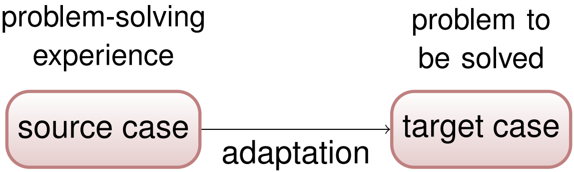
An Algorithm for Adapting Cases Represented in an Expressive Description Logic

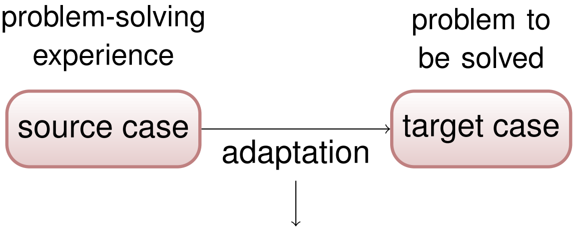
JULIEN COJAN AND JEAN LIEBER

UNIVERSITÉ HENRI POINCARÉ NANCY 1,
ORPAILLEUR,
LORIA (CNRS, INRIA, NANCY UNIVERSITY)

ICCBR 2010

Introduction





consistency test

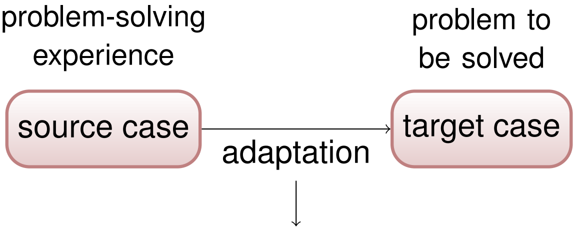
➤ tableaux method

Propositional Logic (PL)

Fragments of First Order Logic (FOL)

➥ $\mathcal{ALC}$ (extends PL, is a FOL fragment)

⋮



consistency test and revision

➤ tableaux method + extension

- Propositional Logic (PL)
- Fragments of First Order Logic (FOL)
  - ➥ $\mathcal{ALC}$ (extends PL, is a FOL fragment)
  - ⋮

Adaptation in CBR

Cases

- A case describes an experience
  (in general, a problem-solving experience)

Example: a cooking recipe

`Source`: a starter dish with raw carrots and vinaigrette

DK: Domain Knowledge

- General knowledge about the domain of application
- Knowledge in complement of the cases

Example

- Ingredient classes: `vegetable`, `root`
- The roots considered are: `carrot`, `parsnip`, and `celery`.
- Parsnips, carrots, and celeries can be grated.
- Raw parsnip are not edible.

Reasoning objective: making the target case precise

`Target` : case with an incomplete description
(The "solution part" is missing.)

Example

`Target`: I want a starter without carrots.

Reasoning objective: making the target case precise

`Target` : case with an incomplete description
(The “solution part” is missing.)

Example

`Target`: I want a starter without carrots.

Adaptation principle:

Reusing the source case to solve the target case (i.e., making it precise)

Reasoning objective: making the target case precise

`Target` : case with an incomplete description
(The "solution part" is missing.)

Example

`Target`: I want a starter without carrots.

Adaptation principle:

Reusing the source case to solve the target case (i.e., making it precise)

The inconsistencies must be dealt with.

Example: `Source` is inconsistent with `Target`

Carrots are inconsistent with the target case.

Adaptation

So, we need

- A mean for detecting inconsistencies
- a mean to solve it

Adaptation

So, we need

- A mean for detecting inconsistencies
  - $\rightarrow$ tableaux algorithm
- a mean to solve it

Adaptation

So, we need

- A mean for detecting inconsistencies
  - $\rightarrow$ tableaux algorithm
- a mean to solve it
  - $\rightarrow$ extension of the tableaux algorithm

Adaptation

So, we need

- A mean for detecting inconsistencies
  - → tableaux algorithm
- a mean to solve it
  - → extension of the tableaux algorithm

Example of an adaptation of `Source` to `Target`:

`CompletedTarget`: starter obtained by substituting in `Source` carrots with celeries

Representation formalism: $\mathcal{ALC}$

- $\mathcal{ALC}$ is a description logic (DL)

Representation formalism: $\mathcal{ALC}$

- $\mathcal{ALC}$ is a description logic (DL)
- It extends PL
  (i.e., it is an *expressive* DL)

Representation formalism: $\mathcal{ALC}$

- $\mathcal{ALC}$ is a description logic (DL)
- It extends PL
  (i.e., it is an *expressive* DL)
- It is a fragment of FOL (up to the syntax)

Representation formalism: $\mathcal{ALC}$

- $\mathcal{ALC}$ is a description logic (DL)
- It extends PL
  (i.e., it is an *expressive* DL)
- It is a fragment of FOL (up to the syntax)
- In this talk: use of the well-known syntax of FOL

Representation formalism: $\mathcal{ALC}$

- $\mathcal{ALC}$ is a description logic (DL)
- It extends PL
  (i.e., it is an *expressive* DL)
- It is a fragment of FOL (up to the syntax)
- In this talk: use of the well-known syntax of FOL
- In the paper: use of the $\mathcal{ALC}$ syntax with technical details on the algorithm

Representation of DK, Source, and Target

In $\mathcal{ALC}$ within FOL syntax:

$$\begin{array}{rll} \mathtt{DK} = & & \forall x\ \mathtt{gratedRawCarrot}(x) \Leftrightarrow \mathtt{carrot}(x) \wedge \mathtt{raw}(x) \wedge \mathtt{grated}(x) \\ & \wedge & \forall x\ \mathtt{root}(x) \Leftrightarrow \mathtt{carrot}(x) \vee \mathtt{parsnip}(x) \vee \mathtt{celery}(x) \\ & \wedge & \forall x\ \mathtt{parsnip}(x) \Rightarrow \neg \mathtt{raw}(x) \end{array}$$

$$\begin{array}{rll} \mathtt{Source}(\sigma) = & & \mathtt{starter}(\sigma) \wedge \exists y\ \mathtt{ing}(\sigma, y) \wedge \mathtt{gratedRawCarrot}(y) \\ & \wedge & \exists y\ \mathtt{ing}(\sigma, y) \wedge \mathtt{vinaigrette}(y) \end{array}$$

$$\mathtt{Target}(\theta) = \mathtt{starter}(\theta) \wedge \neg(\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{carrot}(y))$$

Representation of DK, Source, and Target

In $\mathcal{ALC}$ under NNF within FOL syntax:

$$\mathtt{DK} = \begin{array}{rl} \forall x & \neg\mathtt{gratedRawCarrot}(x) \vee (\mathtt{carrot}(x) \wedge \mathtt{raw}(x) \wedge \mathtt{grated}(x)) \\ \wedge & \mathtt{gratedRawCarrot}(x) \vee \neg\mathtt{carrot}(x) \vee \neg\mathtt{raw}(x) \vee \neg\mathtt{grated}(x) \\ \wedge & \neg\mathtt{root}(x) \vee \mathtt{carrot}(x) \vee \mathtt{parsnip}(x) \vee \mathtt{celery}(x) \\ \wedge & \mathtt{root}(x) \vee (\neg\mathtt{carrot}(x) \wedge \neg\mathtt{parsnip}(x) \wedge \neg\mathtt{celery}(x)) \\ \wedge & \neg\mathtt{parsnip}(x) \vee \neg\mathtt{raw}(x) \end{array}$$

$$\mathtt{Source}(\sigma) = \begin{array}{rl} & \mathtt{starter}(\sigma) \wedge \exists y\ \mathtt{ing}(\sigma, y) \wedge \mathtt{gratedRawCarrot}(y) \\ \wedge & \exists y\ \mathtt{ing}(\sigma, y) \wedge \mathtt{vinaigrette}(y) \end{array}$$

$$\mathtt{Target}(\theta) = \mathtt{starter}(\theta) \wedge \forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$$

Adaptation (reminder)

So, we need

- A mean for detecting inconsistencies
  - $\rightarrow$ tableaux algorithm
- a mean to solve it
  - $\rightarrow$ extension of the tableaux algorithm

The tableau method

The tableau method (1/3)

- A classical deductive method in PL and on decidable fragments of FOL
- Input: a knowledge base $\mathtt{KB}$
(a formula or a set of formulas interpreted conjunctively)

$\rightarrow$ Objective: determining whether $\mathtt{KB}$ is consistent is not

Example: $\mathtt{Source}(\theta)$ is in contradiction with $\mathtt{Target}(\theta)$, given $\mathtt{DK}$

$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$ is inconsistent

The tableau method (2/3)

- Principle:
  - Some formalism-dependent transformation rules are applied on formulas to produce new (deduced) formulas, whenever it is possible.
  - `KB` is inconsistent *iff* there is a clash in every branch.

The tableau method (3/3) Ex. on $\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$

$$\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$$

The tableau method (3/3) Ex. on $\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$

$$\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$$

$$\texttt{DK}$$

$$\texttt{Source}(\theta)$$

$$\texttt{Target}(\theta)$$

The tableau method (3/3) Ex. on $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$$

$$\mathtt{DK}$$

$$\mathtt{Source}(\theta)$$

$$\mathtt{Target}(\theta)$$

$$\mathtt{starter}(\theta)$$

$$\exists y \ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$$

$$\exists y \ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$$

$\mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{vinaigrette}(y)$

$\forall y\ (\neg \mathtt{ing}(\theta, y) \lor \neg \mathtt{carrot}(y))$

$$\mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$$

$$\mathtt{DK}$$

$$\mathtt{Source}(\theta)$$

$$\mathtt{Target}(\theta)$$

$$\mathtt{starter}(\theta)$$

$$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{gratedRawCarrot}(y)$$

$$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{vinaigrette}(y)$$

$$\forall y\ (\neg\mathtt{ing}(\theta, y) \lor \neg\mathtt{carrot}(y))$$

$$\mathtt{ing}(\theta, a) \land \mathtt{gratedRawCarrot}(a)$$

$$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$$

$$\mathtt{DK}$$

$$\mathtt{Source}(\theta)$$

$$\mathtt{Target}(\theta)$$

$$\mathtt{starter}(\theta)$$

$$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$$

$$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$$

$$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$$

$$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$$

$$\mathtt{ing}(\theta, a)$$

$$\mathtt{gratedRawCarrot}(a)$$

The tableau method (3/3) Ex. on $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)) \ldots$

The tableau method (3/3) Ex. on $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\forall y\ (\neg \mathtt{ing}(\theta, y) \vee \neg \mathtt{carrot}(y))$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

$\neg \mathtt{gratedRawCarrot}(a) \vee (\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)) \ldots$

$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$

$\neg \mathtt{gratedRawCarrot}(a)$

$\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\forall y\ (\neg \mathtt{ing}(\theta, y) \vee \neg \mathtt{carrot}(y))$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

$\neg \mathtt{gratedRawCarrot}(a) \vee (\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)) \ldots$

$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$

$\neg \mathtt{gratedRawCarrot}(a)$

$\square$

The tableau method (3/3) Ex. on $\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$

$$\forall y\ (\neg\texttt{ing}(\theta, y) \vee \neg\texttt{carrot}(y))$$

$$\texttt{ing}(\theta, a) \wedge \texttt{gratedRawCarrot}(a)$$

$$\texttt{ing}(\theta, a)$$

$$\texttt{gratedRawCarrot}(a)$$

$$\neg\texttt{gratedRawCarrot}(a) \vee (\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a))\ldots$$

$$\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)$$

$$\neg\texttt{gratedRawCarrot}(a)$$

$$\square$$

$$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$$

$$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$$

$$\mathtt{ing}(\theta, a)$$

$$\mathtt{gratedRawCarrot}(a)$$

$$\neg\mathtt{gratedRawCarrot}(a) \vee (\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a))\ldots$$

$$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$$

$$\mathtt{carrot}(a)$$

$$\mathtt{raw}(a)$$

$$\mathtt{grated}(a)$$

$$\neg\mathtt{gratedRawCarrot}(a)$$

$$\square$$

The tableau method (3/3) Ex. on $\texttt{DK} \land \texttt{Source}(\theta) \land \texttt{Target}(\theta)$

$$\forall y\ (\neg \texttt{ing}(\theta, y) \lor \neg \texttt{carrot}(y))$$

$$\texttt{ing}(\theta, a) \land \texttt{gratedRawCarrot}(a)$$

$$\texttt{ing}(\theta, a)$$

$$\texttt{gratedRawCarrot}(a)$$

$$\neg \texttt{gratedRawCarrot}(a) \lor (\texttt{carrot}(a) \land \texttt{raw}(a) \land \texttt{grated}(a)) \ldots$$

$$\texttt{carrot}(a) \land \texttt{raw}(a) \land \texttt{grated}(a)$$

$$\neg \texttt{gratedRawCarrot}(a)$$

$$\square$$

$$\texttt{carrot}(a)$$

$$\texttt{raw}(a)$$

$$\texttt{grated}(a)$$

$$\neg \texttt{ing}(\theta, a) \lor \neg \texttt{carrot}(a)$$

$$\forall y\ (\neg\texttt{ing}(\theta, y) \vee \neg\texttt{carrot}(y))$$

$$\texttt{ing}(\theta, a) \wedge \texttt{gratedRawCarrot}(a)$$

$$\texttt{ing}(\theta, a)$$

$$\texttt{gratedRawCarrot}(a)$$

$$\neg\texttt{gratedRawCarrot}(a) \vee (\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)) \ldots$$

$$\neg\texttt{gratedRawCarrot}(a) \qquad \texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)$$

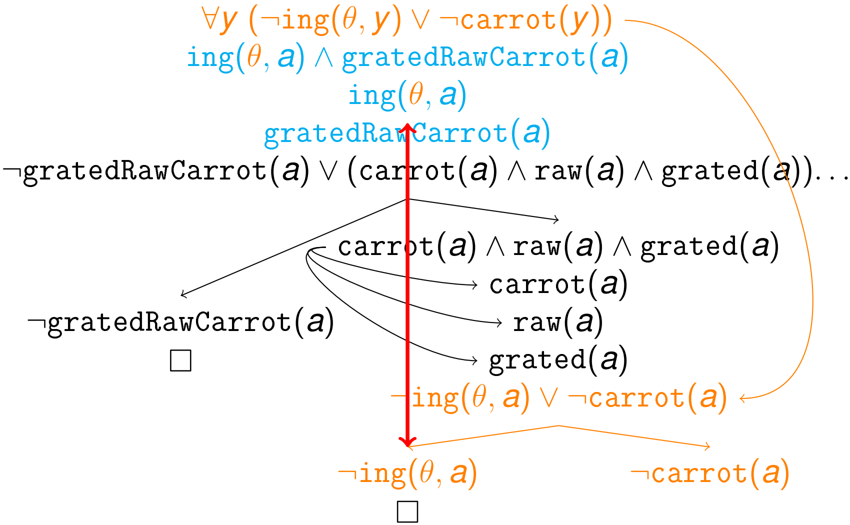
$$\square \qquad \texttt{carrot}(a)$$

$$\texttt{raw}(a)$$

$$\texttt{grated}(a)$$

$$\neg\texttt{ing}(\theta, a) \vee \neg\texttt{carrot}(a)$$

$$\neg\texttt{ing}(\theta, a) \qquad \neg\texttt{carrot}(a)$$


∀y (¬ing(θ, y) ∨ ¬carrot(y))
ing(θ, a) ∧ gratedRawCarrot(a)
ing(θ, a)
gratedRawCarrot(a)
¬gratedRawCarrot(a) ∨ (carrot(a) ∧ raw(a) ∧ grated(a))...
carrot(a) ∧ raw(a) ∧ grated(a)
carrot(a)
raw(a)
grated(a)
¬gratedRawCarrot(a)
□
¬ing(θ, a) ∨ ¬carrot(a)
¬ing(θ, a)
¬carrot(a)
□

The tableau method (3/3) Ex. on $\mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$

$$\forall y\ (\neg \mathtt{ing}(\theta, y) \lor \neg \mathtt{carrot}(y))$$
$$\mathtt{ing}(\theta, a) \land \mathtt{gratedRawCarrot}(a)$$
$$\mathtt{ing}(\theta, a)$$
$$\mathtt{gratedRawCarrot}(a)$$
$$\neg \mathtt{gratedRawCarrot}(a) \lor (\mathtt{carrot}(a) \land \mathtt{raw}(a) \land \mathtt{grated}(a))\ldots$$

$$\neg \mathtt{gratedRawCarrot}(a)$$
$$\square$$

$$\mathtt{carrot}(a) \land \mathtt{raw}(a) \land \mathtt{grated}(a)$$
$$\mathtt{carrot}(a)$$
$$\mathtt{raw}(a)$$
$$\mathtt{grated}(a)$$
$$\neg \mathtt{ing}(\theta, a) \lor \neg \mathtt{carrot}(a)$$

$$\neg \mathtt{ing}(\theta, a)$$
$$\square$$

$$\neg \mathtt{carrot}(a)$$
$$\square$$

$$\forall y\ (\neg \mathtt{ing}(\theta, y) \vee \neg \mathtt{carrot}(y))$$
$$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$$
$$\mathtt{ing}(\theta, a)$$
$$\mathtt{gratedRawCarrot}(a)$$
$$\neg \mathtt{gratedRawCarrot}(a) \vee (\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)) \ldots$$

$$\neg \mathtt{gratedRawCarrot}(a)$$
$$\Box$$

$$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$$
$$\mathtt{carrot}(a)$$
$$\mathtt{raw}(a)$$
$$\mathtt{grated}(a)$$
$$\neg \mathtt{ing}(\theta, a) \vee \neg \mathtt{carrot}(a)$$

$$\neg \mathtt{ing}(\theta, a)$$
$$\Box$$

$$\neg \mathtt{carrot}(a)$$
$$\Box$$

Adaptation by reestablishing consistency

Adaptation by correction of $\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$

- Pretend that $\texttt{Source}(\sigma)$ solves $\texttt{Target}(\theta)$
  - $\texttt{Source}(\theta)$

Adaptation by correction of $\mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$:
  $\mathtt{CompletedTarget} = \mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$

Adaptation by correction of $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$:
  $\mathtt{CompletedTarget} = \mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \wedge \mathtt{Source}(\theta)$

Adaptation by correction of $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$:
  $\mathtt{CompletedTarget} = \mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
- Apply tableau method

Adaptation by correction of $\texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$

- Pretend that $\texttt{Source}(\sigma)$ solves $\texttt{Target}(\theta)$
  - $\texttt{Source}(\theta)$
- If $\texttt{Source}(\theta)$ is consistent with $\texttt{Target}(\theta)$: $\texttt{CompletedTarget} = \texttt{DK} \wedge \texttt{Source}(\theta) \wedge \texttt{Target}(\theta)$
- Else, minimal change on $\texttt{Source}(\theta)$: keep as much as possible from the consequences of $\texttt{DK} \wedge \texttt{Source}(\theta)$
- Apply tableau method
  - To generate the consistent branches S_i from $\texttt{DK} \wedge \texttt{Source}(\theta)$

Adaptation by correction of $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$: $\mathtt{CompletedTarget} = \mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
- Apply tableau method
  - To generate the consistent branches S_i from $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
  - To generate the consistent branches T_j from $\mathtt{DK} \wedge \mathtt{Target}(\theta)$

Adaptation by correction of $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - ➤ $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$: $\mathtt{CompletedTarget} = \mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
- Apply tableau method
  - To generate the consistent branches S_i from $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
  - To generate the consistent branches T_j from $\mathtt{DK} \wedge \mathtt{Target}(\theta)$
  - To generate the (inconsistent) branches B_{ij} from each $S_i \wedge T_j$

Adaptation by correction of $\mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$: $\mathtt{CompletedTarget} = \mathtt{DK} \wedge \mathtt{Source}(\theta) \wedge \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
- Apply tableau method
  - To generate the consistent branches S_i from $\mathtt{DK} \wedge \mathtt{Source}(\theta)$
  - To generate the consistent branches T_j from $\mathtt{DK} \wedge \mathtt{Target}(\theta)$
  - To generate the (inconsistent) branches B_{ij} from each $S_i \wedge T_j$
- Repair (some of) the B_{ij}'s

Adaptation by correction of $\mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$

- Pretend that $\mathtt{Source}(\sigma)$ solves $\mathtt{Target}(\theta)$
  - ➤ $\mathtt{Source}(\theta)$
- If $\mathtt{Source}(\theta)$ is consistent with $\mathtt{Target}(\theta)$: $\mathtt{CompletedTarget} = \mathtt{DK} \land \mathtt{Source}(\theta) \land \mathtt{Target}(\theta)$
- Else, minimal change on $\mathtt{Source}(\theta)$: keep as much as possible from the consequences of $\mathtt{DK} \land \mathtt{Source}(\theta)$
- Apply tableau method
  - To generate the consistent branches S_i from $\mathtt{DK} \land \mathtt{Source}(\theta)$
  - To generate the consistent branches T_j from $\mathtt{DK} \land \mathtt{Target}(\theta)$
  - To generate the (inconsistent) branches B_{ij} from each $S_i \land T_j$
- Repair (some of) the B_{ij}'s
- Use of a cost function to give a priority in the repairs

$$\texttt{DK} \wedge \texttt{Source}(\theta)$$

$$\texttt{DK}$$

$$\texttt{Source}(\theta)$$

$$\texttt{starter}(\theta)$$

$$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{gratedRawCarrot}(y)$$

$$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{vinaigrette}(y)$$

$$\texttt{ing}(\theta, a) \wedge \texttt{gratedRawCarrot}(a)$$

$$\texttt{ing}(\theta, a)$$

$$\texttt{gratedRawCarrot}(a)$$

$$\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)$$

$$\texttt{carrot}(a)$$

$$\texttt{raw}(a)$$

$$\texttt{grated}(a)$$

$$\texttt{carrot}(a) \vee \texttt{parsnip}(a) \vee \texttt{celery}(a)$$

$$\texttt{root}(a)$$

$$\neg\texttt{parsnip}(a)\ [\ldots]$$

$$S_1$$

$$\texttt{DK} \wedge \texttt{Target}(\theta)$$
$$\texttt{DK}$$
$$\texttt{Target}(\theta)$$
$$\texttt{starter}(\theta)$$
$$\forall y\ (\neg\texttt{ing}(\theta, y) \vee \neg\texttt{carrot}(y))$$
$$\neg\texttt{ing}(\theta, a) \vee \neg\texttt{carrot}(a)$$
$$\neg\texttt{gratedRawCarrot}(a) \vee (\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a))$$
$$\texttt{gratedRawCarrot}(a) \vee \neg\texttt{carrot}(a) \vee \neg\texttt{raw}(a) \vee \neg\texttt{grated}(a)$$
$$\neg\texttt{root}(a) \vee \texttt{carrot}(a) \vee \texttt{parsnip}(a) \vee \texttt{celery}(a)$$
$$\texttt{root}(a) \vee \neg\texttt{carrot}(a) \wedge \neg\texttt{parsnip}(a) \wedge \neg\texttt{celery}(a)$$
$$\neg\texttt{parsnip}(a) \vee \neg\texttt{raw}(a)$$

T_1	T_2	…	…
$\neg\texttt{carrot}(a)$	$\neg\texttt{carrot}(a)$	$\ldots$	$\ldots$
$\texttt{celery}(a)$	$\texttt{parsnip}(a)$		
$\texttt{root}(a)$	$\texttt{root}(a)$		
$\texttt{grated}(a)$	$\texttt{grated}(a)$		
$\texttt{raw}(a)$	$\neg\texttt{raw}(a)$		
T_1	T_2		

$S_1 \wedge T_1$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$	$\mathtt{DK} \wedge \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
$\mathtt{Source}(\theta)$	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$	$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$
$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \vee \ldots$
$\mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{root}(a) \vee \ldots$
$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$	$\mathtt{root}(a) \vee \ldots$
$\mathtt{carrot}(a)$	$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{celery}(a)$
$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\mathtt{raw}(a)$

$\square \pm \mathtt{carrot}(a)$

$S_1 \wedge T_1$

$\texttt{DK} \wedge \texttt{Source}(\theta)$

$\texttt{DK}$

$\texttt{Source}(\theta)$

$\texttt{starter}(\theta)$

$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{gratedRawCarrot}(y)$

$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{vinaigrette}(y)$

$\texttt{ing}(\theta, a) \wedge \texttt{gratedRawCarrot}(a)$

$\texttt{ing}(\theta, a)$

$\texttt{gratedRawCarrot}(a)$

$\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)$

$\texttt{carrot}(a)$

$\texttt{raw}(a)$

$\texttt{grated}(a)$

$\texttt{carrot}(a) \vee \texttt{parsnip}(a) \vee \texttt{celery}(a)$

$\texttt{root}(a)$

$\neg\texttt{parsnip}(a)$

$\texttt{DK} \wedge \texttt{Target}(\theta)$

$\texttt{DK}$

$\texttt{Target}(\theta)$

$\texttt{starter}(\theta)$

$\forall y\ (\neg\texttt{ing}(\theta, y) \vee \neg\texttt{carrot}(y))$

$\neg\texttt{ing}(\theta, a) \vee \neg\texttt{carrot}(a)$

$\neg\texttt{gratedRawCarrot}(a) \vee (\ldots)$

$\texttt{gratedRawCarrot}(a) \vee \ldots$

$\neg\texttt{root}(a) \vee \ldots$

$\texttt{root}(a) \vee \ldots$

$\neg\texttt{parsnip}(a) \vee \neg\texttt{raw}(a)$

$\neg\texttt{carrot}(a)$

$\texttt{celery}(a)$

$\texttt{root}(a)$

$\texttt{grated}(a)$

$\texttt{raw}(a)$

~~$\Box \bot \texttt{carrot}(a)$~~

$S_1 \wedge T_1$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$	$\mathtt{DK} \wedge \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
$\mathtt{Source}(\theta)$	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$	$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$
$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{gratedRawCarrot}(a) \vee (\dots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \vee \dots$
$\mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{root}(a) \vee \dots$
$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$	$\mathtt{root}(a) \vee \dots$
~~$\mathtt{carrot}(a)$~~	$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{celery}(a)$
$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\mathtt{raw}(a)$

~~$\square \bot \mathtt{carrot}(a)$~~

$S_1 \wedge T_1$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{celery}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\mathtt{raw}(a)$

~~$\square \bot \mathtt{carrot}(a)$~~

$S_1 \wedge T_1$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

~~$\mathtt{gratedRawCarrot}(a)$~~

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{celery}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\mathtt{raw}(a)$

~~$\square\ \bot\ \mathtt{carrot}(a)$~~

$S_1 \wedge T_1$

$\texttt{DK} \wedge \texttt{Source}(\theta)$

$\texttt{DK}$

$\texttt{Source}(\theta)$

$\texttt{starter}(\theta)$

$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{gratedRawCarrot}(y)$

$\exists y\ \texttt{ing}(\theta, y) \wedge \texttt{vinaigrette}(y)$

~~$\texttt{ing}(\theta, a) \wedge \texttt{gratedRawCarrot}(a)$~~

$\texttt{ing}(\theta, a)$

~~$\texttt{gratedRawCarrot}(a)$~~

~~$\texttt{carrot}(a) \wedge \texttt{raw}(a) \wedge \texttt{grated}(a)$~~

~~$\texttt{carrot}(a)$~~

$\texttt{raw}(a)$

$\texttt{grated}(a)$

$\texttt{carrot}(a) \vee \texttt{parsnip}(a) \vee \texttt{celery}(a)$

$\texttt{root}(a)$

$\neg\texttt{parsnip}(a)$

$\texttt{DK} \wedge \texttt{Target}(\theta)$

$\texttt{DK}$

$\texttt{Target}(\theta)$

$\texttt{starter}(\theta)$

$\forall y\ (\neg\texttt{ing}(\theta, y) \vee \neg\texttt{carrot}(y))$

$\neg\texttt{ing}(\theta, a) \vee \neg\texttt{carrot}(a)$

$\neg\texttt{gratedRawCarrot}(a) \vee (\dots)$

$\texttt{gratedRawCarrot}(a) \vee \dots$

$\neg\texttt{root}(a) \vee \dots$

$\texttt{root}(a) \vee \dots$

$\neg\texttt{parsnip}(a) \vee \neg\texttt{raw}(a)$

$\neg\texttt{carrot}(a)$

$\texttt{celery}(a)$

$\texttt{root}(a)$

$\texttt{grated}(a)$

$\texttt{raw}(a)$

~~$\square \bot \texttt{carrot}(a)$~~

$S_1 \land T_1$

$\mathtt{DK} \land \mathtt{Source}(\theta)$	$\mathtt{DK} \land \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
$\mathtt{Source}(\theta)$	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
~~$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{gratedRawCarrot}(y)$~~	$\forall y\ (\neg\mathtt{ing}(\theta, y) \lor \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \land \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \lor \neg\mathtt{carrot}(a)$
~~$\mathtt{ing}(\theta, a) \land \mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{gratedRawCarrot}(a) \lor (\ldots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \lor \ldots$
~~$\mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{root}(a) \lor \ldots$
~~$\mathtt{carrot}(a) \land \mathtt{raw}(a) \land \mathtt{grated}(a)$~~	$\mathtt{root}(a) \lor \ldots$
~~$\mathtt{carrot}(a)$~~	$\neg\mathtt{parsnip}(a) \lor \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{celery}(a)$
$\mathtt{carrot}(a) \lor \mathtt{parsnip}(a) \lor \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\mathtt{raw}(a)$

~~$\square \bot \mathtt{carrot}(a)$~~

$S_1 \wedge T_1$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$

$\mathtt{DK}$

~~$\mathtt{Source}(\theta)$~~

$\mathtt{starter}(\theta)$

~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

~~$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$~~

$\mathtt{ing}(\theta, a)$

~~$\mathtt{gratedRawCarrot}(a)$~~

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{celery}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\mathtt{raw}(a)$

~~$\square \pm \mathtt{carrot}(a)$~~

$S_1 \wedge T_1$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~

$\mathtt{DK}$

~~$\mathtt{Source}(\theta)$~~

$\mathtt{starter}(\theta)$

~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

~~$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$~~

$\mathtt{ing}(\theta, a)$

~~$\mathtt{gratedRawCarrot}(a)$~~

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{celery}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\mathtt{raw}(a)$

~~$\square \bot \mathtt{carrot}(a)$~~

$S'_1 \wedge T_1$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~	$\mathtt{DK} \wedge \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
~~$\mathtt{Source}(\theta)$~~	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~	$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$
~~$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \vee \ldots$
~~$\mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~	$\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a)$~~	$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{celery}(a)$
$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\mathtt{raw}(a)$

~~$\Box \bot \mathtt{carrot}(a)$~~

$$S_1 \wedge T_2$$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$

$\mathtt{carrot}(a)$

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

$\square \pm \mathtt{carrot}(a)$

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{parsnip}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\neg\mathtt{raw}(a)$

$\square \pm \mathtt{raw}(a)$

$$S_1 \wedge T_2$$

$\mathtt{DK} \wedge \mathtt{Source}(\theta)$

$\mathtt{DK}$

$\mathtt{Source}(\theta)$

$\mathtt{starter}(\theta)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

$\mathtt{gratedRawCarrot}(a)$

$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$

$\mathtt{carrot}(a)$

$\mathtt{raw}(a)$

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

~~$\Box \pm \mathtt{carrot}(a)$~~

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{parsnip}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\neg\mathtt{raw}(a)$

$\Box \pm \mathtt{raw}(a)$

$S'_1 \wedge T_2$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~	$\mathtt{DK} \wedge \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
~~$\mathtt{Source}(\theta)$~~	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~	$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$
$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \vee \ldots$
~~$\mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~	$\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a)$~~	$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{parsnip}(a)$
$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\neg\mathtt{raw}(a)$
~~$\Box\pm\mathtt{carrot}(a)$~~	$\Box\pm\mathtt{raw}(a)$

$S'_1 \wedge T_2$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~	$\mathtt{DK} \wedge \mathtt{Target}(\theta)$
$\mathtt{DK}$	$\mathtt{DK}$
~~$\mathtt{Source}(\theta)$~~	$\mathtt{Target}(\theta)$
$\mathtt{starter}(\theta)$	$\mathtt{starter}(\theta)$
~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~	$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$
$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$	$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$
$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$	$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$
$\mathtt{ing}(\theta, a)$	$\mathtt{gratedRawCarrot}(a) \vee \ldots$
~~$\mathtt{gratedRawCarrot}(a)$~~	$\neg\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~	$\mathtt{root}(a) \vee \ldots$
~~$\mathtt{carrot}(a)$~~	$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$
$\mathtt{raw}(a)$	$\neg\mathtt{carrot}(a)$
$\mathtt{grated}(a)$	$\mathtt{parsnip}(a)$
$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$	$\mathtt{root}(a)$
$\mathtt{root}(a)$	$\mathtt{grated}(a)$
$\neg\mathtt{parsnip}(a)$	$\neg\mathtt{raw}(a)$
~~$\square\pm\mathtt{carrot}(a)$~~	~~$\square\pm\mathtt{raw}(a)$~~

$S_1' \wedge T_2$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~

$\mathtt{DK}$

~~$\mathtt{Source}(\theta)$~~

$\mathtt{starter}(\theta)$

~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

~~$\mathtt{gratedRawCarrot}(a)$~~

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

~~$\mathtt{raw}(a)$~~

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

~~□±$\mathtt{carrot}(a)$~~

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{parsnip}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\neg\mathtt{raw}(a)$

~~□±$\mathtt{raw}(a)$~~

$S''_1 \wedge T_2$

~~$\mathtt{DK} \wedge \mathtt{Source}(\theta)$~~

$\mathtt{DK}$

~~$\mathtt{Source}(\theta)$~~

$\mathtt{starter}(\theta)$

~~$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{gratedRawCarrot}(y)$~~

$\exists y\ \mathtt{ing}(\theta, y) \wedge \mathtt{vinaigrette}(y)$

$\mathtt{ing}(\theta, a) \wedge \mathtt{gratedRawCarrot}(a)$

$\mathtt{ing}(\theta, a)$

~~$\mathtt{gratedRawCarrot}(a)$~~

~~$\mathtt{carrot}(a) \wedge \mathtt{raw}(a) \wedge \mathtt{grated}(a)$~~

~~$\mathtt{carrot}(a)$~~

~~$\mathtt{raw}(a)$~~

$\mathtt{grated}(a)$

$\mathtt{carrot}(a) \vee \mathtt{parsnip}(a) \vee \mathtt{celery}(a)$

$\mathtt{root}(a)$

$\neg\mathtt{parsnip}(a)$

~~□±$\mathtt{carrot}(a)$~~

$\mathtt{DK} \wedge \mathtt{Target}(\theta)$

$\mathtt{DK}$

$\mathtt{Target}(\theta)$

$\mathtt{starter}(\theta)$

$\forall y\ (\neg\mathtt{ing}(\theta, y) \vee \neg\mathtt{carrot}(y))$

$\neg\mathtt{ing}(\theta, a) \vee \neg\mathtt{carrot}(a)$

$\neg\mathtt{gratedRawCarrot}(a) \vee (\ldots)$

$\mathtt{gratedRawCarrot}(a) \vee \ldots$

$\neg\mathtt{root}(a) \vee \ldots$

$\mathtt{root}(a) \vee \ldots$

$\neg\mathtt{parsnip}(a) \vee \neg\mathtt{raw}(a)$

$\neg\mathtt{carrot}(a)$

$\mathtt{parsnip}(a)$

$\mathtt{root}(a)$

$\mathtt{grated}(a)$

$\neg\mathtt{raw}(a)$

~~□±$\mathtt{raw}(a)$~~

- $S_1 \wedge T_1$ requires less repair than $S_1 \wedge T_2$
- In fact, $S_1 \wedge T_1$ requires strictly less repair than $S_1 \wedge T_j$ ($j \neq 1$)
- Therefore:

$$\texttt{CompletedTarget} = S_1' \wedge T_1$$

- In other words:
  to adapt your raw and grated carrots with vinaigrette,
  substitute carrots with celery
  (better than parsnips, since raw parsnips cannot be eaten)

Conclusion and Future Work

Conclusion

- Extension of the tableau method to adaptation in CBR
- Studied and implemented for $\mathcal{ALC}$
  (and propositional logic which is a fragment of $\mathcal{ALC}$)
- A prototype working with $\mathcal{ALC}$ has been developed.
- Complexity $\geq$ consistency test:
  - EXPTime-hard for $\mathcal{ALC}$
  - Much quicker for practical applications

Future Work

- Deeper study of the properties of this adaptation
- Links with other approaches of adaptation
- Towards an efficient implementation
  - What are the optimizations of the tableaux method that are compatible with our extension?
- Generalisation to more expressive DLs
  - E.g., to $\mathcal{ALC}(\mathtt{D})$ whith a numerical concrete domain