make good career.*
- *Peter is a student.*
Example 13.1: Clever Students

Consider the following sentences:

- Students are usually clever.
- Clever people usually make good career.
- Peter is a student.

Representation in DL:

\[ A = \{ \text{Student}(peter) \} \]

\[ \Delta = \left\{ \frac{\text{Student}(x) : \text{Clever}(x)}{\text{Clever}(x)}, \frac{\text{Clever}(x) : \text{MakesCareer}(x)}{\text{MakesCareer}(x)} \right\} . \]
Example 13.1 (cont.)

The closure of $T = (A, \Delta)$:

$$A = \{ \text{Student}(\text{peter}) \}$$

$$\Delta = \left\{ \begin{array}{l}
\delta_1 = \frac{\text{Student}(\text{peter}) : \text{Clever}(\text{peter})}{\text{Clever}(\text{peter})}, \\
\delta_2 = \frac{\text{Clever}(\text{peter}) : \text{MakesCareer}(\text{peter})}{\text{MakesCareer}(\text{peter})}
\end{array} \right\}. $$
Example 13.1 (cont.)

The closure of $T = (A, \Delta)$:

$$A = \{ \text{Student}(peter) \}$$

\[
\Delta = \left\{ \begin{array}{l}
\delta_1 = \frac{\text{Student}(peter) : \text{Clever}(peter)}{\text{Clever}(peter)}, \\
\delta_2 = \frac{\text{Clever}(peter) : \text{MakesCareer}(peter)}{\text{MakesCareer}(peter)}
\end{array} \right. 
\]

Application of $\delta_1$ gives $\text{Clever}(peter)$. Then $\delta_2$ is applicable, after its application we get $\text{MakesCareer}(peter)$. 
The closure of $T = (A, \Delta)$:

\[ A = \{ \text{Student}(\text{peter}) \} \]

\[ \Delta = \left\{ \begin{array}{l}
\delta_1 = \frac{\text{Student}(\text{peter}) : \text{Clever}(\text{peter})}{\text{Clever}(\text{peter})}, \\
\delta_2 = \frac{\text{Clever}(\text{peter}) : \text{MakesCareer}(\text{peter})}{\text{MakesCareer}(\text{peter})}
\end{array} \right\}. \]

Application of $\delta_1$ gives $\text{Clever}(\text{peter})$. Then $\delta_2$ is applicable, after its application we get $\text{MakesCareer}(\text{peter})$.

Here *transitivity of defaults* is desirable.
Example 13.2: Married Students

Consider the following sentences:

- *Mike is a student.*
- *Students are usually adults.*
- *Adults are usually married.*

Representation in DL:

\[ A = \{ \text{Student}(mike) \} \]

\[ \Delta = \left\{ \frac{\text{Student}(x) : \text{Adult}(x)}{\text{Adult}(x)}, \frac{\text{Adult}(x) : \text{IsMarried}(x)}{\text{IsMarried}(x)} \right\} \]
Example 13.2: Married Students

Consider the following sentences:

- *Mike is a student.*
- *Students are usually adults.*
- *Adults are usually married.*

Representation in DL:

\[
A = \{ \text{Student}(mike) \} \\
\Delta = \left\{ \frac{\text{Student}(x)}{\text{Adult}(x)} , \frac{\text{Adult}(x)}{\text{IsMarried}(x)} \right\}
\]

One can easily note that we get *IsMarried(mike)*. Moreover, for an arbitrary student we will obtain a similar conclusion!
Example 13.2: Married Students

Consider the following sentences:

- Mike is a student.
- Students are usually adults.
- Adults are usually married.

Representation in DL:

\[
A = \{ \text{Student}(\text{mike}) \}
\]

\[
\Delta = \left\{ \frac{\text{Student}(x) : \text{Adult}(x)}{\text{Adult}(x)}, \frac{\text{Adult}(x) : \text{IsMarried}(x)}{\text{IsMarried}(x)} \right\}
\]

One can easily note that we get \text{IsMarried}(\text{mike}). Moreover, for an arbitrary student we will obtain a similar conclusion!

Here \textit{transitivity of defaults} is not desirable!
Example 13.2 (cont.)

In other to block undesirable conclusion, we can use a representation:

\[
\Delta = \left\{ \begin{array}{c}
\text{Student}(x) : \text{Adult}(x) \\
\text{Adult}(x) \\
\text{Adult}(x) : \text{IsMarried}(x) \land \neg \text{Student}(x) \\
\text{IsMarried}(x)
\end{array} \right\}.
\]

In other words, we substitute the second rule by:

*Typical adults are married unless they are students.*

Only the first default is applicable and we get \( \text{Adult}(mike) \). The previous conclusion \( \text{IsMarried}(mike) \) cannot be derived anymore.
Example 13.3: Reasoning by Case

Consider the following sentences:

- *Tweety is a bird or a bee.*
- *Typical birds can fly.*
- *Typical bees can fly.*

Intuitively we feel that *Tweety can fly.*
Example 13.3: Reasoning by Case

Consider the following sentences:

- Tweety is a bird or a bee.
- Typical birds can fly.
- Typical bees can fly.

Intuitively we feel that Tweety can fly.

Representation in DL:

\[ A = \{ \text{Bird}(\text{tweety}) \lor \text{Bee}(\text{tweety}) \} \]

\[ \Delta = \left\{ \frac{\text{Bird}(x) : \text{CanFly}(x)}{\text{CanFly}(x)}, \frac{\text{Bee}(x) : \text{CanFly}(x)}{\text{CanFly}(x)} \right\} \]
Example 13.3: Reasoning by Case

Consider the following sentences:

- *Tweety is a bird or a bee.*
- *Typical birds can fly.*
- *Typical bees can fly.*

Intuitively we feel that *Tweety can fly.*

Representation in DL:

\[
A = \{ \text{Bird}(\text{tweety}) \lor \text{Bee}(\text{tweety}) \} \\
\Delta = \left\{ \frac{\text{Bird}(x) : \text{CanFly}(x)}{\text{CanFly}(x)}, \frac{\text{Bee}(x) : \text{CanFly}(x)}{\text{CanFly}(x)} \right\}
\]

None of two defaults is applicable wrt \( A \), so \( E = Th(A) \). Consequently, we cannot derive a natural conclusion “*Tweety can fly*.”
We can use the following representation:

\[
\Delta = \left\{ \begin{array}{c}
\frac{\text{Bird}(x) \rightarrow \text{CanFly}(x)}{\text{Bird}(x) \rightarrow \text{CanFly}(x)}, \\
\frac{\text{Bee}(x) \rightarrow \text{CanFly}(x)}{\text{Bee}(x) \rightarrow \text{CanFly}(x)}
\end{array} \right\}
\]
Example 13.3 (cont.)

We can use the following representation:

\[ \Delta = \left\{ \frac{Bird(x) \rightarrow CanFly(x)}{Bird(x) \rightarrow CanFly(x)}, \frac{Bee(x) \rightarrow CanFly(x)}{Bee(x) \rightarrow CanFly(x)} \right\} \]

Now we get

\[ Bird(tweety) \rightarrow CanFly(tweety) \]
\[ Bee(tweety) \rightarrow CanFly(tweety), \]

or equivalently \[ Bird(tweety) \lor Bee(tweety) \rightarrow CanFly(tweety). \] From \( A \) we immediately get: \( CanFly(tweety). \)
Example 13.4

Consider the following sentences:

- *Teenagers are not adults.*
- *Students are usually adults.*
- *Peter is a teenager.*

Intuitively, we want to conclude: *Peter is not a student.*
Example 13.4

Consider the following sentences:

- Teenagers are not adults.
- Students are usually adults.
- Peter is a teenager.

Intuitively, we want to conclude: Peter is not a student.

Representation in DL:

\[
A = \left\{ \begin{array}{l}
\text{Teenager}(\text{peter}) \\
\forall x. \text{Teenager}(x) \rightarrow \neg \text{Adult}(x)
\end{array} \right\}
\]

\[
\Delta = \left\{ \begin{array}{l}
\delta(x) = \frac{\text{Student}(x) : \text{Adult}(x)}{\text{Adult}(x)}
\end{array} \right\}
\]
Consider the following sentences:

- Teenagers are not adults.
- Students are usually adults.
- Peter is a teenager.

Intuitively, we want to conclude: Peter is not a student.

Representation in DL:

\[
A = \left\{ \begin{array}{c}
\text{Teenager}(\text{peter}) \\
\forall x. \text{Teenager}(x) \rightarrow \neg \text{Adult}(x)
\end{array} \right\}
\]

\[
\Delta = \left\{ \begin{array}{c}
\delta(x) = \frac{\text{Student}(x) : \text{Adult}(x)}{\text{Adult}(x)}
\end{array} \right\}
\]

From \( A \) we conclude \( \neg \text{Adult}(\text{peter}) \). Since \( \delta(\text{peter}) \) is not applicable, we cannot derive the expected conclusion!
We can use another representation:

\[
A = \left\{ \begin{array}{l}
\text{Teenager}(peter) \\
\forall x. \text{Teenager}(x) \rightarrow \neg \text{Adult}(x)
\end{array} \right\}
\]

\[
\Delta = \left\{ \begin{array}{l}
\delta(x) = \frac{\text{Student}(x) \rightarrow \text{Adult}(x)}{\text{Student}(x) \rightarrow \text{Adult}(x)}
\end{array} \right\}
\]
We can use another representation:

\[ A = \left\{ \begin{array}{l}
\text{Teenager}(peter) \\
\forall x. \text{Teenager}(x) \rightarrow \neg \text{Adult}(x)
\end{array} \right\} \]

\[ \Delta = \left\{ \begin{array}{l}
\delta(x) = \frac{\text{Student}(x) \rightarrow \text{Adult}(x)}{\text{Student}(x) \rightarrow \text{Adult}(x)}
\end{array} \right\} \]

From \( A \) we get \( \neg \text{Adult}(peter) \).
\( \delta(peter) \) is applicable and after its application we obtain
\( \text{Student}(peter) \rightarrow \text{Adult}(peter) \).
Hence we get \( \neg \text{Student}(peter) \).
Algorithm for computing extensions
Auxiliary notations and notions

$\text{Mod}(\alpha)$ — the set of all models of $\alpha$
Auxiliary notations and notions

- $Mod(\alpha)$ — the set of all models of $\alpha$
- $PERM(\Delta)$ — the set of all permutations of defaults from $\Delta$
Auxiliary notations and notions

- $Mod(\alpha)$ — the set of all models of $\alpha$
- $PERM(\Delta)$ — the set of all permutations of defaults from $\Delta$

Let $M$ be a set of models (of some set $B$ of formulas – beliefs) and let $J$ be the set of formulas (justifications of these beliefs).

A pair $(M, J)$ is a $(M, J)$–pair iff for every $\varphi \in J$, $M \cap Mod(\varphi) \neq \emptyset$.

E.g.,

- $(\{m : m \models p \land q\}, \{r, \neg r\})$ is an MJ–pair
- $(\{m : m \models p \land q\}, \{r, \neg q\})$ is not an MJ–pair.
Auxiliary notations and notions

- $Mod(\alpha)$ — the set of all models of $\alpha$
- $PERM(\Delta)$ — the set of all permutations of defaults from $\Delta$

Let $M$ be a set of models (of some set $B$ of formulas – beliefs) and let $J$ be the set of formulas (justifications of these beliefs).
A pair $(M, J)$ is a $(M, J)$–pair iff for every $\varphi \in J$, $M \cap Mod(\varphi) \neq \emptyset$.
E.g.,
- $(\{m : m \models p \land q\}, \{r, \neg r\})$ is an MJ–pair
- $(\{m : m \models p \land q\}, \{r, \neg q\})$ is not an MJ–pair.

A default $\delta = \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma}$ is applicable wrt $M$ iff
- every model $m \in M$ is a model of $\alpha$
- $M \cap Mod(\beta_i) \neq \emptyset$ for every $i = 1, \ldots, n$
Auxiliary notations and notions

- $Mod(\alpha)$ — the set of all models of $\alpha$

- $PERM(\Delta)$ — the set of all permutations of defaults from $\Delta$

Let $M$ be a set of models (of some set $B$ of formulas – beliefs) and let $J$ be the set of formulas (justifications of these beliefs).

A pair $(M, J)$ is a $(M, J)$–pair iff for every $\varphi \in J$, $M \cap Mod(\varphi) \neq \emptyset$.

E.g.,

- $\left(\{m : m \models p \land q\}, \{r, \neg r\}\right)$ is an MJ–pair
- $\left(\{m : m \models p \land q\}, \{r, \neg q\}\right)$ is not an MJ–pair.

A default $\delta = \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma}$ is applicable wrt $M$ iff

- every model $m \in M$ is a model of $\alpha$
- $M \cap Mod(\beta_i) \neq \emptyset$ for every $i = 1, \ldots, n$
Auxiliary notations and notions

- $\text{Mod}(\alpha)$ — the set of all models of $\alpha$
- $\text{PERM}(\Delta)$ — the set of all permutations of defaults from $\Delta$

Let $M$ be a set of models (of some set $B$ of formulas – beliefs) and let $J$ be the set of formulas (justifications of these beliefs).

A pair $(M, J)$ is a $(M, J)$–pair iff for every $\varphi \in J$, $M \cap \text{Mod}(\varphi) \neq \emptyset$.

E.g.,
- $(\{m : m \models p \land q\}, \{r, \neg r\})$ is an MJ–pair
- $(\{m : m \models p \land q\}, \{r, \neg q\})$ is not an MJ–pair.

A default $\delta = \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma}$ is applicable wrt $M$ iff
- every model $m \in M$ is a model of $\alpha$
- $M \cap \text{Mod}(\beta_i) \neq \emptyset$ for every $i = 1, \ldots, n$
Auxiliary notations and notions

- $\text{Mod}(\alpha)$ — the set of all models of $\alpha$
- $\text{PERM}(\Delta)$ — the set of all permutations of defaults from $\Delta$

Let $M$ be a set of models (of some set $B$ of formulas – beliefs) and let $J$ be the set of formulas (justifications of these beliefs). A pair $(M, J)$ is a $(M, J)$–pair iff for every $\varphi \in J$, $M \cap \text{Mod}(\varphi) \neq \emptyset$.

E.g.,
- $(\{m : m \models p \land q\}, \{r, \neg r\})$ is an MJ–pair
- $(\{m : m \models p \land q\}, \{r, \neg q\})$ is not an MJ–pair.

A default $\delta = \frac{\alpha}{\gamma} : \beta_1, \ldots, \beta_n$ is applicable wrt $M$ iff

- every model $m \in M$ is a model of $\alpha$
- $M \cap \text{Mod}(\beta_i) \neq \emptyset$ for every $i = 1, \ldots, n$
**Auxiliary notations and notions**

- \( \text{Mod}(\alpha) \) — the set of all models of \( \alpha \)
- \( \text{PERM}(\Delta) \) — the set of all permutations of defaults from \( \Delta \)

Let \( M \) be a set of models (of some set \( B \) of formulas – beliefs) and let \( J \) be the set of formulas (justifications of these beliefs).

A pair \((M, J)\) is a \((M, J)\)-pair iff for every \( \varphi \in J \), \( M \cap \text{Mod}(\varphi) \neq \emptyset \).

E.g.,

- \((\{m : m \models p \land q\}, \{r, \neg r\})\) is an MJ–pair
- \((\{m : m \models p \land q\}, \{r, \neg q\})\) is not an MJ–pair.

A default \( \delta = \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma} \) is applicable wrt \( M \) iff

- every model \( m \in M \) is a model of \( \alpha \)
- \( M \cap \text{Mod}(\beta_i) \neq \emptyset \) for every \( i = 1, \ldots, n \)
Let \((M, J)\) be an MJ–pair and let \(\Delta\) be a set of closed defaults. \((M, J)\) is called \(\Delta\text{-stable}\) iff for every default \(\delta = (\alpha : \beta_1, \ldots, \beta_n) / \gamma \in \Delta\) it holds:

- if \(\delta\) is applicable wrt \(M\), then
  - \(M \cap Mod(\gamma) = M\)
  - \(\beta_1, \ldots, \beta_n \in J\)
Stability

Let \((M, J)\) be an MJ–pair and let \(\Delta\) be a set of closed defaults. \((M, J)\) is called \(\Delta\)-stable iff for every default \(\delta = (\alpha : \beta_1, \ldots, \beta_n)/\gamma \in \Delta\) it holds:

\[
\text{if } \delta \text{ is applicable wrt } M, \text{ then}
\]

- \(M \cap Mod(\gamma) = M\)
- \(\beta_1, \ldots, \beta_n \in J\)

In other words,

- either \(\delta\) is applicable, but its application does not introduce new information,

- or \(\delta\) it is not applicable (and is just skipped).
Let $M$ be a set of models and let $J$ be a set of formulas. For simplicity we will consider defaults of the form $\delta = (\alpha : \beta)/\gamma$. For every $\delta$ define a mapping $d_\delta$, which transforms a pair $(M, J)$ into another pair $(M', J')$ as follows:

$$d_\delta(M, J) = \begin{cases} 
(M \cap Mod(\alpha), J \cup \{\beta\}) & \text{iff } \delta \text{ is applicable wrt } M \\
(M, J) & \text{iff } \delta \text{ is not applicable wrt } M \\
(\emptyset, \mathcal{L}) & \text{otherwise.}
\end{cases}$$

and $(M, J)$ is an (M,J)–pair.

and $(M, J)$ is an (M,J)–pair.

otherwise.
Example 13.5

Consider the following sets:

\[ M = \{ m : m \models p \land q \} = \begin{cases} \{p, q, r, s\} \\ \{p, q, r, \neg s\} \\ \{p, q, \neg r, s\} \\ \{p, q, \neg r, \neg s\} \end{cases}, \]

\[ J = \{ \neg r \land q, s \}, \]

\[ \Delta = \left\{ \delta_1 = \frac{p : q \land \neg r}{q}, \quad \delta_2 = \frac{r : s}{q} \right\}. \]

- \( d_{\delta_1} (M, J) = (M, J) \) — \( \delta_1 \) is applicable
- \( d_{\delta_2} (M, J) = (M, J) \) — \( \delta_2 \) is inapplicable.
Consider the following sets:

\[ M = \{ m : m \models p \land q \} = \begin{cases} \{ p, q, r, s \} \\ \{ p, q, r, \neg s \} \\ \{ p, q, \neg r, s \} \\ \{ p, q, \neg r, \neg s \} \end{cases}, \]

\[ J = \{ \neg r \land q, s \}, \]

\[ \Delta = \left\{ \delta_1 = \frac{p : q \land \neg r}{q}, \delta_2 = \frac{r : s}{q} \right\}. \]

- \[ d_{\delta_1}(M, J) = (M, J) — \delta_1 \text{ is applicable} \]
- \[ d_{\delta_2}(M, J) = (M, J) — \delta_2 \text{ is inapplicable.} \]

\((M, J)\) is \(\Delta\)-stable.
Example 13.6

\[ M = \{ m : m \models p \land q \} = \left\{ \begin{array}{l} \{ p, q, r, s \} \\ \{ p, q, r, \neg s \} \\ \{ p, q, \neg r, s \} \\ \{ p, q, \neg r, \neg s \} \end{array} \right\}, \quad J = \{ q \land \neg r, r \} \]

\[ \Delta = \left\{ \begin{array}{l} \delta_1 = \frac{p : q \land \neg r}{q} \\ \delta_2 = \frac{q : r}{r} \end{array} \right\}. \]
Example 13.6

\[
M = \{ m : m \models p \land q \} = \begin{cases} 
\{ p, q, r, s \} \\
\{ p, q, r, \neg s \} \\
\{ p, q, \neg r, s \} \\
\{ p, q, \neg r, \neg s \} 
\end{cases}, \quad J = \{ q \land \neg r, r \}
\]

\[
\Delta = \left\{ \delta_1 = \frac{p \land q \land \neg r}{q}, \quad \delta_2 = \frac{q \land r}{r} \right\}.
\]

- \( d_{\delta_1}(M, J) = (M, J) \quad \text{— \( \delta_1 \) is applicable} \)
- \( d_{\delta_2}(M, J) = \{ m : m \models p \land q \land r \}, \quad J \neq (M, J). \)

Notice that \( \varphi = q \land \neg r \) does not hold in \( \{ m : m \models p \land q \land r \} \), so \( d_{\delta_2}(M, J) \) is not an MJ–pair.
Example 13.6

\[
M = \{ m : m \models p \land q \} = \left\{ \begin{array}{l}
\{ p, q, r, s \} \\
\{ p, q, r, \neg s \} \\
\{ p, q, \neg r, s \} \\
\{ p, q, \neg r, \neg s \}
\end{array} \right\}, \quad J = \{ q \land \neg r, r \}
\]

\[
\Delta = \left\{ \begin{array}{l}
\delta_1 = \frac{p \land \neg r}{q}, \quad \delta_2 = \frac{q \land r}{r}
\end{array} \right\}.
\]

- \( d_{\delta_1}(M, J) = (M, J) \) — \( \delta_1 \) is applicable
- \( d_{\delta_2}(M, J) = \{ m : m \models p \land q \land r \}, J \neq (M, J). \)

Notice that \( \varphi = q \land \neg r \) does not hold in \( \{ m : m \models p \land q \land r \} \), so \( d_{\delta_2}(M, J) \) is not an MJ–pair.

\( (M, J) \) is not \( \Delta \)–stable.
Algorithm

**Input:** A closed default theory $T = (A, \Delta)$

**Output:** All extensions $E_1, \ldots, E_k$ of $T$.

In fact, we will determine the set $M_1, \ldots, M_k$ of all models of extensions of $T$. 
Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

(S1) Put $P := \text{PERM}(\Delta)$; $\mathcal{M} := \emptyset$
Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

(S1) Put $P := \text{PERM}(\Delta)$; $M := \emptyset$

(S2) If $P = \emptyset$ then STOP $\implies M$ semantically corresponds to the set of all extensions of $T$; otherwise $M := \text{Mod}(A)$; $J := \emptyset$;
Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

**S1**  Put $P := \text{PERM}(\Delta)$; $\mathcal{M} := \emptyset$

**S2**  If $P = \emptyset$ then STOP $\implies \mathcal{M}$ semantically corresponds to the set of all extensions of $T$; otherwise $M := \text{Mod}(A)$; $J := \emptyset$;

**S3**  Take $\delta = (\delta_1, \ldots, \delta_k) \in P$, $P := P \setminus \{\delta\}$; $i := 1$
Algorithm (cont.)

Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

(S1) Put $P := \text{PERM} (\Delta)$; $\mathcal{M} := \emptyset$

(S2) If $P = \emptyset$ then STOP $\Longrightarrow \mathcal{M}$ semantically corresponds to the set of all extensions of $T$; otherwise $M := \text{Mod} (A)$; $J := \emptyset$

(S3) Take $\delta = (\delta_1, \ldots, \delta_k) \in P$, $P := P \setminus \{\delta\}$; $i := 1$

(S4) If $i > k$ then
  if $(M, J)$ is $\Delta$–stable then $\mathcal{M} := \mathcal{M} \cup \{M\}$; go to (S2);
  otherwise go to (S2);
otherwise take $\delta_i = (\alpha : \beta)/\gamma$ and put $i := i + 1$;
Algorithm (cont.)

Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

(S1) Put $P := \text{PERM}(\Delta)$; $\mathcal{M} := \emptyset$

(S2) If $P = \emptyset$ then STOP $\implies \mathcal{M}$ semantically corresponds to the set of all extensions of $T$; otherwise $M := \text{Mod}(A)$; $J := \emptyset$;

(S3) Take $\delta = (\delta_1, \ldots, \delta_k) \in P$, $P := P \setminus \{\delta\}$; $i := 1$

(S4) If $i > k$ then
   if $(M, J)$ is $\Delta$–stable then $\mathcal{M} := \mathcal{M} \cup \{M\}$; go to (S2);
   otherwise go to (S2);
   otherwise take $\delta_i = (\alpha : \beta)/\gamma$ and put $i := i + 1$;

(S.5) If $M \subseteq \text{Mod}(\alpha)$ and $M \cap \text{Mod}(\beta) \neq \emptyset$
   then $J := J \cup \{\beta\}$; $M := M \cap \text{Mod}(\gamma)$; go to (S6);
   otherwise go to (S4)
Let $T(A, \Delta)$ and assume that $\Delta \neq \emptyset$.

(S1) Put $P := PERM(\Delta)$; $\mathcal{M} := \emptyset$

(S2) If $P = \emptyset$ then STOP $\iff \mathcal{M}$ semantically corresponds to the set of all extensions of $T$; otherwise $M := \text{Mod}(A)$; $J := \emptyset$;

(S3) Take $\delta = (\delta_1, \ldots, \delta_k) \in P$, $P := P \setminus \{\delta\}$; $i := 1$

(S4) If $i > k$ then
   - if $(M, J)$ is $\Delta$–stable then $\mathcal{M} := \mathcal{M} \cup \{M\}$; go to (S2);
   - otherwise go to (S2);
   - otherwise take $\delta_i = (\alpha : \beta)/\gamma$ and put $i := i + 1$;

(S.5) If $M \subseteq \text{Mod}(\alpha)$ and $M \cap \text{Mod}(\beta) \neq \emptyset$
   then $J := J \cup \{\beta\}$; $M := M \cap \text{Mod}(\gamma)$; go to (S6);
   otherwise go to (S4)

(S6) If $(M, J)$ is an MJ–pair then go to (S4); otherwise go to (S2).
The algorithm for computing extensions of $T = (A, \Delta)$ can be depicted by a transition network $N = (V, E)$ such that

- Each node $v \in V$ is labeled by $(M, J)$, where $M$ is a set of models (of a set of beliefs) and $J$ is a set of sentences (justifications of applied defaults).
The algorithm for computing extensions of $T = (A, \Delta)$ can be depicted by a

transition network $N = (V, E)$ such that

- Each node $v \in V$ is labeled by $(M, J)$, where $M$ is a set of models (of a set of beliefs) and $J$ is a set of sentences (justifications of applied defaults).

- Each node is either
  - **viable** — labeled by $(M, J)$, where $(M, J)$ is an MJ–pair
  - **contradictory** — otherwise.
The algorithm for computing extensions of $T = (A, \Delta)$ can be depicted by a transition network $N = (V, E)$ such that

- Each node $v \in V$ is labeled by $(M, J)$, where $M$ is a set of models (of a set of beliefs) and $J$ is a set of sentences (justifications of applied defaults).
- Each node is either
  - **viable** — labeled by $(M, J)$, where $(M, J)$ is an MJ–pair
  - **contradictory** — otherwise.
- A root is labelled by $(Mod(A), \emptyset)$. 

**Transition network**
The algorithm for computing extensions of \( T = (A, \Delta) \) can be depicted by a transition network \( N = (V, E) \) such that

- Each node \( v \in V \) is labeled by \((M, J)\), where \( M \) is a set of models (of a set of beliefs) and \( J \) is a set of sentences (justifications of applied defaults).

- Each node is either
  - **viable** — labeled by \((M, J)\), where \((M, J)\) is an MJ–pair
  - **contradictory** — otherwise.

- A root is labelled by \((\text{Mod}(A), \emptyset)\).
The algorithm for computing extensions of \( T = (A, \Delta) \) can be depicted by a transition network \( N = (V, E) \) such that

- Each node \( v \in V \) is labeled by \( (M, J) \), where \( M \) is a set of models (of a set of beliefs) and \( J \) is a set of sentences (justifications of applied defaults).
- Each node is either
  - **viable** — labeled by \( (M, J) \), where \( (M, J) \) is an MJ–pair
  - **contradictory** — otherwise.
- A root is labelled by \( (Mod(A), \emptyset) \).
The algorithm for computing extensions of $T = (A, \Delta)$ can be depicted by a transition network $N = (V, E)$ such that

- Each node $v \in V$ is labeled by $(M, J)$, where $M$ is a set of models (of a set of beliefs) and $J$ is a set of sentences (justifications of applied defaults).
- Each node is either
  - viable — labeled by $(M, J)$, where $(M, J)$ is an MJ–pair
  - contradictory — otherwise.
- A root is labelled by $(\text{Mod}(A), \emptyset)$. 
Transition network (cont.)

- Each arc is labeled by \( \delta \in \Delta \).
Transition network (cont.)

- Each arc is labeled by $\delta \in \Delta$.
- From each viable node there are $n = |\Delta|$ arcs labeled by $\delta \in \Delta$. 
Transition network (cont.)

- Each arc is labeled by $\delta \in \Delta$.
- From each viable node there are $n = |\Delta|$ arcs labeled by $\delta \in \Delta$.
- If a node $v$ is labeled by $(M, J)$ and $\delta = (\alpha : \beta)/\gamma$ is such that $M \subseteq \text{Mod}(\alpha)$ and $M \cap \text{Mod}(\neg \beta) = \emptyset$, then there is an arc labeled by $\delta$ leading to $v'$ labeled by $(M \cap \text{Mod}(\gamma), J \cup \{\beta\})$. 
Each arc is labeled by $\delta \in \Delta$.

From each viable node there are $n = |\Delta|$ arcs labeled by $\delta \in \Delta$.

If a node $v$ is labeled by $(M, J)$ and $\delta = (\alpha : \beta)/\gamma$ is such that $M \subseteq Mod(\alpha)$ and $M \cap Mod(\neg \beta) = \emptyset$, then there is an arc labeled by $\delta$ leading to $v'$ labeled by $(M \cap Mod(\gamma), J \cup \{\beta\})$.

A leaf is either

- a contradictory node, or
- a node such that all arcs starting in this node loop back — such a node, labeled by $(M, J)$ represents an extension of $T$: $M$ is the set of all models of the extension.
Transition network (cont.)

- Each arc is labeled by $\delta \in \Delta$.
- From each viable node there are $n = |\Delta|$ arcs labeled by $\delta \in \Delta$.
- If a node $v$ is labeled by $(M, J)$ and $\delta = (\alpha : \beta)/\gamma$ is such that $M \subseteq Mod(\alpha)$ and $M \cap Mod(\neg \beta) = \emptyset$, then there is an arc labeled by $\delta$ leading to $v'$ labeled by $(M \cap Mod(\gamma), J \cup \{\beta\})$.
- A leaf is either
  - a contradictory node, or
  - a node such that all arcs starting in this node loop back — such a node, labeled by $(M, J)$ represents an extension of $T$: $M$ is the set of all models of the extension.
Transition network (cont.)

- Each arc is labeled by $\delta \in \Delta$.

- From each viable node there are $n = |\Delta|$ arcs labeled by $\delta \in \Delta$.

- If a node $v$ is labeled by $(M, J)$ and $\delta = (\alpha : \beta)/\gamma$ is such that $M \subseteq Mod(\alpha)$ and $M \cap Mod(\neg \beta) = \emptyset$, then there is an arc labeled by $\delta$ leading to $v'$ labeled by $(M \cap Mod(\gamma), J \cup \{\beta\})$.

- A leaf is either
  - a contradictory node, or
  - a node such that all arcs starting in this node loop back — such a node, labeled by $(M, J)$ represents an extension of $T$: $M$ is the set of all models of the extension.
Example 13.7

\[ T = (A, \Delta), \text{ where } A = \{p, q\}, \Delta = \left\{ \delta_1 = \frac{p \cdot r \land \neg s}{r}, \delta_2 = \frac{q \cdot s}{s} \right\}. \]
Example 13.7

\[ T = (A, \Delta), \text{ where } A = \{p, q\}, \Delta = \left\{ \delta_1 = \frac{p \land r \land \neg s}{r}, \delta_2 = \frac{q}{s} \right\}. \]
Example 13.8

\[ T = (A, \Delta), \text{ where } A = \emptyset \text{ and } \Delta = \delta = \left\{ \frac{\neg p}{p} \right\}. \]
Example 13.8

\[ T = (A, \Delta), \text{ where } A = \emptyset \text{ and } \Delta = \delta = \left\{ \frac{\neg p}{p} \right\}. \]

\[ \{(m : m \models \top), \emptyset\} \]

Contradictory node!

\[ \{(m : m \models \neg p), \neg p \}\]
Example 13.9

\[ T = (A, \Delta), \text{ where } A = \{p\} \text{ and } \Delta = \left\{ \delta_1 = \frac{p:q}{q}, \delta_2 = \frac{p:q, \neg q}{r} \right\} \]
Example 13.9

\[ T = (A, \Delta), \text{ where } A = \{p\} \text{ and } \Delta = \left\{ \delta_1 = \frac{p : q}{q}, \delta_2 = \frac{p : q, \neg q}{r} \right\} \]

\[ (\{m : m \vDash p\}, \emptyset) \]

\[ (\{m : m \vDash p \land q\}, \{q\}) \]

\[ (\{m : m \vDash p \land r\}, \{q, \neg q\}) \]

extension node

\[ (\{m : m \vDash p \land r \land q\}, \{q, \neg q\}) \]

contradictory node!
Problems with DL

Some default theories have no extensions, because the criterion of defaults’ applicability is too weak — it enforces the default to be applicable, but after its application the default is not applicable.
Some default theories have no extensions, because the criterion of defaults’ applicability is too weak — it enforces the default to be applicable, but after its application the default is not applicable.

The consequent of the default, together with axioms and consequents of previously applied defaults, contradicts some of its own justifications. For example,

\[ A = \emptyset, \quad \Delta = \left\{ \vdash p, \neg p \right\}. \]
The consequent of the default, together with axioms and consequents of previously applied defaults, denies some justification of previously applied defaults.

For example,

\[ A = \emptyset, \quad \Delta = \left\{ \frac{p \land r}{p}, \frac{p : \neg r}{\neg r} \right\}. \]
The consequent of the default, together with axioms and consequents of previously applied defaults, denies some justification of previously applied defaults.
For example,

\[ A = \emptyset, \quad \Delta = \left\{ \frac{p \land r}{p}, \frac{p \land \neg r}{\neg r} \right\}. \]

The consequent of the default contradicts some sentence derivable from axioms and consequents of already applied defaults.
For example,

\[ A = \{p \rightarrow q\}, \quad \Delta = \left\{ \frac{p}{p}, \frac{p \land s}{\neg q} \right\}. \]
Two default logics

In order to overcome these problems, the alternative default system was introduced (Łukaszewicz 1988). Henceforth we will use the following terminology:

- Reiter’s default logic (RDL) for the standard default logic
- Alternative default logic (ADL) for the new one.
Alternative Default Logic
Let $T = (A, \Delta)$ be a closed default theory over the language $\mathcal{L}$. Define two operators $\Gamma_1$ and $\Gamma_2$ specified for pairs of sentences such that for any pair $(B, J)$, $\Gamma_1(B, J)$ and $\Gamma_2(B, J)$ are the smallest sets of sentences satisfying:
Let \( T = (A, \Delta) \) be a closed default theory over the language \( \mathcal{L} \). Define two operators \( \Gamma_1 \) and \( \Gamma_2 \) specified for pairs of sentences such that for any pair \( (B, J) \), \( \Gamma_1(B, J) \) and \( \Gamma_2(B, J) \) are the smallest sets of sentences satisfying:
Let $T = (A, \Delta)$ be a closed default theory over the language $\mathcal{L}$. Define two operators $\Gamma_1$ and $\Gamma_2$ specified for pairs of sentences such that for any pair $(B, J)$, $\Gamma_1(B, J)$ and $\Gamma_2(B, J)$ are the smallest sets of sentences satisfying:
Let $T = (A, \Delta)$ be a closed default theory over the language $\mathcal{L}$. Define two operators $\Gamma_1$ and $\Gamma_2$ specified for pairs of sentences such that for any pair $(B, J)$, $\Gamma_1(B, J)$ and $\Gamma_2(B, J)$ are the smallest sets of sentences satisfying:

A set $E \subseteq \mathcal{L}$ of sentences is an \textit{alternative extension of $T$ wrt $F$} iff $E = \Gamma_1(E, F)$ and $F = \Gamma_2(E, F)$. $E$ is an \textit{alternative extension of $T$} iff there is a set $F \subseteq \mathcal{L}$ of sentences such that $E$ is an alternative extension of $T$ wrt $F$. 


dr Anna M. Radzikowska, Knowledge Representation 13, – p. 28/??
Given two sets of sentences, $B$ and $J$, if

then $\gamma \in \Gamma_1(B, J)$ and $\beta_1, \ldots, \beta_n \in \Gamma_2(B, J)$. 
Given two sets of sentences, $B$ and $J$, if

then $\gamma \in \Gamma_1(B, J)$ and $\beta_1, \ldots, \beta_n \in \Gamma_2(B, J)$. 
Given two sets of sentences, $B$ and $J$, if

then $\gamma \in \Gamma_1(B, J)$ and $\beta_1, \ldots, \beta_n \in \Gamma_2(B, J)$. 
Given two sets of sentences, $B$ and $J$, if

$$\gamma \in \Gamma_1(B, J) \quad \text{and} \quad \beta_1, \ldots, \beta_n \in \Gamma_2(B, J).$$
Algorithm for computing alternative extensions
Algorithm

S1. \( P := \text{PERM}(\Delta); \)
Algorithm

S1. \( P := PERM(\Delta); \)

S2. If \( P = \emptyset \) then STOP \( \implies \mathcal{M} \) semantically corresponds to the set of all extensions of \( T \); otherwise \( M := \text{Mod}(A); J := \emptyset; \mathcal{M} := \emptyset \)
Algorithm

S1. \( P := \text{PERM}(\Delta); \)

S2. If \( P = \emptyset \) then STOP \( \Rightarrow \mathcal{M} \) semantically corresponds to the set of all extensions of \( T \); otherwise \( M := \text{Mod}(A); \ J := \emptyset; \ \mathcal{M} := \emptyset \)

S3. Take \( \delta = (\delta_1, \ldots, \delta_k) \in P, \ P := P \setminus \{\delta\}; \ i := 1 \)
Algorithm

S1. \[ P := \text{PERM}(\Delta); \]

S2. If \( P = \emptyset \) then STOP \( \implies \mathcal{M} \) semantically corresponds to the set of all extensions of \( T \); otherwise \( M := \text{Mod}(A); \ J := \emptyset; \ \mathcal{M} := \emptyset \)

S3. Take \( \delta = (\delta_1, \ldots, \delta_k) \in P, \ P := P \setminus \{\delta\}; \ i := 1 \)

S4. If \( i > k \) then
   - if \((M, J)\) is \( \Delta \)-stable then \( \mathcal{M} := \mathcal{M} \cup \{M\}; \) go to S2;
   - otherwise go to S2;
   - otherwise take \( \delta_i = (\alpha : \beta)/\gamma \) and put \( i := i + 1; \)
Algorithm

S1. \( P := PERM(\Delta); \)

S2. If \( P = \emptyset \) then STOP \( \Rightarrow \mathcal{M} \) semantically corresponds to the set of all extensions of \( T \); otherwise \( M := \text{Mod}(A); \ J := \emptyset; \ \mathcal{M} := \emptyset \)

S3. Take \( \delta = (\delta_1, \ldots, \delta_k) \in P, \ P := P \setminus \{\delta\}; \ i := 1 \)

S4. If \( i > k \) then
   if \( (M, J) \) is \( \Delta \)-stable then \( \mathcal{M} := \mathcal{M} \cup \{M\}; \) go to S2;
   otherwise go to S2;
   otherwise take \( \delta_i = (\alpha : \beta)/\gamma \) and put \( i := i + 1; \)

S5. If \( M \subseteq \text{Mod}(\alpha) \) and \( M \cap \text{Mod}(\gamma) \cap \text{Mod}(\varphi) \neq \emptyset \) for every \( \varphi \in J \cup \{\beta\} \) then \( J := J \cup \{\beta\}; \ M := M \cap \text{Mod}(\gamma); \) go to S4;
   otherwise go to S4
Algorithm

S1. \[ P := P ERM(\Delta); \]

S2. If \( P = \emptyset \) then STOP \( \implies \mathcal{M} \) semantically corresponds to the set of all extensions of \( T \); otherwise \( M := \operatorname{Mod}(A); \ J := \emptyset; \ \mathcal{M} := \emptyset \)

S3. Take \( \delta = (\delta_1, \ldots, \delta_k) \in P, \ P := P \setminus \{\delta\}; \ i := 1 \)

S4. If \( i > k \) then
   if \((M, J)\) is \(\Delta\)–stable then \(M := \mathcal{M} \cup \{M\};\) go to \(S2\);
   otherwise go to \(S2\);
   otherwise take \(\delta_i = (\alpha : \beta)/\gamma\) and put \(i := i + 1;\)

S5. If \( M \subseteq \operatorname{Mod}(\alpha) \) and \( M \cap \operatorname{Mod}(\gamma) \cap \operatorname{Mod}(\varphi) \neq \emptyset \) for every \( \varphi \in J \cup \{\beta\} \)
   then \( J := J \cup \{\beta\}; \ M := M \cap \operatorname{Mod}(\gamma); \) go to \(S4\);
   otherwise go to \(S4\)

Remark: \((M, J)\) determine in \(S5\) are MJ–pairs!
Example 13.10

Consider the following story:

- **On Sundays Bill usually goes fishing except when he is tired.**
- **If Bill worked hard the day before, then he is usually tired except when he woke up late.**
- **If Bill is on vacations, then he usually wakes up late except when he goes fishing.**
- **Today is Sunday, Bill worked hard yesterday, and he has vacations.**

**Representation in DL:**

\[
A = \{s, w, v\}, \quad \Delta = \\
\left\{ \delta_1 = \frac{s : g \land \neg t}{g}, \quad \delta_2 = \frac{w : t \land \neg l}{t}, \quad \delta_3 = \frac{v : l \land \neg g}{l} \right\}.
\]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s \land \neg t}{g}, \delta_2 = \frac{w \land \neg l}{t}, \delta_3 = \frac{v \land \neg g}{l} \right\}. \]

Take \((\delta_1, \delta_2, \delta_3)\).
Example 13.10 (cont.)

\[ A = \{s, w, v\} , \; \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g} , \delta_2 = \frac{w : t \land \neg l}{t} , \delta_3 = \frac{v : l \land \neg g}{l} \right\} \]

Take \((\delta_1, \delta_2, \delta_3)\).

\[ \delta_1 \text{ is applicable, so } \]

\[ M_1 = \begin{cases} 
\{s, w, v, g, t, l\} \\
\{s, w, v, g, t, \neg l\} \\
\{s, w, v, g, \neg t, l\} \\
\{s, w, v, g, \neg t, \neg l\} 
\end{cases} , \; J_1 = \{g \land \neg t\} , \; B = B_1 = Th(A \cup \{g\}). \]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \begin{array}{l} \delta_1 = \frac{s : g \land \neg t}{g} , \quad \delta_2 = \frac{w : t \land \neg l}{t} , \quad \delta_3 = \frac{v : l \land \neg g}{l} \end{array} \right\} . \]

Take \((\delta_1, \delta_2, \delta_3)\).

- \(\delta_1\) is applicable, so

\[
M_1 = \begin{cases} 
\{s, w, v, g, t, l\} \\
\{s, w, v, g, t, \neg l\} \\
\{s, w, v, g, \neg t, l\} \\
\{s, w, v, g, \neg t, \neg l\} 
\end{cases}, \quad J_1 = \{g \land \neg t\}, \quad B = B_1 = Th(A \cup \{g\}).
\]

- \(\delta_2\) is not applicable (\(t\) contradicts the justification of \(\delta_1\)), so

\(M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1.\)
Example 13.10 (cont.)

\[ A = \{s, w, v\} , \ \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g} , \delta_2 = \frac{w : t \land \neg l}{t} , \delta_3 = \frac{v : l \land \neg g}{l} \right\} . \]

Take \((\delta_1, \delta_2, \delta_3)\).

- \(\delta_1\) is applicable, so
  \[ M_1 = \left\{ \begin{array}{l}
  \{s, w, v, g, t, l\} \\
  \{s, w, v, g, t, \neg l\} \\
  \{s, w, v, g, \neg t, l\} \\
  \{s, w, v, g, \neg t, \neg l\}
  \end{array} \right\} , \ J_1 = \{g \land \neg t\} , \ B = B_1 = Th(A \cup \{g\}) . \]

- \(\delta_2\) is not applicable (\(t\) contradicts the justification of \(\delta_1\)), so
  \[ M_2 = M_1 , \ J_2 = J_1 , \ B_2 = B_1 . \]

- \(\delta_3\) is not applicable (as in Reiter’s criterion), so
  \[ M_3 = M_2 , \ J_3 = J_2 , \ B_3 = B_2 . \]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s: g \land \neg t}{g}, \delta_2 = \frac{w: t \land \neg l}{t}, \delta_3 = \frac{v: l \land \neg g}{l} \right\}. \]

Take \((\delta_1, \delta_2, \delta_3)\).

- \(\delta_1\) is applicable, so

\[
M_1 = \left\{ \begin{array}{l}
\{s, w, v, g, t, l\} \\
\{s, w, v, g, t, \neg l\} \\
\{s, w, v, g, \neg t, l\} \\
\{s, w, v, g, \neg t, \neg l\}
\end{array} \right\}, \quad J_1 = \{g \land \neg t\}, \quad B = B_1 = Th(A \cup \{g\}). \]

- \(\delta_2\) is not applicable (\(t\) contradicts the justification of \(\delta_1\)), so

\[ M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1. \]

- \(\delta_3\) is not applicable (as in Reiter’s criterion), so

\[ M_3 = M_2, \quad J_3 = J_2, \quad B_3 = B_2. \]

Hence \(B\) is the alternative extension of \(T\).
Example 13.10 (cont.)

\[ A = \{ s, w, v \} , \ \Delta = \left\{ \delta_1 = \frac{s \cdot g \land \neg t}{g}, \delta_2 = \frac{w \cdot t \land \neg l}{t}, \delta_3 = \frac{v \cdot l \land \neg g}{l} \right\} \]

Take \((\delta_2, \delta_1, \delta_3)\).
\[ A = \{s, w, v\}, \Delta = \left\{ \delta_1 = \frac{s}{g} : g \land \neg t, \delta_2 = \frac{w}{t} : t \land \neg l, \delta_3 = \frac{v}{l} : l \land \neg g \right\} \].

Take \((\delta_2, \delta_1, \delta_3)\).

\[ \delta_2 \] is applicable, so

\[ M_1 = \left\{ \begin{array}{l}
\{s, w, v, t, g, l\} \\
\{s, w, v, t, g, \neg l\} \\
\{s, w, v, t, \neg g, l\} \\
\{s, w, v, t, \neg g, \neg l\}
\end{array} \right\}, \quad J_1 = \{t \land \neg l\}, \quad B' = B_1 = Th(A \cup \{t\}). \]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \; \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g}, \; \delta_2 = \frac{w : t \land \neg l}{t}, \; \delta_3 = \frac{v : l \land \neg g}{l} \right\}. \]

Take \((\delta_2, \delta_1, \delta_3)\).

- \(\delta_2\) is applicable, so

\[
M_1 = \left\{ \begin{array}{l}
\{s, w, v, t, g, l\} \\
\{s, w, v, t, g, \neg l\} \\
\{s, w, v, t, \neg g, l\} \\
\{s, w, v, t, \neg g, \neg l\} \\
\end{array} \right\}, \quad J_1 = \{t \land \neg l\}, \; B' = B_1 = Th(A \cup \{t\}).
\]

- \(\delta_1\) is not applicable \((g \land \neg t \text{ contradicts } B_1)\), so

\[
M_2 = M_1, \; J_2 = J_1, \; B_2 = B_1.
\]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g}, \delta_2 = \frac{w : t \land \neg l}{t}, \delta_3 = \frac{v : l \land \neg g}{l} \right\}. \]

Take \((\delta_2, \delta_1, \delta_3)\).

- \(\delta_2\) is applicable, so

\[
M_1 = \left\{ \begin{array}{c}
\{s, w, v, t, g, l\} \\
\{s, w, v, t, g, \neg l\} \\
\{s, w, v, t, \neg g, l\} \\
\{s, w, v, t, \neg g, \neg l\}
\end{array} \right\}, \quad J_1 = \{t \land \neg l\}, \quad B' = B_1 = Th(A \cup \{t\}).
\]

- \(\delta_1\) is not applicable \((g \land \neg t\) contradicts \(B_1)\), so

\[ M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1. \]

- \(\delta_3\) is not applicable \((l\) contradicts the justification of \(\delta_2\)), so

\[ M_3 = M_2, \quad J_3 = J_2, \quad B_3 = B_2. \]
Example 13.10 (cont.)

\[ A = \{s, w, v\} , \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g} , \delta_2 = \frac{w : t \land \neg l}{t} , \delta_3 = \frac{v : l \land \neg g}{l} \right\} . \]

Take \((\delta_2, \delta_1, \delta_3)\).

\(\delta_2\) is applicable, so

\[ M_1 = \left\{ \begin{array}{l}
\{s, w, v, t, g, l\} \\
\{s, w, v, t, g, \neg l\} \\
\{s, w, v, t, \neg g, l\} \\
\{s, w, v, t, \neg g, \neg l\}
\end{array} \right\} , \quad J_1 = \{t \land \neg l\} , \quad B' = B_1 = Th(A \cup \{t\}) . \]

\(\delta_1\) is not applicable \((g \land \neg t\) contradicts \(B_1\)), so

\[ M_2 = M_1 , \quad J_2 = J_1 , \quad B_2 = B_1 . \]

\(\delta_3\) is not applicable \((l\) contradicts the justification of \(\delta_2\)), so

\[ M_3 = M_2 , \quad J_3 = J_2 , \quad B_3 = B_2 . \]

Hence \(B'\) is another alternative extension of \(T\).
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s \land \neg t}{g}, \quad \delta_2 = \frac{w \land \neg l}{t}, \quad \delta_3 = \frac{v \land \neg g}{l} \right\}. \]

Take \((\delta_3, \delta_1, \delta_2)\).
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g}, \delta_2 = \frac{w : t \land \neg l}{t}, \delta_3 = \frac{v : l \land \neg g}{l} \right\}. \]

Take \((\delta_3, \delta_1, \delta_2)\).

\[ \delta_3 \text{ is applicable, so } \]

\[ M_1 = \left\{ \begin{array}{l} \{s, w, v, l, g, t\} \\ \{s, w, v, l, g, \neg t\} \\ \{s, w, v, l, \neg g, t\} \\ \{s, w, v, l, \neg g, \neg t\} \end{array} \right\}, \quad J_1 = \{l \land \neg g\}, \quad B'' = B_1 = Th(A \cup \{l\}). \]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \Delta = \left\{ \delta_1 = \frac{s: g \land \neg t}{g}, \delta_2 = \frac{w: t \land \neg l}{t}, \delta_3 = \frac{v: l \land \neg g}{l} \right\}. \]

Take \((\delta_3, \delta_1, \delta_2)\).

- \(\delta_3\) is applicable, so

  \[ M_1 = \begin{cases} 
  \{s, w, v, l, g, t\} \\
  \{s, w, v, l, g, \neg t\} \\
  \{s, w, v, l, \neg g, t\} \\
  \{s, w, v, l, \neg g, \neg t\} 
  \end{cases}, \quad J_1 = \{l \land \neg g\}, \quad B'' = B_1 = \text{Th}(A \cup \{l\}). \]

- \(\delta_1\) is not applicable (\(g\) contradicts the justification of \(\delta_3\)), so

  \[ M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1. \]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g}, \delta_2 = \frac{w : t \land \neg l}{t}, \delta_3 = \frac{v : l \land \neg g}{l} \right\}. \]

Take \((\delta_3, \delta_1, \delta_2)\).

- \(\delta_3\) is applicable, so

\[
M_1 = \left\{ \begin{array}{l}
\{s, w, v, l, g, t\} \\
\{s, w, v, l, g, \neg t\} \\
\{s, w, v, l, \neg g, t\} \\
\{s, w, v, l, \neg g, \neg t\}
\end{array} \right\}, \quad J_1 = \{l \land \neg g\}, \quad B'' = B_1 = Th(A \cup \{l\}).
\]

- \(\delta_1\) is not applicable (\(g\) contradicts the justification of \(\delta_3\)), so

\[
M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1.
\]

- \(\delta_2\) is not applicable (as in Reiter’s criterion), so

\[
M_3 = M_2, \quad J_3 = J_2, \quad B_3 = B_2.
\]
Example 13.10 (cont.)

\[ A = \{s, w, v\}, \quad \Delta = \left\{ \delta_1 = \frac{s: g \wedge \neg t}{g}, \delta_2 = \frac{w: t \wedge \neg l}{t}, \delta_3 = \frac{v: l \wedge \neg g}{l} \right\}. \]

Take \((\delta_3, \delta_1, \delta_2)\).

- \(\delta_3\) is applicable, so

\[ M_1 = \begin{cases} 
\{s, w, v, l, g, t\} \\
\{s, w, v, l, g, \neg t\} \\
\{s, w, v, l, \neg g, t\} \\
\{s, w, v, l, \neg g, \neg t\} 
\end{cases}, \quad J_1 = \{l \wedge \neg g\}, \quad B'' = B_1 = Th(A \cup \{l\}). \]

- \(\delta_1\) is not applicable (\(g\) contradicts the justification of \(\delta_3\)), so

\[ M_2 = M_1, \quad J_2 = J_1, \quad B_2 = B_1. \]

- \(\delta_2\) is not applicable (as in Reiter’s criterion), so

\[ M_3 = M_2, \quad J_3 = J_2, \quad B_3 = B_2. \]

Hence \(B''\) is yet another alternative extension of \(T\).
For the remaining sequences we obtain the same alternative extensions.
Example 13.10 (cont.)

- For the remaining sequences we obtain the same alternative extensions.
- $T$ has then three extensions:
For the remaining sequences we obtain the same alternative extensions.

$T$ has then three extensions:

$E_1 = Th(A \cup \{g\})$
For the remaining sequences we obtain the same alternative extensions.

\( T \) has then three extensions:

- \( E_1 = Th(A \cup \{g\}) \)
- \( E_2 = Th(A \cup \{t\}) \)
For the remaining sequences we obtain the same alternative extensions.

$T$ has then three extensions:

- $E_1 = Th(A \cup \{g\})$
- $E_2 = Th(A \cup \{t\})$
- $E_3 = Th(A \cup \{l\})$. 
Comparison with Reiter’s DL

\[ A = \{ s, w, v \} , \Delta = \left\{ \delta_1 = \frac{s : g \land \neg t}{g} , \delta_2 = \frac{w : t \land \neg l}{t} , \delta_3 = \frac{v : l \land \neg g}{l} \right\} . \]

Recall that \( T \) has no extension in RDL!
Example 13.11

Alternative extensions need not be maximal set of sentences. For example, consider the following default theory:

\[ A = \{p\}, \quad \Delta = \left\{ \frac{r}{p}, \frac{q}{\neg r} \right\}. \]

One can easily check that \( T \) has two extensions:

\[ E_1 = Th(\{p\}), \quad E_2 = Th(\{p, \neg r\}) \]

wrt \( F_1 = \{r\} \) and \( F_2 = \{q\} \), respectively.
Example 13.11

Alternative extensions need not be maximal set of sentences. For example, consider the following default theory:

\[ A = \{p\}, \quad \Delta = \left\{ \frac{\vdash r}{p}, \frac{\vdash q}{\neg r} \right\}. \]

One can easily check that \( T \) has two extensions:

\[ E_1 = Th(\{p\}), \quad E_2 = Th(\{p, \neg r\}) \]

wrt \( F_1 = \{r\} \) and \( F_2 = \{q\} \), respectively.

Non–maximality of alternative extensions is the result of applying a default which consequent is already believed, but which justification blocks the application of other defaults.
The new approach does not model behavior of ideally rational agent — an unwise agent can accept a smaller (alternative) extension, a more rational agent will choose a broader one.
Non–rationality of ADL?

- The new approach does not model behavior of ideally rational agent — an unwise agent can accept a smaller (alternative) extension, a more rational agent will choose a broader one.

- This, however, is not an important problem, since we are actually interested in the following problem:
The new approach does not model behavior of ideally rational agent — an unwise agent can accept a smaller (alternative) extension, a more rational agent will choose a broader one.

This, however, is not an important problem, since we are actually interested in the following problem:

Given a default theory and a set of sentences determine whether all these sentences can be simultaneously believed, i.e., whether this is a subset of some (alternative) extension.
Properties of ADL

**Existence of Extensions**

Every default theory (closed and open) has an alternative extension.
Properties of ADL

- **Existence of Extensions**
  Every default theory (closed and open) has an alternative extension.

- **Weak Maximality of Extensions**
  Let $T = (A, \Delta)$ be a closed default theory and let $E$ and $E'$ be alternative extensions of $T$ wrt $F$ and $F'$, respectively, such that $E \subseteq E'$ and $F \subseteq F'$. Then $E = E'$ and $F = F'$. 
Properties of ADL

- **Existence of Extensions**
  Every default theory (closed and open) has an alternative extension.

- **Weak Maximality of Extensions**
  Let $T = (A, \Delta)$ be a closed default theory and let $E$ and $E'$ be alternative extensions of $T$ wrt $F$ and $F'$, respectively, such that $E \subseteq E'$ and $F \subseteq F'$. Then $E = E'$ and $F = F'$.

- **Semi–monotonicity**
  Let $\Delta_1$ and $\Delta_2$ be two sets of closed defaults such that $\Delta_1 \subseteq \Delta_2$ and let $E_1$ be an alternative extension of $T_1 = (A, \Delta_1)$ wrt $F_1$. Then $T_2 = (A, \Delta_2)$ has an alternative extension $E_2$ wrt $F_2$ such that $E_1 \subseteq E_2$ and $F_1 \subseteq F_2$. 
Properties of ADL (cont.)

- Relationships between extensions and alternative extensions
Properties of ADL (cont.)

- Relationships between extensions and alternative extensions
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
Properties of ADL (cont.)

- **Relationships between extensions and alternative extensions**
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
  - Let $T = (A, \Delta)$ be a closed normal default theory and let $E$ be its alternative extension. Then $E$ is its extension.
Properties of ADL (cont.)

- Relationships between extensions and alternative extensions
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
  - Let $T = (A, \Delta)$ be a closed normal default theory and let $E$ be its alternative extension. Then $E$ is its extension.
  - A default theory (closed or open) $T = (A, \Delta)$ has an inconsistent alternative extension iff $A$ is inconsistent.
Properties of ADL (cont.)

- Relationships between extensions and alternative extensions
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
  - Let $T = (A, \Delta)$ be a closed normal default theory and let $E$ be its alternative extension. Then $E$ is its extension.
  - A default theory (closed or open) $T = (A, \Delta)$ has an inconsistent alternative extension iff $A$ is inconsistent.
  - If a default theory $T = (A, \Delta)$ has an inconsistent alternative extension, then it is its only alternative extension.
Properties of ADL (cont.)

- **Relationships between extensions and alternative extensions**
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
  - Let $T = (A, \Delta)$ be a closed normal default theory and let $E$ be its alternative extension. Then $E$ is its extension.
  - A default theory (closed or open) $T = (A, \Delta)$ has an inconsistent alternative extension iff $A$ is inconsistent.
  - If a default theory $T = (A, \Delta)$ has an inconsistent alternative extension, then it is its only alternative extension.
  - The set of justifications for the inconsistent alternative extension is the empty set.
Properties of ADL (cont.)

- **Relationships between extensions and alternative extensions**
  - Let $T = (A, \Delta)$ be a closed default theory and let $E$ be its extension. Then $E$ is its alternative extension.
  - Let $T = (A, \Delta)$ be a closed normal default theory and let $E$ be its alternative extension. Then $E$ is its extension.

A default theory (closed or open) $T = (A, \Delta)$ has an inconsistent alternative extension iff $A$ is inconsistent.

If a default theory $T = (A, \Delta)$ has an inconsistent alternative extension, then it is its only alternative extension.

The set of justifications for the inconsistent alternative extension is the empty set.

If $E$ is an alternative extension of a closed default theory $T$ wrt $F$, then for every sentence $\varphi \in F$ it holds $E \not\models \neg \varphi$. 
Thank you for your attention!

Any questions are welcome.