Chapter 3

Basic Model Theory

Interpretations of $\mathcal{ALC}$ can be viewed as graphs (with labeled edges and nodes).

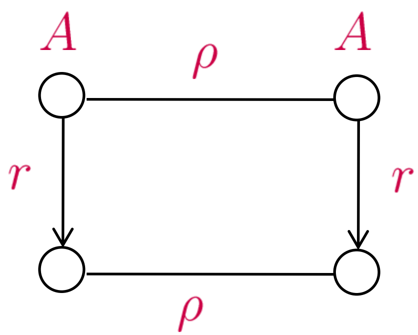
- We introduce the notion of bisimulation between graphs/interpretations
- We show that $\mathcal{ALC}$-concepts cannot distinguish bisimular nodes
- We use this to show restrictions of the expressive power of $\mathcal{ALC}$
- We use this to show interesting properties of models for $\mathcal{ALC}$:
  - tree model property
  - closure under disjoint union
- We show the finite model property of $\mathcal{ALC}$.

Definition 3.1 (bisimulation)

Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be interpretations.

The relation $\rho \subseteq \Delta^{\mathcal{I}_1} \times \Delta^{\mathcal{I}_2}$ is a bisimulation between $\mathcal{I}_1$ and $\mathcal{I}_2$ iff

- $d_1 \; \rho \; d_2$ implies $d_1 \in A^{\mathcal{I}_1}$ iff $d_2 \in A^{\mathcal{I}_2}$ for all $A \in N_C$
- $d_1 \; \rho \; d_2$ and $(d_1, d_1') \in r^{\mathcal{I}_1}$ implies the existence of $d_2' \in \Delta^{\mathcal{I}_2}$ such that $d_1' \; \rho \; d_2'$ and $(d_2, d_2') \in r^{\mathcal{I}_2}$ for all $r \in N_R$
- $d_1 \; \rho \; d_2$ and $(d_2, d_2') \in r^{\mathcal{I}_2}$ implies the existence of $d_1' \in \Delta^{\mathcal{I}_1}$ such that $d_1' \; \rho \; d_2'$ and $(d_1, d_1') \in r^{\mathcal{I}_1}$ for all $r \in N_R$



Note:

- $\mathcal{I}_1 = \mathcal{I}_2$ is possible
- the empty relation $\emptyset$ is a bisimulation.

Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be interpretations and $d_1 \in \Delta^{\mathcal{I}_1}$, $d_2 \in \Delta^{\mathcal{I}_2}$.

$(\mathcal{I}_1, d_1) \sim (\mathcal{I}_2, d_2)$ iff there is a bisimulation ρ between $\mathcal{I}_1$ and $\mathcal{I}_2$ such that $d_1 \rho d_2$

Theorem 3.2 (bisimulation invariance of $\mathcal{ALC}$)

If $(\mathcal{I}_1, d_1) \sim (\mathcal{I}_2, d_2)$, then the following holds for all $\mathcal{ALC}$-concepts C:

$$d_1 \in C^{\mathcal{I}_1} \quad \text{iff} \quad d_2 \in C^{\mathcal{I}_2}$$

"$\mathcal{ALC}$-concepts cannot distinguish between d_1 and d_2"

Proof: blackboard

Dresden

Expressive power of $\mathcal{ALC}$

We have introduced extensions of $\mathcal{ALC}$ by the concept constructors number restrictions, nominals and the role constructor inverse role.

How can we show that these constructors really extend $\mathcal{ALC}$, i.e., that they cannot be expressed using the constructors of $\mathcal{ALC}$.

To this purpose, we show that, using any of these constructors, we can construct concept descriptions

- that cannot be expressed by $\mathcal{ALC}$-concept descriptions,
- i.e, there is no equivalent $\mathcal{ALC}$-concept description.

Expressive power of $\mathcal{ALC}$

Proposition 3.3 ($\mathcal{ALCN}$ is more expressive than $\mathcal{ALC}$)

No $\mathcal{ALC}$-concept description is equivalent to the $\mathcal{ALCN}$-concept description $(\leq 1 r)$.

Proof: blackboard

Dresden

Expressive power of $\mathcal{ALC}$

Proposition 3.4 ($\mathcal{ALCI}$ is more expressive than $\mathcal{ALC}$)

> No $\mathcal{ALC}$-concept description is equivalent to
> the $\mathcal{ALCI}$-concept description $\exists r^{-1}.\top$.

Proof: blackboard

Expressive power of $\mathcal{ALC}$

Proposition 3.5 ($\mathcal{ALCO}$ is more expressive than $\mathcal{ALC}$)

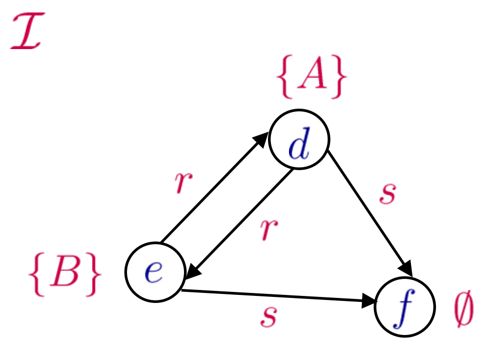
No $\mathcal{ALC}$-concept description is equivalent to the $\mathcal{ALCO}$-concept description $\{a\}$.

Proof: blackboard

Tree model property of $\mathcal{ALC}$.

Recall that interpretations can be viewed as graphs:

- nodes are the elements of $\Delta^{\mathcal{I}}$;
- interpretation of role names yields edges;
- interpretation of concept names yields node labels.



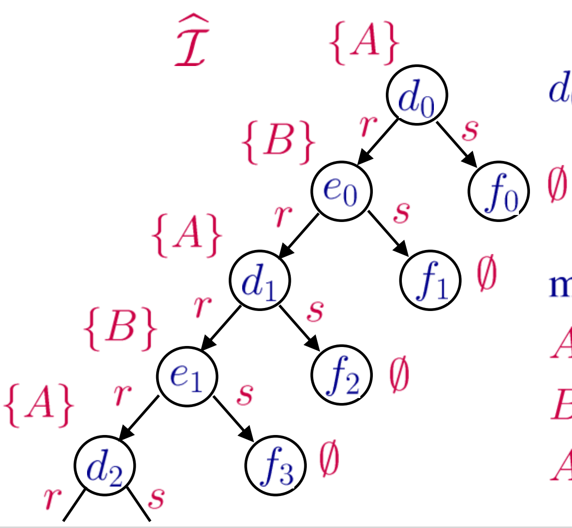
model of

$A \sqsubseteq \exists r.B$

$B \sqsubseteq \exists r.A$ $\quad d \in A^{\mathcal{I}}$

$A \sqcup B \sqsubseteq \exists s.\top$

Starting with a given node, the graph can be unraveled into a tree without "changing membership" in concepts.



$d_0 \in A^{\widehat{\mathcal{I}}}$

model of

$A \sqsubseteq \exists r.B$

$B \sqsubseteq \exists r.A$

$A \sqcup B \sqsubseteq \exists s.\top$

Definition 3.6 (tree model)

Let $\mathcal{T}$ be a TBox and C a concept description.

The interpretation $\mathcal{I}$ is a tree model of C w.r.t. $\mathcal{T}$ iff
$\mathcal{I}$ is a model of $\mathcal{T}$, and the graph

$$(\Delta^{\mathcal{I}}, \bigcup_{r \in N_R} r^{\mathcal{I}})$$

is a tree whose root belongs to $C^{\mathcal{I}}$.

Theorem 3.7 (tree model property of $\mathcal{ALC}$)

$\mathcal{ALC}$ has the tree model property,

i.e., if $\mathcal{T}$ is an $\mathcal{ALC}$-TBox and C an $\mathcal{ALC}$-concept description such that C is satisfiable w.r.t. $\mathcal{T}$, then C has a tree model w.r.t. $\mathcal{T}$.

Proof: blackboard

Proposition 3.8 (no tree model property)

$\mathcal{ALCO}$ does not have the tree model property.

Proof:

The concept $\{a\}$ does not have a tree model w.r.t. $\{\{a\} \sqsubseteq \exists r.\{a\}\}$.

Dresden

Disjoint union

Definition 3.9

Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be interpretations over disjoint domains.

Their disjoint union $\mathcal{I}_1 \uplus \mathcal{I}_2$ is defined as follows:

$$\begin{array}{rcl} \Delta^{\mathcal{I}_1 \uplus \mathcal{I}_2} & = & \Delta^{\mathcal{I}_1} \cup \Delta^{\mathcal{I}_2} \\ A^{\mathcal{I}_1 \uplus \mathcal{I}_2} & = & A^{\mathcal{I}_1} \cup A^{\mathcal{I}_2} \text{ for all } A \in N_C \\ r^{\mathcal{I}_1 \uplus \mathcal{I}_2} & = & r^{\mathcal{I}_1} \cup r^{\mathcal{I}_2} \text{ for all } r \in N_R \end{array}$$

Lemma 3.10

For all $\mathcal{ALC}$-concept descriptions C, and all $d \in \Delta^{\mathcal{I}_i}$ with $i \in \{1, 2\}$ we have

$$d \in C^{\mathcal{I}_i} \text{ iff } d \in C^{\mathcal{I}_1 \uplus \mathcal{I}_2}$$

Disjoint union

Definition 3.9

Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be interpretations over disjoint domains.

Their disjoint union $\mathcal{I}_1 \uplus \mathcal{I}_2$ is defined as follows:

$$
\begin{aligned}
\Delta^{\mathcal{I}_1 \uplus \mathcal{I}_2} &= \Delta^{\mathcal{I}_1} \cup \Delta^{\mathcal{I}_2} \\
A^{\mathcal{I}_1 \uplus \mathcal{I}_2} &= A^{\mathcal{I}_1} \cup A^{\mathcal{I}_2} \text{ for all } A \in N_C \\
r^{\mathcal{I} \uplus \mathcal{J}} &= r^{\mathcal{I}_1} \cup r^{\mathcal{I}_2} \text{ for all } r \in N_R
\end{aligned}
$$

Theorem 3.10b

Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be interpretations over disjoint domains, and $\mathcal{T}$ an $\mathcal{ALC}$-TBox.

If both $\mathcal{I}_1$ and $\mathcal{I}_2$ are a model of $\mathcal{T}$, then $\mathcal{I}_1 \uplus \mathcal{I}_2$ is also a model of $\mathcal{T}$.

Proof: blackboard

Finite model property

Definition 3.11 (finite model)

Let $\mathcal{T}$ be a TBox and C a concept description.

The interpretation $\mathcal{I}$ is a finite model of C w.r.t. $\mathcal{T}$ iff
$\mathcal{I}$ is a model of $\mathcal{T}$, $C^{\mathcal{I}} \neq \emptyset$, and $\Delta^{\mathcal{I}}$ is finite.

Theorem 3.12 (finite model property)

$\mathcal{ALC}$ has the finite model property,

i.e., if $\mathcal{T}$ is an $\mathcal{ALC}$-TBox and C an $\mathcal{ALC}$-concept description such that C is satisfiable w.r.t. $\mathcal{T}$, then C has a finite model w.r.t. $\mathcal{T}$.

Proof first requires some definitions.

Size

of $\mathcal{ALC}$-concept descriptions

- $C = A$: $|A| := 1$ for $A \in N_C$;
- $C = C_1 \sqcap C_2$ or $C = C_1 \sqcup C_2$: $|C| := 1 + |C_1| + |C_2|$;
- $C = \neg D$ or $C = \exists r.D$ or $C = \forall r.D$: $|C| := 1 + |D|$.

$$|A \sqcap \exists r.(A \sqcup B)| = 1 + 1 + (1 + (1 + 1 + 1)) = 6$$

Counts the occurrences of concept names, role names, and Boolean operators.

$$|\mathcal{T}| := \sum_{C \sqsubseteq D \in \mathcal{T}} |C| + |D|$$

Dresden

Subdescriptions

of $\mathcal{ALC}$-concept descriptions

- $C = A$: $\text{Sub}(A) := \{A\}$ for $A \in N_C$;
- $C = C_1 \sqcap C_2$ or $C = C_1 \sqcup C_2$: $\text{Sub}(C) := \{C\} \cup \text{Sub}(C_1) \cup \text{Sub}(C_2)$;
- $C = \neg D$ or $C = \exists r.D$ or $C = \forall r.D$: $\text{Sub}(C) := \{C\} \cup \text{Sub}(D)$.

$$\text{Sub}(A \sqcap \exists r.(A \sqcup B)) = \{A \sqcap \exists r.(A \sqcup B),\ A,\ \exists r.(A \sqcup B),\ A \sqcup B,\ B\}$$

$$\text{Sub}(\mathcal{T}) := \bigcup_{C \sqsubseteq D \in \mathcal{T}} \text{Sub}(C) \cup \text{Sub}(D)$$

- the cardinality of $\text{Sub}(C)$ is bounded by $|C|$;
- the cardinality of $\text{Sub}(\mathcal{T})$ is bounded by $|\mathcal{T}|$.

Dresden

Type

of an element of a model

Definition 3.13 (S-type)

Let S be a finite set of concept descriptions, and $\mathcal{I}$ an interpretation.

The S-type of $d \in \Delta^\mathcal{I}$ is defined as

$$t_S(d) := \{C \in S \mid d \in C^\mathcal{I}\}.$$

Lemma 3.14 (number of S-types)

$$|\{t_S(d) \mid d \in \Delta^\mathcal{I}\}| \leq 2^{|S|}$$

Proof: obvious

Filtration

of a model

Definition 3.15 (S-filtration)

Let S be a finite set of concept descriptions, and $\mathcal{I}$ an interpretation.

We define an equivalence relation $\simeq$ on $\Delta^{\mathcal{I}}$ as follows:

$$d \simeq e \text{ iff } t_S(d) = t_S(e)$$

The $\simeq$-equivalence class of $d \in \Delta^{\mathcal{I}}$ is denoted by $[d]$.

The S-filtration of $\mathcal{I}$ is the following interpretation $\mathcal{J}$:

- $\Delta^{\mathcal{J}} := \{[d] \mid d \in \Delta^{\mathcal{I}}\}$
- $A^{\mathcal{J}} := \{[d] \mid \exists d' \in [d].\ d' \in A^{\mathcal{I}}\}$ for all $A \in N_C$
- $r^{\mathcal{J}} := \{([d],[e]) \mid \exists d' \in [d], e' \in [e].\ (d', e') \in r^{\mathcal{I}}\}$ for all $r \in N_R$

Obviously, $|\Delta^{\mathcal{J}}| \leq 2^{|S|}$.

Dresden

Filtration

important property

We say that the finite set S of concept descriptions is closed iff

$$\bigcup\{\mathrm{Sub}(C) \mid C \in S\} \subseteq S$$

Lemma 3.16

Let S be a finite set of $\mathcal{ALC}$-concept descriptions, that is closed, $\mathcal{I}$ an interpretation, and $\mathcal{J}$ the S-filtration of $\mathcal{I}$. Then we have

$$d \in C^{\mathcal{I}} \quad \text{iff} \quad [d] \in C^{\mathcal{J}}$$

for all $d \in \Delta^{\mathcal{I}}$ and $C \in S$.

Proof: blackboard

The following proposition shows that $\mathcal{ALC}$ satisfies a property that is even stronger than the finite model property.

Proposition 3.17 (bounded model property)

Let $\mathcal{T}$ is an $\mathcal{ALC}$-TBox and C an $\mathcal{ALC}$-concept description, and $S := \mathrm{Sub}(C) \cup \mathrm{Sub}(\mathcal{T})$.

> If C is satisfiable w.r.t. $\mathcal{T}$, then there is a model $\widehat{\mathcal{I}}$ of $\mathcal{T}$ such that $C^{\widehat{\mathcal{I}}} \neq \emptyset$ and $|\Delta^{\widehat{\mathcal{I}}}| \leq 2^{|S|}$.

Proof: let $\mathcal{I}$ be a model of $\mathcal{T}$ with $C^{\mathcal{I}} \neq \emptyset$, and $\widehat{\mathcal{I}}$ be the S-filtration of $\mathcal{I}$

We must show:

- $|\Delta^{\widehat{\mathcal{I}}}| \leq 2^{|S|}$ — Lemma 3.14
- $C^{\widehat{\mathcal{I}}} \neq \emptyset$
- $\widehat{\mathcal{I}}$ is a model of $\mathcal{T}$

(the last two) follow from Lemma 3.16

The following proposition shows that $\mathcal{ALC}$ satisfies a property that is even stronger than the finite model property.

Proposition 3.17 (bounded model property)

Let $\mathcal{T}$ be a TBox, C a concept description, and $S := \mathrm{Sub}(C) \cup \mathrm{Sub}(\mathcal{T})$.

> If C is satisfiable w.r.t. $\mathcal{T}$, then there is a model $\widehat{\mathcal{I}}$ of $\mathcal{T}$ such that $C^{\widehat{\mathcal{I}}} \neq \emptyset$ and $|\Delta^{\widehat{\mathcal{I}}}| \leq 2^{|S|}$.

Corollary 3.17b (decidability)

In $\mathcal{ALC}$, satisfiability of a concept description w.r.t. a TBox is decidable.

No finite model property

Theorem 3.18 (no finite model property)

$\mathcal{ALCNI}$ does not have the finite model property.

Proof: blackboard

Chapter 4

Reasoning with tableaux algorithms

We start with an algorithm for deciding consistency of an ABox without a TBox since this covers most of the inference problems introduced in Chapter 2:

- acyclic TBoxes can be eliminated by expansion
- satisfiability, subsumption, and the instance problem can be reduced to ABox consistency

The tableau-based consistency algorithm tries to generate a finite model for the input ABox $\mathcal{A}_0$:

- applies tableau rules to extend the ABox *one rule per constructor*
- checks for obvious contradictions
- an ABox that is complete (no rule applies) and open (contains no obvious contradictions) describes a model

Tableau algorithm

example

$\mathcal{T}$ $\boxed{\text{GoodStudent} \equiv \text{Smart} \sqcap \text{Studious}}$

Subsumption question:

$$\exists\text{attended.Smart} \sqcap \exists\text{attended.Studious} \sqsubseteq^?_{\mathcal{T}} \exists\text{attended.GoodStudent}$$

Reduction to satisfiability: is the following concept unsatisfiable w.r.t. $\mathcal{T}$?

$$\exists\text{attended.Smart} \sqcap \exists\text{attended.Studious} \sqcap \neg\exists\text{attended.GoodStudent}$$

Reduction to consistency: is the following ABox inconsistent w.r.t. $\mathcal{T}$?

$$\{\ (\exists\text{attended.Smart} \sqcap \exists\text{attended.Studious} \sqcap \neg\exists\text{attended.GoodStudent})(a)\ \}$$

Expansion: is the following ABox inconsistent?

$$\{\ (\exists\text{attended.Smart} \sqcap \exists\text{attended.Studious} \sqcap \neg\exists\text{attended.}(\text{Smart} \sqcap \text{Studious}))(a)\ \}$$

Negation normal form: is the following ABox inconsistent?

$$\{\ (\exists\text{attended.Smart} \sqcap \exists\text{attended.Studious} \sqcap \forall\text{attended.}(\neg\text{Smart} \sqcup \neg\text{Studious}))(a)\ \}$$

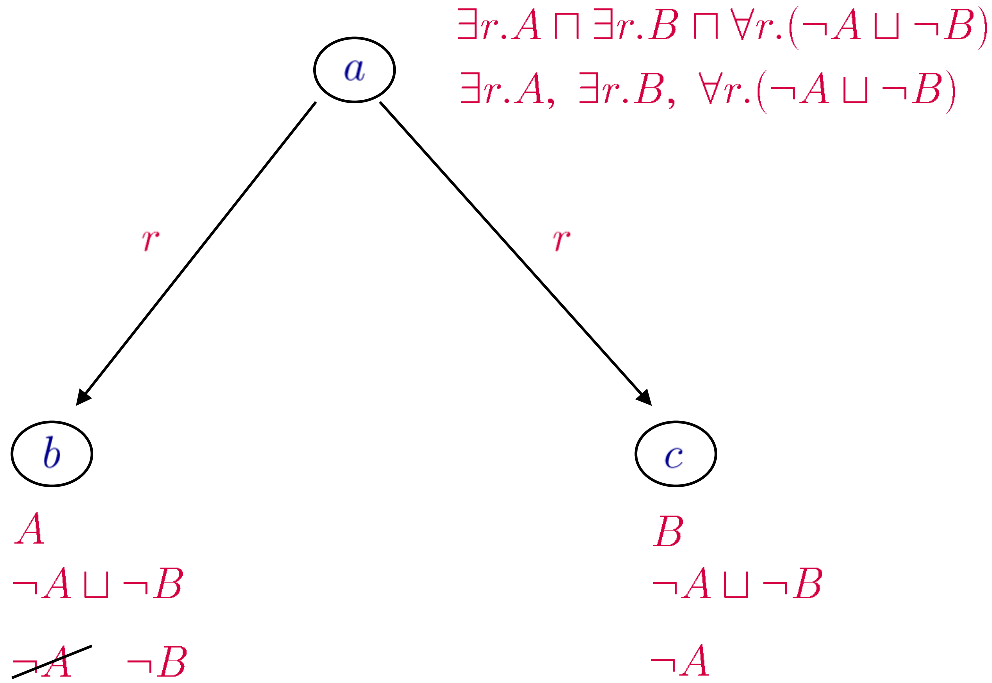
Dresden

Tableau algorithm

example continued

Is the following ABox inconsistent?

$\{\ (\exists \text{attended.Smart} \sqcap \exists \text{attended.Studious} \sqcap \forall \text{attended.}(\neg\text{Smart} \sqcup \neg\text{Studious}))(a)\ \}$



complete and open ABox

yields a model for the input ABox

and thus a counterexample

to the subsumption relationship

Dresden

Tableau algorithm

more formal description

Input: An $\mathcal{ALC}$-ABox $\mathcal{A}_0$

Output: "yes" if $\mathcal{A}_0$ is consistent

"no" otherwise

negation only in front of concept names

Preprocessing:

transform all concept descriptions in $\mathcal{A}_0$ into negation normal form (NNF) by applying the following rules:

$$
\begin{aligned}
\neg(C \sqcap D) &\rightsquigarrow \neg C \sqcup \neg D \\
\neg(C \sqcup D) &\rightsquigarrow \neg C \sqcap \neg D \\
\neg\neg C &\rightsquigarrow C \\
\neg(\exists r.C) &\rightsquigarrow \forall r.\neg C \\
\neg(\forall r.C) &\rightsquigarrow \exists r.\neg C
\end{aligned}
$$

The NNF can be computed in polynomial time, and it does not change the semantics of the concept.

Tableau algorithm

more formal description

Data structure:

finite set of ABoxes rather than a single ABox: start with $\{\mathcal{A}_0\}$ ← *in NNF*

Application of tableau rules:

the rules take one ABox from the set and replace it by finitely many new ABoxes

Termination:

if no more rules apply to any ABox in the set

complete ABox:
no rule applies to it

Answer:

"consistent" if the set contains an open ABox, i.e., an ABox not containing an obvious contradiction of the form

$$A(a) \text{ and } \neg A(a) \quad \text{for some individual name } a$$

"inconsistent" if all ABoxes in the set are closed (i.e., not open)

Dresden

Tableau rules

one for every constructor (except for negation)

The $\sqcap$-rule

Condition: $\mathcal{A}$ contains $(C \sqcap D)(a)$, but not both $C(a)$ and $D(a)$

Action: $\mathcal{A}' := \mathcal{A} \cup \{C(a), D(a)\}$

The $\sqcup$-rule

Condition: $\mathcal{A}$ contains $(C \sqcup D)(a)$, but neither $C(a)$ nor $D(a)$

Action: $\mathcal{A}' := \mathcal{A} \cup \{C(a)\}$ and $\mathcal{A}'' := \mathcal{A} \cup \{D(a)\}$

The $\exists$-rule

Condition: $\mathcal{A}$ contains $(\exists r.C)(a)$, but there is no c with $\{r(a,c), C(c)\} \subseteq \mathcal{A}$

Action: $\mathcal{A}' := \mathcal{A} \cup \{r(a,b), C(b)\}$ where b is a new individual name

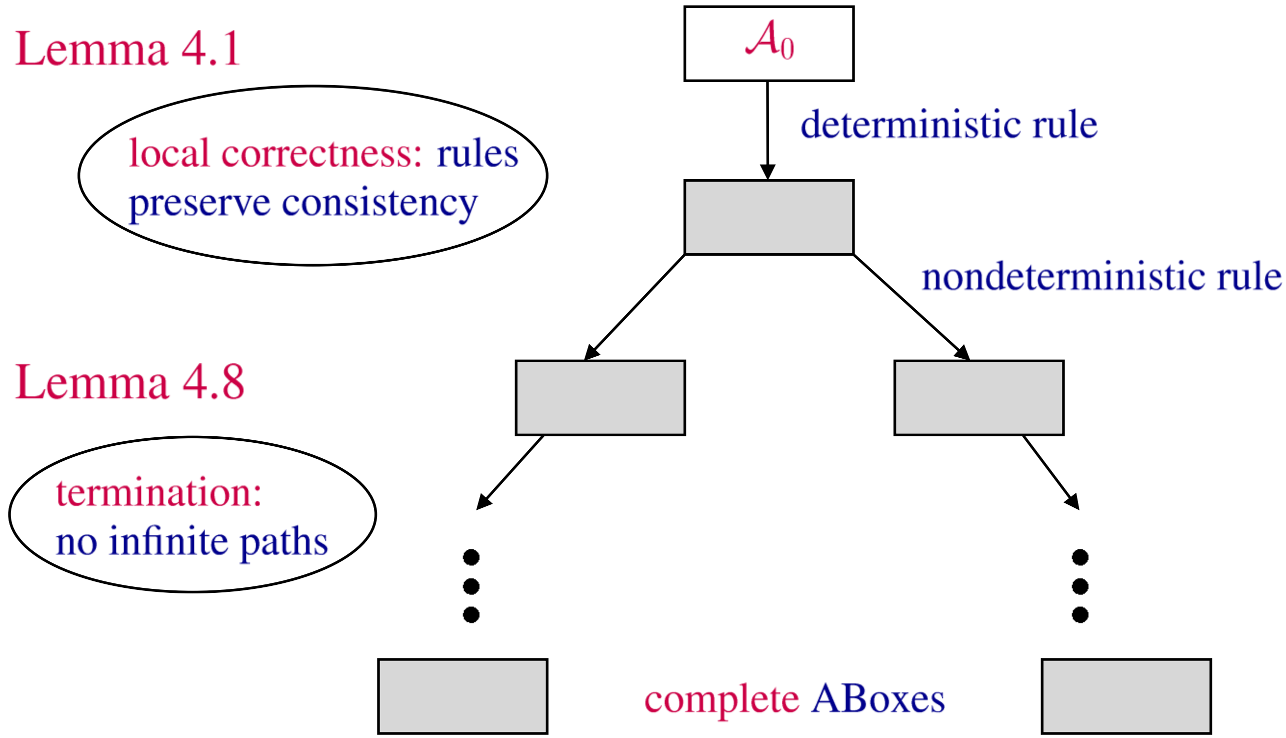
The $\forall$-rule

Condition: $\mathcal{A}$ contains $(\forall r.C)(a)$ and $r(a,b)$, but not $C(b)$

Action: $\mathcal{A}' := \mathcal{A} \cup \{C(b)\}$

Tableau algorithm

is a decision procedure for consistency



soundness: any complete and open ABox has a model

completeness: closed ABoxes do not have a model

Lemma 4.2

Dresden