Completeness and Decidability Results for CTL in Coq

Christian Doczkal   Gert Smolka

ITP 2014, Vienna

July 17, 2014

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Basic Result

- CTL formulas:

$$s, t := p \mid s \to t \mid \bot \mid \mathsf{AX}\, s \mid \mathsf{A}(s\, \mathsf{U}\, t) \mid \mathsf{A}(s\, \mathsf{R}\, t)$$

- Kripke Models:

$$\mathcal{M}, w \models s$$

- Hilbert Axiomatization:

$$\vdash s$$

Theorem (Certifying Decision Method)

$$\forall s. (\Sigma\, \mathcal{M}\, w.\ \mathcal{M}\ \mathit{finite} \wedge \mathcal{M}, w \models s) + (\vdash \neg s)$$

Constructed in Coq/Ssreflect without axioms

Basic Result

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- CTL formulas:

$$s, t := p \mid s \to t \mid \bot \mid \mathsf{AX}\, s \mid \mathsf{A}(s\,\mathsf{U}\,t) \mid \mathsf{A}(s\,\mathsf{R}\,t)$$

- Kripke Models:

$$\mathcal{M}, w \models s$$

- Hilbert Axiomatization:

$$\vdash s$$

Theorem (Certifying Decision Method)

$$\forall s. (\Sigma\, \mathcal{M}\, w.\ \mathcal{M}\ \textit{finite} \wedge \mathcal{M}, w \models s) + (\vdash \neg s)$$

Note: We do not claim this is executable!

Basic Result

- CTL formulas:

$$s, t := p \mid s \to t \mid \bot \mid \mathsf{AX}\, s \mid \mathsf{A}(s\, \mathsf{U}\, t) \mid \mathsf{A}(s\, \mathsf{R}\, t)$$

- Kripke Models:

$$\mathcal{M}, w \models s$$

- Hilbert Axiomatization:

$$\vdash s$$

Theorem (Certifying Decision Method)

$$\forall s. (\Sigma\, \mathcal{M}\, w.\ \mathcal{M}\ \textit{finite} \land \mathcal{M}, w \models s) + (\vdash \neg s)$$

- Satisfiability and provability of formulas is decidable
- Small model theorem
- $\forall s.\ \models s \implies\ \vdash s$

Certifying Decision Method

$\forall s.\ sat\ s + \neg sat\ s$

model construction → $\Sigma\, \mathcal{M}\, w.\ \mathcal{M}$ finite $\wedge\ \mathcal{M}, w \models s$

translation → $\vdash \neg s$

Completeness Proofs for decidable Modal Logics

SAARLAND UNIVERSITY
COMPUTER SCIENCE

1. Maximal consistent sets / canonical model
   - inherently classical
   - orthogonal to decidability
   - difficult for non-compact logics (like CTL)

2. Based on model search procedures
   - algorithmic definition of satisfiability
   - definition of unsatisfiability implicit

3. Based on analytic Tableaux
   - Inductive definition of unsatisfiability
   - Closure conditions for satisfiability

Completeness Proofs for decidable Modal Logics

1. Maximal consistent sets / canonical model
   - inherently classical
   - orthogonal to decidability
   - difficult for non-compact logics (like CTL)

2. Based on model search procedures
   - algorithmic definition of satisfiability
   - definition of unsatisfiability implicit

3. Based on analytic Tableaux $\Leftarrow$ This is what we want!
   - Inductive definition of unsatisfiability
   - Closure conditions for satisfiability

Related Work on CTL

- Pruning-based EXPTIME decision-procedure, small model theorem, completeness of Hilbert axiomatization [Emerson Halpern 85]
- Simpler declarative model construction [Emerson 90]
- Analytic cut-free sequent calculus for CTL [Brünnler Lange 08]

Related Work on CTL

- Pruning-based EXPTIME decision-procedure, small model theorem, completeness of Hilbert axiomatization [Emerson Halpern 85]
- Simpler declarative model construction [Emerson 90]
- Analytic cut-free sequent calculus for CTL [Brünnler Lange 08]
- Model checking algorithms / decision procedures . . .

Related Work on CTL

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- Pruning-based EXPTIME decision-procedure, small model theorem, completeness of Hilbert axiomatization [Emerson Halpern 85]
- Simpler declarative model construction [Emerson 90]
- Analytic cut-free sequent calculus for CTL [Brünnler Lange 08]
- Model checking algorithms / decision procedures ...

- No formalized completeness results
- Certifying Decision Method for K^+ [Doczkal Smolka 12]

Certifying Decision Method

$\forall s.\ \neg tab\ \{s\} + tab\ \{s\}$

model construction

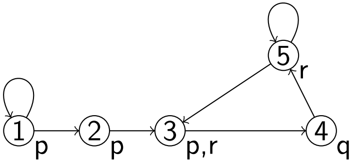
translation

$\Sigma\, \mathcal{M}\, w.\ \mathcal{M}$ finite $\wedge\, \mathcal{M}, w \models s$

$\vdash \neg s$

CTL [Emerson Clarke 81,Emerson Halpern 85]

- Interpreted over serial transition systems



- CTL formulas:

$$s, t := p \mid s \to t \mid \bot \mid \mathsf{AX}\, s \mid \mathsf{A}(s\, \mathsf{U}\, t) \mid \mathsf{A}(s\, \mathsf{R}\, t)$$

Fixpoint Formulas

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- "Path" modalities are fixpoint formulas:

$$\frac{\mathcal{M}, w \models t}{\mathcal{M}, w \models \mathsf{A}(s\,\mathsf{U}\,t)} \qquad \frac{\mathcal{M}, w \models s \qquad \forall v.\, w \to_{\mathcal{M}} v \Longrightarrow \mathsf{A}(s\,\mathsf{U}\,t)}{\mathcal{M}, w \models \mathsf{A}(s\,\mathsf{U}\,t)}$$

$$\frac{\mathcal{M}, w \models s \qquad \mathcal{M}, w \models t}{\mathcal{M}, w \models \mathsf{A}(s\,\mathsf{R}\,t)} \qquad \frac{\mathcal{M}, w \models t \qquad \forall v.\, w \to_{\mathcal{M}} v \Longrightarrow \mathsf{A}(s\,\mathsf{R}\,t)}{\mathcal{M}, w \models \mathsf{A}(s\,\mathsf{R}\,t)}$$

Fixpoint Formulas

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- "Path" modalities are fixpoint formulas:

$$\frac{\mathcal{M}, w \models t}{\mathcal{M}, w \models \mathsf{A}(s \,\mathsf{U}\, t)} \qquad \frac{\mathcal{M}, w \models s \qquad \forall v.\, w \to_{\mathcal{M}} v \Longrightarrow \mathsf{A}(s \,\mathsf{U}\, t)}{\mathcal{M}, w \models \mathsf{A}(s \,\mathsf{U}\, t)}$$

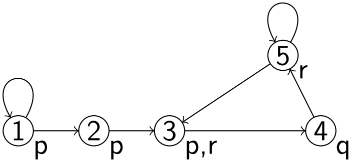
$$\frac{\mathcal{M}, w \models s \qquad \mathcal{M}, w \models t}{\mathcal{M}, w \models \mathsf{A}(s \,\mathsf{R}\, t)} \qquad \frac{\mathcal{M}, w \models t \qquad \forall v.\, w \to_{\mathcal{M}} v \Longrightarrow \mathsf{A}(s \,\mathsf{R}\, t)}{\mathcal{M}, w \models \mathsf{A}(s \,\mathsf{R}\, t)}$$

- Classically equivalent to infinite path semantics
- Characteristic Equivalences:

$$\begin{aligned} \mathsf{A}(s \,\mathsf{U}\, t) &\equiv t \vee s \wedge \mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}\, t) \\ \mathsf{A}(s \,\mathsf{R}\, t) &\equiv s \wedge t \vee t \wedge \mathsf{AX}\,\mathsf{A}(s \,\mathsf{R}\, t) \end{aligned}$$

CTL [Emerson Clarke 81,Emerson Halpern 85]

- Interpreted over serial transition systems

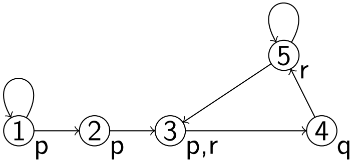


- CTL formulas:

$$s, t := p \mid s \rightarrow t \mid \bot \mid \mathsf{AX}\, s \mid \mathsf{A}(s\, \mathsf{U}\, t) \mid \mathsf{A}(s\, \mathsf{R}\, t)$$

CTL [Emerson Clarke 81,Emerson Halpern 85]

- Interpreted over serial transition systems



- CTL formulas:

$$
\begin{aligned}
s, t := p \mid s \to t \mid \bot \; &\mid \mathsf{AX}\, s \mid \mathsf{A}(s \,\mathsf{U}\, t) \mid \mathsf{A}(s \,\mathsf{R}\, t) \\
&\mid \mathsf{EX}\, s \mid \mathsf{E}(s \,\mathsf{R}\, t) \mid \mathsf{E}(s \,\mathsf{U}\, t)
\end{aligned}
$$

Analytic Tableaux for CTL

- Obtained dualizing sequent calculus CT [Brünnler Lange 08]
- Derives unsatisfiable clauses (finite sets of formulas)

$$\frac{C_1 \dots C_n}{C}$$

- Backward derivations correspond to construction of a model
- Literal clauses correspond to potential states in a model

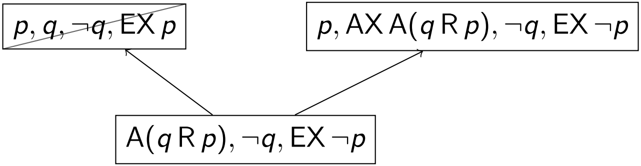
$$\{p, \neg q, \mathsf{AX}\, s, \mathsf{EX}\, t\}$$

Example I : Successful Derivation

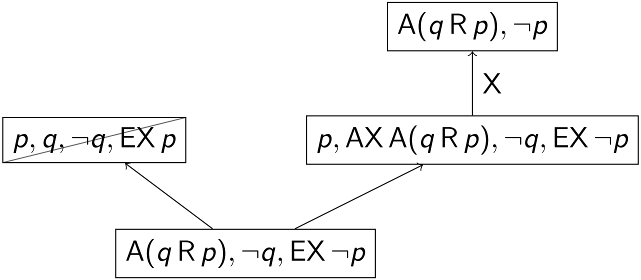
SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$\boxed{\mathsf{A}(q \,\mathsf{R}\, p), \neg q, \mathsf{EX}\,\neg p}$$

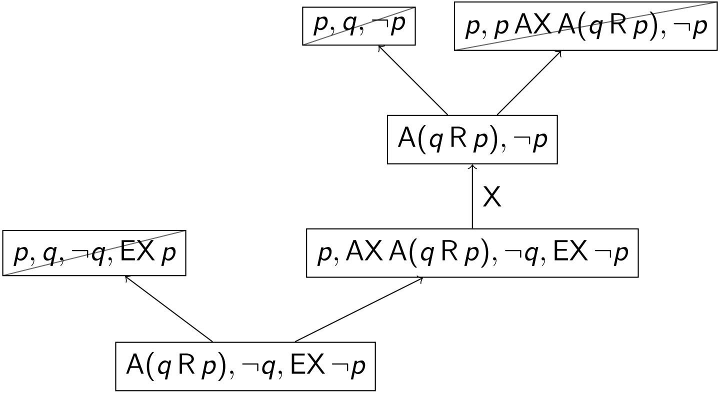
Example I : Successful Derivation



Example I : Successful Derivation



Example I : Successful Derivation



Example II : Failed Derivation

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$p, \mathsf{AX}\,\mathsf{A}(q\,\mathsf{R}\,p), \neg q, \mathsf{EX}\,\mathsf{EX}\,\neg p$$

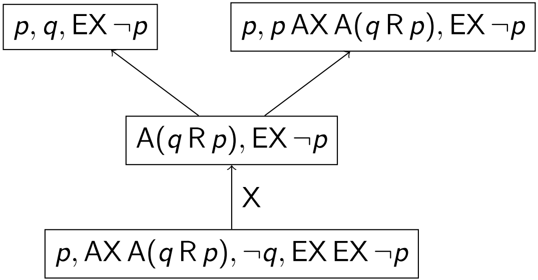
Example II : Failed Derivation

$$\mathsf{A}(q\,\mathsf{R}\,p),\ \mathsf{EX}\,\neg p$$

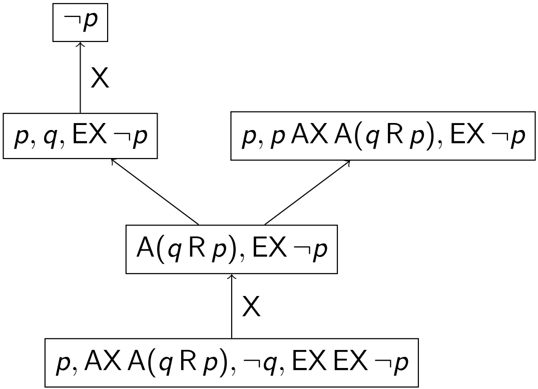
$$\uparrow\ \mathsf{X}$$

$$p,\ \mathsf{AX}\,\mathsf{A}(q\,\mathsf{R}\,p),\ \neg q,\ \mathsf{EX}\,\mathsf{EX}\,\neg p$$

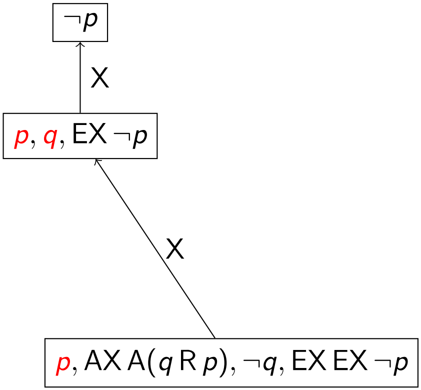
Example II : Failed Derivation



Example II : Failed Derivation



Example II : Failed Derivation



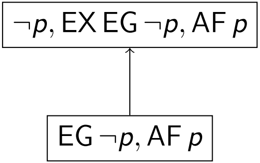
Example III : Cycle

$$\mathsf{AF}\, s \equiv \mathsf{A}(\top\, \mathsf{U}\, s) \qquad\qquad \mathsf{EG}\, s \equiv \mathsf{E}(\bot\, \mathsf{R}\, s)$$

$$\boxed{\mathsf{EG}\, \neg p, \mathsf{AF}\, p}$$

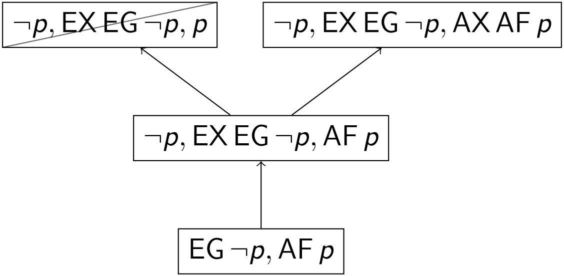
Example III : Cycle

$$\mathsf{AF}\,s \equiv \mathsf{A}(\top\,\mathsf{U}\,s) \qquad\qquad \mathsf{EG}\,s \equiv \mathsf{E}(\bot\,\mathsf{R}\,s)$$



Example III : Cycle

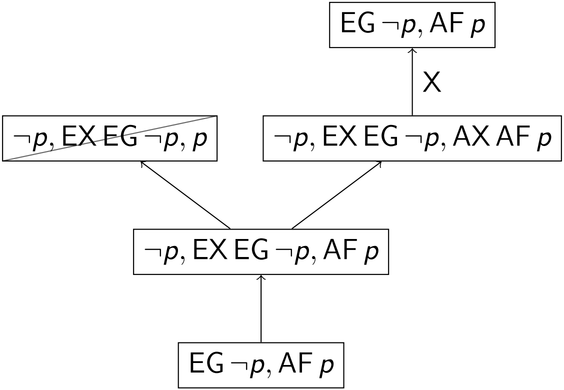
$$\mathsf{AF}\, s \equiv \mathsf{A}(\top\, \mathsf{U}\, s) \qquad\qquad \mathsf{EG}\, s \equiv \mathsf{E}(\bot\, \mathsf{R}\, s)$$



Example III : Cycle

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$\mathsf{AF}\, s \equiv \mathsf{A}(\top\, \mathsf{U}\, s)$ $\qquad$ $\mathsf{EG}\, s \equiv \mathsf{E}(\bot\, \mathsf{R}\, s)$



Histories [Brünnler Lange 08]

$$\frac{C, t \qquad C, s, \mathrm{AX}\ \mathrm{A}(s\,\mathrm{U}\,t)}{C,\ \mathrm{A}\,(s\,\mathrm{U}\,t)}$$

$$\frac{C|\ \mathrm{A}(s\,\mathrm{U}_{\emptyset}\,t)}{C,\ \mathrm{A}(s\,\mathrm{U}\,t)} \qquad \frac{C, t \qquad C, s|\ \mathrm{AX}\ \mathrm{A}(s\,\mathrm{U}_{H,C}\,t)}{C|\ \mathrm{A}(s\,\mathrm{U}_{H}\,t)} \qquad \frac{}{C|\ \mathrm{A}(s\,\mathrm{U}_{H,C}\,t)}$$

- Focusing rule only applies to history-free clauses
- At most one annotated eventuality

Histories [Brünnler Lange 08]

$$\frac{C, t \qquad C, s, \mathsf{EX}\ \mathsf{E}(s\,\mathsf{U}\,t)}{C,\ \mathsf{E}(s\,\mathsf{U}\,t)}$$

$$\frac{C|\ \mathsf{E}(s\,\mathsf{U}_{\emptyset}\,t)}{C,\ \mathsf{E}(s\,\mathsf{U}\,t)} \qquad \frac{C, t \qquad C, s|\ \mathsf{EX}\ \mathsf{E}(s\,\mathsf{U}_{H,C}\,t)}{C|\ \mathsf{E}(s\,\mathsf{U}_{H}\,t)} \qquad \frac{}{C|\ \mathsf{E}(s\,\mathsf{U}_{H,C}\,t)}$$

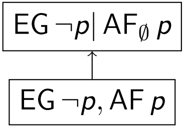
- Focusing rule only applies to history-free clauses
- At most one annotated eventuality

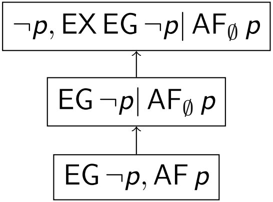
Example IV : Histories

SAARLAND UNIVERSITY
COMPUTER SCIENCE

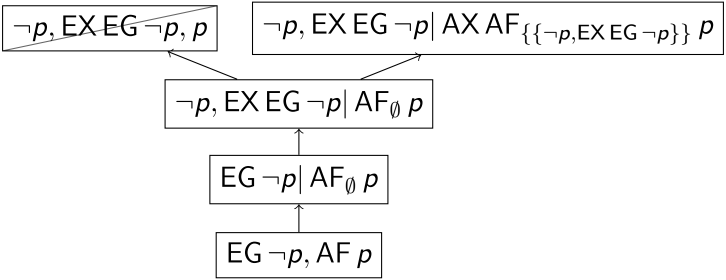
$\mathsf{EG}\,\neg p, \mathsf{AF}\,p$

Example IV : Histories

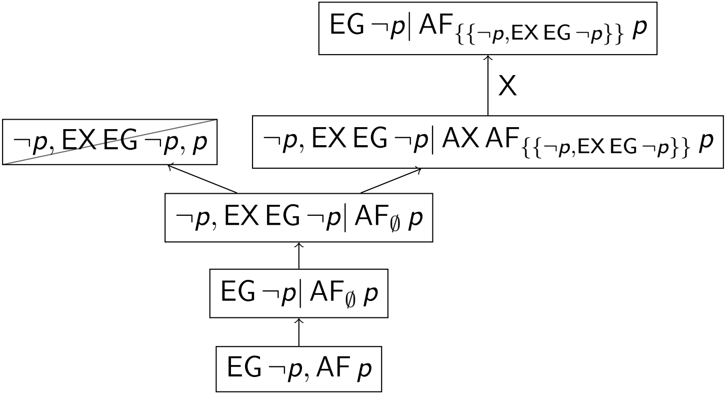



$\neg p, \mathsf{EX}\,\mathsf{EG}\,\neg p \,|\, \mathsf{AF}_{\emptyset}\, p$
$\mathsf{EG}\,\neg p \,|\, \mathsf{AF}_{\emptyset}\, p$
$\mathsf{EG}\,\neg p, \mathsf{AF}\, p$

Example IV : Histories

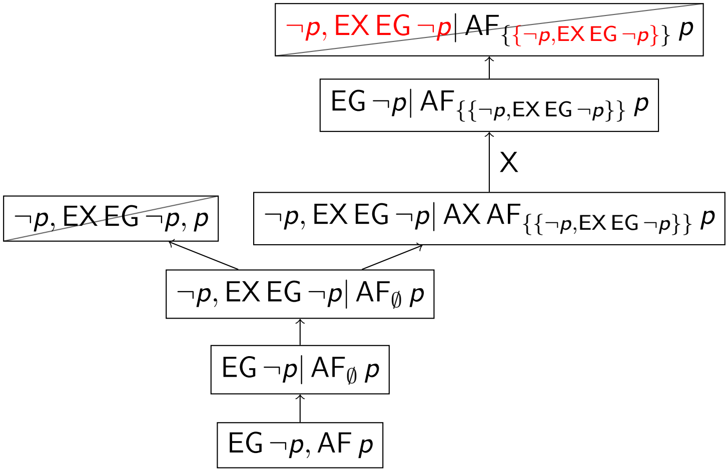


Example IV : Histories



Example IV : Histories

SAARLAND UNIVERSITY
COMPUTER SCIENCE



Certifying Decision Method

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$\forall s.\ \neg tab\ \{s\} + tab\ \{s\}$

model construction

translation

$\Sigma\, \mathcal{M}\, w.\ \mathcal{M}$ finite $\wedge\ \mathcal{M}, w \models s$

$\vdash \neg s$

Decidability

- Exist finite clause universes closed under backward rule application
  - at most one annotated eventuality
  - histories contain only history-free clauses
- Rule instantiation decidable

Decidability

- Exist finite clause universes closed under backward rule application
  - at most one annotated eventuality
  - histories contain only history-free clauses
- Rule instantiation decidable
- Express one-step derivability (within a universe) as monotone function

$$\mathit{step} : \mathit{set}\ \mathit{clause} \to \mathit{set}\ \mathit{clause}$$

Theorem (Decidability)

$\forall \mathcal{U}. \forall C \in \mathcal{U}. C\ \mathit{derivable} \iff C \in \mathit{iter}\ |\mathcal{U}|\ \emptyset\ \mathit{step}$

Decidability

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- Exist finite clause universes closed under backward rule application
  - at most one annotated eventuality
  - histories contain only history-free clauses
- Rule instantiation decidable
- Express one-step derivability (within a universe) as monotone function

$$\mathit{step} : \mathit{set\ clause} \to \mathit{set\ clause}$$

Theorem (Decidability)

$\forall \mathcal{U}. \forall C \in \mathcal{U}. C\ \mathit{derivable} \iff C \in \mathit{iter}\ |\mathcal{U}|\ \emptyset\ \mathit{step}$

- Every clause is element of some clause universe

Finite Sets in Coq

- Formulas, sets of formulas, sets of sets of formulas, ...
- Finite sets not native to type theory
- Want a set library with:
  - Decidable membership
  - Extensional representation
  - All the usual operations, i.e., $\{\, x \in X \mid P \,\}$, $\{\, f\,x \mid x \in X \,\}$
  - Powerset (if $A : \mathit{set}\ T$ then $\mathcal{P}(A) : \mathit{set}\ (\mathit{set}\ T)$)
- Ssreflect: base type must be finite

Finite Sets in Coq

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- Formulas, sets of formulas, sets of sets of formulas, ...
- Finite sets not native to type theory
- Want a set library with:
  - Decidable membership
  - Extensional representation
  - All the usual operations, i.e., $\{ x \in X \mid P \}$, $\{ f\,x \mid x \in X \}$
  - Powerset (if $A : \mathit{set}\ T$ then $\mathcal{P}(A) : \mathit{set}\ (\mathit{set}\ T)$)
- Ssreflect: base type must be finite

- New Set Library over countable base types:
  - Implemented as constructive quotient on duplicate free lists
  - Lift list operations to set operations
  - $\approx$ 150 Lemmas incl. indexed unions, fixpoint theorem, automation

Certifying Decision Method

$\forall s.\ \neg tab\ \{s\} + tab\ \{s\}$ ✓

model construction

translation

$\Sigma\,\mathcal{M}\,w.\ \mathcal{M}$ finite $\wedge\ \mathcal{M}, w \models s$

$\vdash \neg s$

Soundness

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$C = \{s_1, \dots, s_n\} \equiv s_1 \land \dots \land s_n$$
$$H = \{C_1, \dots, C_n\} \equiv \neg C_1 \land \dots \land \neg C_n$$
$$s \,\mathsf{U}_H\, t \equiv (s \land H)\,\mathsf{U}(t \land H)$$

$$\frac{C \mid \mathsf{E}(s\,\mathsf{U}_\emptyset\, t)}{C, \mathsf{E}(s\,\mathsf{U}\, t)} \qquad \frac{C, t \qquad C, s \mid \mathsf{EX}\,\mathsf{E}(s\,\mathsf{U}_{H,C}\, t)}{C \mid \mathsf{E}(s\,\mathsf{U}_H\, t)} \qquad \frac{}{C \mid \mathsf{E}(s\,\mathsf{U}_{H,C}\, t)}$$

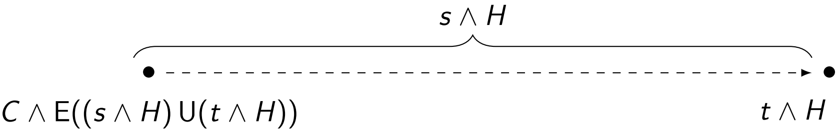
Soundness

$$C = \{s_1, \ldots, s_n\} \equiv s_1 \wedge \ldots \wedge s_n$$

$$H = \{C_1, \ldots, C_n\} \equiv \neg C_1 \wedge \ldots \wedge \neg C_n$$

$$s \,\mathsf{U}_H\, t \equiv (s \wedge H)\,\mathsf{U}(t \wedge H)$$

$$\frac{C \mid \mathsf{E}(s \,\mathsf{U}_\emptyset\, t)}{C, \mathsf{E}(s \,\mathsf{U}\, t)} \qquad \frac{C, t \qquad C, s \mid \mathsf{EX}\,\mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}{C \mid \mathsf{E}(s \,\mathsf{U}_H\, t)} \qquad \frac{}{C \mid \mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}$$



Soundness

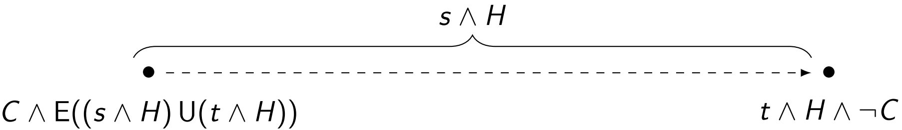
SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$C = \{s_1, \dots, s_n\} \equiv s_1 \land \dots \land s_n$$

$$H = \{C_1, \dots, C_n\} \equiv \neg C_1 \land \dots \land \neg C_n$$

$$s \,\mathsf{U}_H\, t \equiv (s \land H)\,\mathsf{U}(t \land H)$$

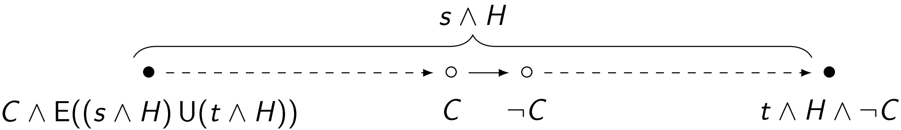
$$\frac{C \mid \mathsf{E}(s \,\mathsf{U}_\emptyset\, t)}{C, \mathsf{E}(s \,\mathsf{U}\, t)} \qquad \frac{C, t \qquad C, s \mid \mathsf{EX}\,\mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}{C \mid \mathsf{E}(s \,\mathsf{U}_H\, t)} \qquad \frac{}{C \mid \mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}$$



Soundness

$$C = \{s_1, \dots, s_n\} \equiv s_1 \wedge \dots \wedge s_n$$
$$H = \{C_1, \dots, C_n\} \equiv \neg C_1 \wedge \dots \wedge \neg C_n$$
$$s \,\mathsf{U}_H\, t \equiv (s \wedge H)\,\mathsf{U}(t \wedge H)$$

$$\frac{C \mid \mathsf{E}(s \,\mathsf{U}_\emptyset\, t)}{C, \mathsf{E}(s \,\mathsf{U}\, t)} \qquad \frac{C, t \qquad C, s \mid \mathsf{EX}\,\mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}{C \mid \mathsf{E}(s \,\mathsf{U}_H\, t)} \qquad \frac{}{C \mid \mathsf{E}(s \,\mathsf{U}_{H,C}\, t)}$$

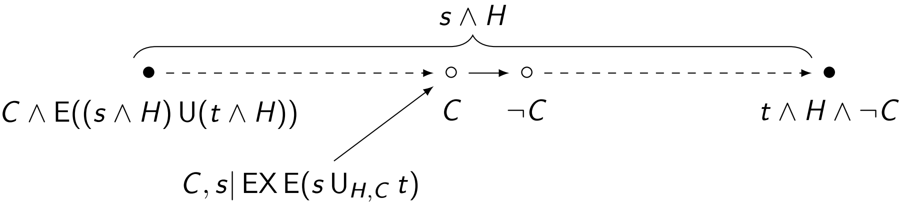


Soundness

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$C = \{s_1, \dots, s_n\} \equiv s_1 \wedge \dots \wedge s_n$$
$$H = \{C_1, \dots, C_n\} \equiv \neg C_1 \wedge \dots \wedge \neg C_n$$
$$s \,\mathsf{U}_H\, t \equiv (s \wedge H)\,\mathsf{U}(t \wedge H)$$

$$\frac{C|\,\mathsf{E}(s\,\mathsf{U}_\emptyset\, t)}{C, \mathsf{E}(s\,\mathsf{U}\, t)} \qquad \frac{C, t \qquad C, s|\,\mathsf{EX}\,\mathsf{E}(s\,\mathsf{U}_{H,C}\, t)}{C|\,\mathsf{E}(s\,\mathsf{U}_H\, t)} \qquad \frac{}{C|\,\mathsf{E}(s\,\mathsf{U}_{H,C}\, t)}$$



Translation

- One lemma per rule

$$\frac{C_1 \dots C_n}{C} \quad \Longrightarrow \quad \vdash (\neg C_1 \wedge \dots \wedge \neg C_n \to \neg C)$$

- Proofs mirror semantic soundness proof *inside* the Hilbert system
- Only possible since CTL can express semantics of histories
- Difficult Rules:

$$\frac{C, t \qquad C, s \mid \mathsf{AX}\,\mathsf{A}(s\,\mathsf{U}_{H,C}\,t)}{C \mid \mathsf{A}(s\,\mathsf{U}_H\,t)} \qquad \frac{C, t \qquad C, s \mid \mathsf{EX}\,\mathsf{E}(s\,\mathsf{U}_{H,C}\,t)}{C \mid \mathsf{E}(s\,\mathsf{U}_H\,t)}$$

- Requires non-trivial instantiations of induction axioms

Hilbert System

$$\begin{array}{rl}
\mathsf{K} & s \to t \to s \\
\mathsf{S} & ((u \to s \to t) \to (u \to s) \to u \to t) \\
\mathsf{DN} & ((s \to \bot) \to \bot) \to s \\
\mathsf{N} & \mathsf{AX}(s \to t) \to \mathsf{AX}\, s \to \mathsf{AX}\, t \\
\mathsf{U1} & t \to \mathsf{A}(s\, \mathsf{U}\, t) \\
\mathsf{U2} & s \to \mathsf{AX}\, \mathsf{A}(s\, \mathsf{U}\, t) \to \mathsf{A}(s\, \mathsf{U}\, t) \\
\mathsf{R1} & \mathsf{A}(s\, \mathsf{R}\, t) \to t \\
\mathsf{R2} & \mathsf{A}(s\, \mathsf{R}\, t) \to (s \to \bot) \to \mathsf{AX}\, \mathsf{A}(s\, \mathsf{R}\, t) \\
\mathsf{AX} & \mathsf{AX}\, \bot \to \bot
\end{array}$$

$$\frac{s \qquad s \to t}{t}\ \mathsf{MP} \qquad \frac{s}{\mathsf{AX}\, s}\ \mathsf{Nec} \qquad \frac{t \to u \qquad s \to \mathsf{AX}\, u \to u}{\mathsf{A}(s\, \mathsf{U}\, t) \to u}\ \mathsf{AU_{ind}}$$

$$\frac{u \to t \qquad u \to (s \to \bot) \to \mathsf{AX}\, u}{u \to \mathsf{A}(s\, \mathsf{R}\, t)}\ \mathsf{AR_{ind}}$$

Hilbert System

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$
\begin{array}{rl}
\mathsf{K} & s \to t \to s \\
\mathsf{S} & ((u \to s \to t) \to (u \to s) \to u \to t) \\
\mathsf{DN} & ((s \to \bot) \to \bot) \to s \\
\mathsf{N} & \mathsf{AX}(s \to t) \to \mathsf{AX}\, s \to \mathsf{AX}\, t \\
\mathsf{U1} & t \to \mathsf{A}(s\,\mathsf{U}\,t) \\
\mathsf{U2} & s \to \mathsf{AX}\,\mathsf{A}(s\,\mathsf{U}\,t) \to \mathsf{A}(s\,\mathsf{U}\,t) \\
\mathsf{R1} & \mathsf{A}(s\,\mathsf{R}\,t) \to t \\
\mathsf{R2} & \mathsf{A}(s\,\mathsf{R}\,t) \to (s \to \bot) \to \mathsf{AX}\,\mathsf{A}(s\,\mathsf{R}\,t) \\
\mathsf{AX} & \mathsf{AX}\,\bot \to \bot
\end{array}
$$

$$
\frac{s \qquad s \to t}{t}\ \mathsf{MP} \qquad \frac{s}{\mathsf{AX}\, s}\ \mathsf{Nec} \qquad \frac{t \to u \qquad s \to \mathsf{AX}\, u \to u}{\mathsf{A}(s\,\mathsf{U}\,t) \to u}\ \mathsf{AU}_{\mathsf{ind}}
$$

$$
\frac{u \to t \qquad u \to (s \to \bot) \to \mathsf{AX}\, u}{u \to \mathsf{A}(s\,\mathsf{R}\,t)}\ \mathsf{AR}_{\mathsf{ind}}
$$

Hilbert Proofs in Coq

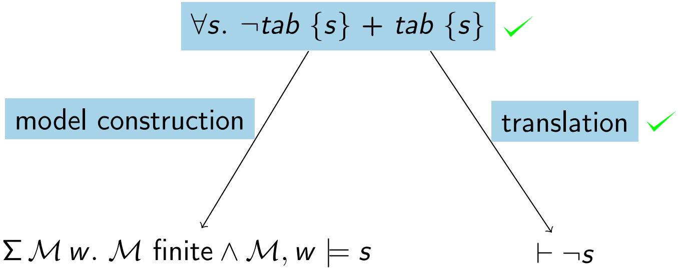
- Finding proofs in bare Hilbert system is tedious
- Need infrastructure in Coq
  - ND-style reasoning
    1. Define big conjunction
    2. Simulate contexts with formulas of the form $C \to s$
    3. Write tactics for "ND-style" reasoning
  - Use setoid rewriting with preorder

$$s \preccurlyeq t \quad \equiv \quad \vdash s \to t$$

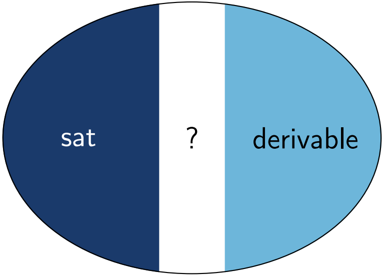
  - Build modular set of lemmas: $M \subseteq P \subseteq K \subseteq CTL$

Certifying Decision Method

SAARLAND UNIVERSITY
COMPUTER SCIENCE



Histories and Completeness



$$\emptyset |\, \mathsf{A}(p\, \mathsf{U}_{\{\{p\}\}}\, p) \equiv \mathsf{A}(p \wedge \neg p\, \mathsf{U}\, p \wedge \neg p) \equiv \bot$$

$$\frac{p \qquad p |\, \mathsf{AX}\, \mathsf{A}(p\, \mathsf{U}_{\{\{p\},\emptyset\}}\, p)}{\emptyset |\, \mathsf{A}(p\, \mathsf{U}_{\{\{p\}\}}\, p)}$$

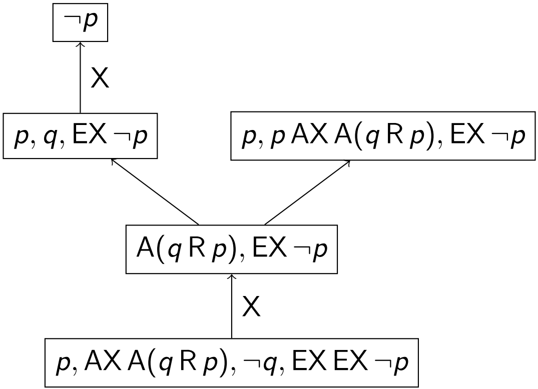
Model Construction

- Build models syntactic universes, not single formulas
  - Fix some syntactic universe $\mathcal{U}$
  - $\mathcal{D} = \{\, C \in \mathcal{U} \mid C\, underivable, history\text{–}free, literal \,\}$
  - Build model $\mathcal{M}$ where every clause from $\mathcal{D}$ labels some state and

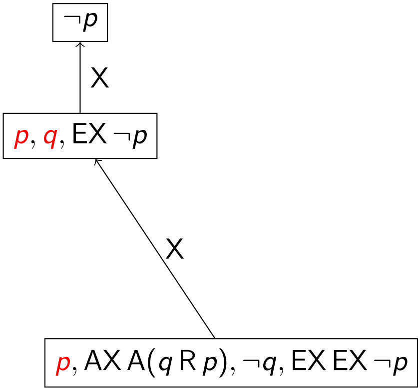
$$\forall s. \forall w \in \mathcal{M}. s \in C_w \implies \mathcal{M}, w \models s$$

- Construction differs significantly from [Brünnler Lange 08]

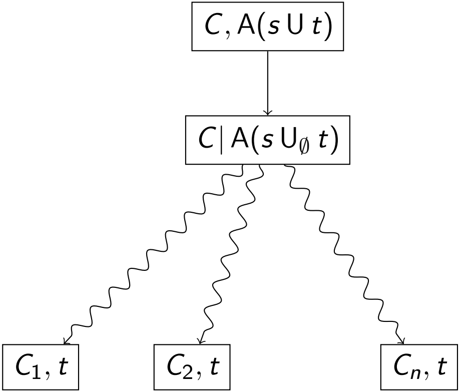
Example II : Failed Derivation



Example II : Failed Derivation



Fragments

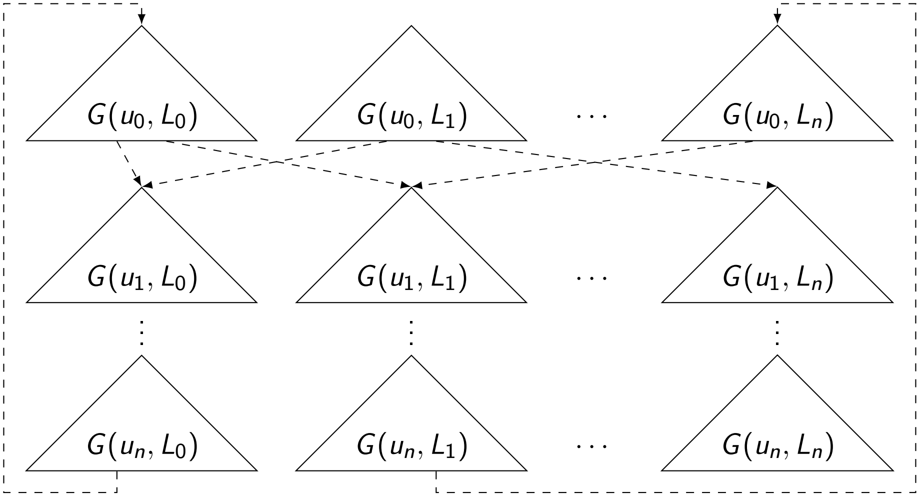


$$\frac{C \mid \mathsf{A}(s\,\mathsf{U}_\emptyset\,t)}{C, \mathsf{A}(s\,\mathsf{U}\,t)}$$

$$\frac{C, t \qquad C, s \mid \mathsf{AX}\,\mathsf{A}(s\,\mathsf{U}_{H,C}\,t)}{C \mid \mathsf{A}(s\,\mathsf{U}_H\,t)}$$

$$\frac{}{C \mid \mathsf{A}(s\,\mathsf{U}_{H,C}\,t)}$$

Matrix of Fragments [Emerson 90]



$u_i \approx$ *eventualities in* $\mathcal{U}$ $\qquad L_i \in \mathcal{D}$

Trees vs. Graphs

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- Trees:
  - Inductive/compositional structure
  - acyclic and rooted by construction
  - set of nodes/edge relation implicit
- Trees ideal for fragment construction
- Graphs:
  - monolithic structure
  - set of nodes/edge relation explicit
  - Rootedness and acyclicity as separate conditions
- Graphs required for matrix assembly
- Need to transfer fragment properties from trees to graphs

Certifying Decision Method

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$\forall s.\ \neg tab\ \{s\} + tab\ \{s\}$ ✓

model construction ✓

translation ✓

$\Sigma\, \mathcal{M}\, w.\ \mathcal{M}$ finite $\wedge\ \mathcal{M}, w \models s$

$\vdash \neg s$

Summary

- Formalized Results on CTL
  - Decidability (satisfiability, validity, provability)
  - Completeness of history-based tableau calculus (orig.)
  - Small model theorem
  - Compositional translation from tableaux to Hilbert (orig.)
  - Completeness of Hilbert axiomatization
- Statistics (coqwc)

	Spec	**Proof**
Finite set library	377	332
CTL def + Hilbert soundness	221	158
Tableau decidability (incl. def)	263	141
Translation (incl. infrastructure)	396	376
Tableau completeness	407	671
⋮	⋮	⋮
Total	1789	1790

Thank You!

Questions?

http://www.ps.uni-saarland.de/extras/itp14

Support [Kaminski Smolka 10]

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Theorem (Model Correctness)

$$\forall s. \forall w \in \mathcal{M}. s \in C_w \implies \mathcal{M}, w \models s$$

Support [Kaminski Smolka 10]

SAARLAND UNIVERSITY
COMPUTER SCIENCE

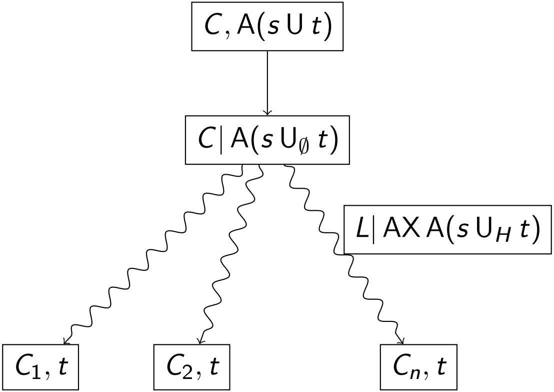
Theorem (Model Correctness)

$$\forall s. \forall w \in \mathcal{M}. C_w \rhd s \Longrightarrow \mathcal{M}, w \models s$$

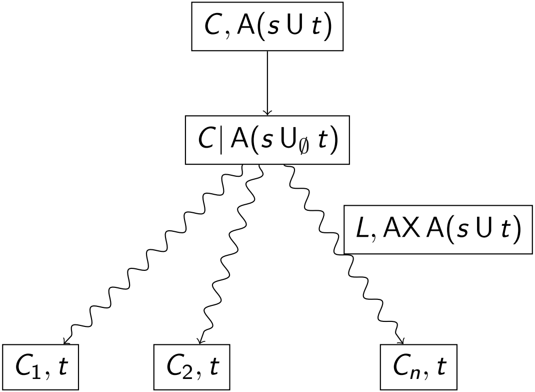
$$\begin{aligned} C \rhd s &\equiv s \in C \text{ and } s \textit{ literal} \\ C \rhd (s \to t) &\equiv C \not\rhd s \text{ or } C \rhd t \\ C \rhd (\mathsf{A}(s \,\mathsf{U}\, t)) &\equiv C \rhd t \text{ or } (C \rhd s \text{ and } C \rhd \mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}\, t)) \\ &\vdots \end{aligned}$$

"upwards closure" of literal clause to form infinite Hintikka set

Fragments


$C, \mathsf{A}(s \,\mathsf{U}\, t)$
$C| \,\mathsf{A}(s \,\mathsf{U}_\emptyset\, t)$
$L| \,\mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}_H\, t)$
C_1, t
C_2, t
C_n, t

Fragments



$C, \mathsf{A}(s \,\mathsf{U}\, t)$
$C \mid \mathsf{A}(s \,\mathsf{U}_\emptyset\, t)$
$L, \mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}\, t)$
C_1, t
C_2, t
C_n, t

Local Consistency

$$\frac{C, \mathsf{A}(s \,\mathsf{U}\, t)}{C | A(s \,\mathsf{U}_H\, t)}$$

- Sufficient criterion *local consistency*:

$$lc(C) \equiv \bot \notin C \wedge \{p, \neg p\} \not\subseteq C$$

$$lc(C | \,\mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}_H\, t)) \Longrightarrow lc(C, \mathsf{AX}\,\mathsf{A}(s \,\mathsf{U}\, t))$$

- Fragment labels:
  - Root and leafs: consistent, history-free, literal clauses ($\mathcal{D}$)
  - Internal nodes: locally consistent clauses

Tableau Calculus

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$\frac{}{C, p^+, p^- | a} \qquad \frac{}{C, \bot^+ | a} \qquad \frac{C, s^- | a \qquad C, t^+ | a}{C, s \to t^+ | a} \to^+ \qquad \frac{C, s^+, t^- | a}{C, s \to t^- | a} \to^-$$

$$\frac{\mathcal{R}\, C | r\, a}{C | a} \mathsf{X} \qquad \frac{\mathcal{R} C, u^- | r\, a}{C, \mathsf{AX}\, u^- | a} \mathsf{AX}^- \qquad \frac{\mathcal{R} C | \mathsf{A}(s\, \mathsf{U}_H\, t)^-}{C | \mathsf{A}^+(s\, \mathsf{U}_H\, t)^-} \mathsf{R}_\mathsf{H}^+$$

$$\frac{C, t^+ | a \qquad C, s^+, \mathsf{A}^+(s\, \mathsf{U}\, t)^+ | a}{C, \mathsf{A}(s\, \mathsf{U}\, t)^+ | a} \mathsf{U}^+ \qquad \frac{C, t^-, s^- | a \qquad C, t^-, \mathsf{A}^+(s\, \mathsf{U}\, t)^- | a}{C, \mathsf{A}(s\, \mathsf{U}\, t)^- | a} \mathsf{U}^-$$

$$\frac{C, s^+, t^+ | a \qquad C, t^+, \mathsf{A}^+(s\, \mathsf{R}\, t)^+ | a}{C, \mathsf{A}(s\, \mathsf{R}\, t)^+ | a} \mathsf{R}^+ \qquad \frac{C, t^- | a \qquad C, s^-, \mathsf{A}^+(s\, \mathsf{R}\, t)^- | a}{C, \mathsf{A}(s\, \mathsf{R}\, t)^- | a} \mathsf{R}^-$$

$$\frac{C | \mathsf{A}(s\, \mathsf{U}_\emptyset\, t)^+}{C, \mathsf{A}(s\, \mathsf{U}\, t)^+ | \cdot} \mathsf{A}_\emptyset \qquad \frac{C, t^+ | \cdot \qquad C, s^+ | \mathsf{A}^+(s\, \mathsf{U}_{H,C}\, t)^+}{C | \mathsf{A}(s\, \mathsf{U}_H\, t)^+} \mathsf{A}_\mathsf{H} \qquad \frac{}{C | \mathsf{A}(s\, \mathsf{U}_{H,C}\, t)^+} \overline{\mathsf{A}}$$

$$\frac{C | \mathsf{A}(s\, \mathsf{R}_\emptyset\, t)^-}{C, \mathsf{A}(s\, \mathsf{R}\, t)^- | \cdot} \mathsf{R}_\emptyset \qquad \frac{C, t^- | \cdot \qquad C, s^- | \mathsf{A}^+(s\, \mathsf{R}_{H,C}\, t)^-}{C | A(s\, \mathsf{R}_H\, t)^-} \mathsf{R}_\mathsf{H} \qquad \frac{}{C | \mathsf{A}(s\, \mathsf{R}_{H,C}\, t)} \overline{\mathsf{R}}$$

Example: Non-Compactness

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$\{\mathsf{E}(\top \,\mathsf{U}\, \neg p), \mathsf{AX}\, p, \mathsf{AX}(\mathsf{AX}\, p), \ldots\}$$