A Formal Completeness Proof for PDL

First Bachelor Seminar Talk

Joachim Bard
Advisor: Christian Doczkal

July 15, 2016

Outline

Propositional Dynamic Logic

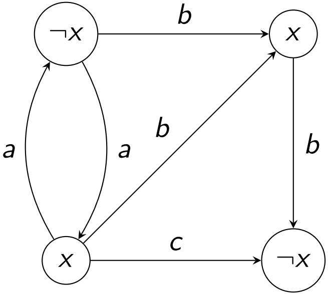
Definition

$$s, t ::= x \mid \bot \mid s \to t \mid [\alpha]s \qquad (x : \mathbb{N})$$
$$\alpha, \beta ::= a \mid \alpha + \beta \mid \alpha ; \beta \mid \alpha^* \mid s? \qquad (a : \mathbb{N})$$

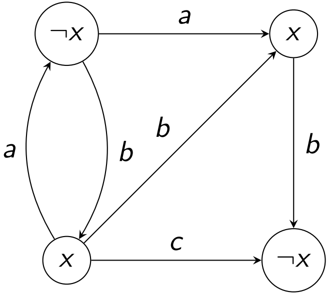
- extends classical propostional logic
- restrict to test-free PDL
- models are labeled transition systems
- $[\alpha]s$: at all α-reachable states s has to hold
- $\neg[\alpha]s$: there is some α-reachable state such that $\neg s$ holds
- $M, w \vDash s$: s holds at state w in model M

Example

$[a^*; b]x$



$\neg[a^*; b]x$



Hilbert System

Definition

$$\vdash s \to t \to s \qquad \vdash (u \to s \to t) \to (u \to s) \to u \to t$$

$$\vdash \neg\neg s \to s \qquad \frac{\vdash s \to t \qquad \vdash s}{\vdash t}$$

$$\vdash [\alpha](s \to t) \to [\alpha]s \to [\alpha]t \qquad \frac{\vdash s}{\vdash [\alpha]s}$$

$$\vdash [\alpha]s \to [\beta]s \to [\alpha + \beta]s \qquad \vdash [\alpha + \beta]s \to [\alpha]s$$

$$\vdash [\alpha + \beta]s \to [\beta]s \qquad \vdash [\alpha; \beta]s \to [\alpha][\beta]s$$

$$\vdash [\alpha][\beta]s \to [\alpha; \beta]s \qquad \vdash [\alpha^*]s \to s \qquad \vdash [\alpha^*]s \to [\alpha][\alpha^*]s$$

$$\frac{\vdash u \to [\alpha]u \qquad \vdash u \to s}{\vdash u \to [\alpha^*]s}$$

Completeness

Theorem (Completeness)

$(\forall M\, w.\, M, w \vDash s) \to \vdash s$

- adopt techniques in Christian's PhD thesis to PDL

Theorem (Informative Completeness)

$\{\vdash \neg s\} + \{\exists M\, w. M, w \vDash s\}$

- instance for $\neg s$ yields completeness
- now focus on model construction

Support

- similar to tableaux method
- decompose formulas into literals $(x, \bot, [a]s)$
- used later to construct models

$$C \triangleright s^{\sigma}$$

- negative sign serves as top level negation
- C is a set of signed formulas (clause)

Naive Attempt

Definition

$$
\begin{aligned}
C \triangleright s^\sigma &:= s^\sigma \in C && s \text{ literal} \\
C \triangleright s \to t^+ &:= C \triangleright s^- \parallel C \triangleright t^+ \\
C \triangleright s \to t^- &:= C \triangleright s^+ \mathbin{\&\&} C \triangleright t^- \\
C \triangleright [\alpha^*]s^+ &:= C \triangleright s^+ \mathbin{\&\&} C \triangleright [\alpha][\alpha^*]s^+
\end{aligned}
$$

- $C \triangleright [\alpha][\alpha^*]s^+$ is not structurally recursive
- results in divergence

Naive Attempt

Definition

$C \triangleright [\alpha^*]s^+ := C \triangleright s^+ \; \&\& \; C \triangleright [\alpha][\alpha^*]s^+$

$$
\begin{tikzcd}
& C \triangleright [a^{*^*}]s^+ \arrow[dl] \arrow[dr] & \\
C \triangleright s^+ & & C \triangleright [a^*][a^{*^*}]s^+ \arrow[ul, bend left] \arrow[d] \\
& & C \triangleright [a][a^*][a^{*^*}]s^+
\end{tikzcd}
$$

- decomposition does not terminate
- observation: right subgraph should not look behind any boxes

Support

Definition

$C \triangleright [\alpha]s^+ := (\textit{if}\, \varepsilon \in \mathcal{L}\,\alpha \textit{ then } C \triangleright s^+ \textit{ else true})\ \&\&\ C \triangleright_\Box [\alpha]s^+$

- $\varepsilon \in \mathcal{L}\,\alpha$ can be defined structurally on α
- $C \triangleright s^\sigma$ recursive on s

Definition

$$\begin{aligned} C \triangleright_\Box [a]s^\sigma &:= [a]s^\sigma \in C \\ C \triangleright_\Box [\alpha^*]s^+ &:= C \triangleright_\Box [\alpha][\alpha^*]s^+ \\ C \triangleright_\Box [\alpha;\beta]s^+ &:= C \triangleright_\Box [\alpha][\beta]s^+\ \&\&\ (\textit{if}\, \varepsilon \in \mathcal{L}\,\alpha \textit{ then } C \triangleright_\Box [\beta]s^+ \textit{ else true}) \end{aligned}$$

- $C \triangleright_\Box [\alpha]s^\sigma$ recursive on α

Support

Definition

$$C \triangleright [\alpha]s^+ := (\textit{if}\ \varepsilon \in \mathcal{L}\,\alpha\ \textit{then}\ C \triangleright s^+\ \textit{else true})\ \&\&\ C \triangleright_\Box [\alpha]s^+$$

$$C \triangleright_\Box [\alpha^*]s^+ := C \triangleright_\Box [\alpha][\alpha^*]s^+$$

$$\begin{tikzcd}
 & C \triangleright [a^{*^*}]s^+ \arrow[dl] \arrow[dr] & \\
C \triangleright s^+ & & C \triangleright_\Box [a^{*^*}]s^+ \arrow[d] \\
 & & C \triangleright_\Box [a^*][a^{*^*}]s^+ \arrow[d] \\
 & & C \triangleright_\Box [a][a^*][a^{*^*}]s^+
\end{tikzcd}$$

- solves nested-star problem

Correctness

Definition

$$C \triangleright [\alpha]s^+ := (\mathit{if}\, \varepsilon \in \mathcal{L}\, \alpha \ \mathit{then}\ C \triangleright s^+ \ \mathit{else\ true}) \ \&\& \ C \triangleright_\square [\alpha]s^+$$
$$C \triangleright_\square [\alpha^*]s^+ := C \triangleright_\square [\alpha][\alpha^*]s^+$$

Lemma

$C \triangleright [\alpha^*]s^+ = C \triangleright s^+ \ \&\& \ C \triangleright [\alpha][\alpha^*]s^+$

$$\begin{tikzcd}
 & C \triangleright [\alpha^*]s^+ \arrow[dl] \arrow[dr] & \\
C \triangleright s^+ & & C \triangleright_\square [\alpha^*]s^+ \arrow[d] \\
 & & C \triangleright_\square [\alpha][\alpha^*]s^+
\end{tikzcd}$$

$$\begin{tikzcd}
 & C \triangleright [\alpha][\alpha^*]s^+ \arrow[dl, dashed] \arrow[dr] & \\
C \triangleright [\alpha^*]s^+ & & C \triangleright_\square [\alpha][\alpha^*]s^+
\end{tikzcd}$$

Correctness

Lemma

$$
\begin{aligned}
C \triangleright [\alpha^*]s^+ &= C \triangleright s^+ \ \&\& \ C \triangleright [\alpha][\alpha^*]s^+ \\
C \triangleright [\alpha;\beta]s^+ &= C \triangleright [\alpha][\beta]s^+ \\
C \triangleright [\alpha+\beta]s^+ &= C \triangleright [\alpha]s^+ \ \&\& \ C \triangleright [\beta]s^+
\end{aligned}
$$

- analogously for negative signs

Demo

- model with clauses as states
- $C \triangleright s^\sigma \to C \models s^\sigma$
- $C \overset{a}{\Rightarrow} D := D \triangleright \mathcal{R}_a C$

Definition

$\mathcal{R}_a C := \{s^+ \mid [a]s^+ \in C\}$

- we need rules for $[\alpha]s^-$

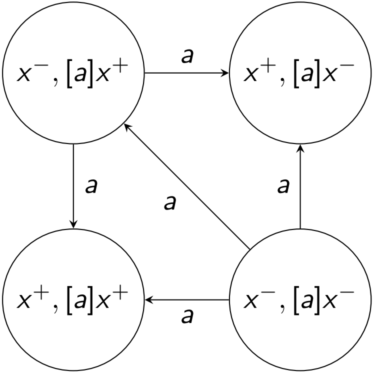
Definition

S is demo if:

- $\forall [a]s^- \in C \in S. \exists D \in S. D \triangleright \mathcal{R}_a C \wedge D \triangleright s^-$
- $\dots$

Pruning

$$\forall [a]s^- \in C \in S.\, \exists D \in S.\, D \triangleright \mathcal{R}_a C \wedge D \triangleright s^-$$



- start with a finite model
- successively remove states
- eventually arrive at a demo

References

- Christian Doczkal. *A Machine-Checked Constructive Metatheory of Computation Tree Logic*. PhD thesis, Saarland University, Mar 2016.
- Mark Kaminski. *Incremental Decision Procedures for Modal Logics with Nominals and Eventualities*. PhD thesis, Saarland University, Feb 2012.
- David Harel, Dexter Kozen, and Jerzy Tiuryn. *Dynamic Logic*. The MIT Press, 2000.

Thanks for your attention!
Questions?

Support

Definition

$$\begin{aligned}
C \rhd s^\sigma &:= s^\sigma \in C && s \text{ literal} \\
C \rhd s \to t^+ &:= C \rhd s^- \parallel C \rhd t^+ \\
C \rhd s \to t^- &:= C \rhd s^+ \;\&\&\; C \rhd t^- \\
C \rhd [\alpha]s^+ &:= (\mathit{if}\, \varepsilon \in \mathcal{L}\,\alpha \;\mathit{then}\; C \rhd s^+ \;\mathit{else\ true}) \;\&\&\; C \rhd_\Box [\alpha]s^+ \\
C \rhd [\alpha]s^- &:= (\mathit{if}\, \varepsilon \in \mathcal{L}\,\alpha \;\mathit{then}\; C \rhd s^- \;\mathit{else\ false}) \parallel C \rhd_\Box [\alpha]s^-
\end{aligned}$$

Definition

$$\begin{aligned}
C \rhd_\Box [a]s^\sigma &:= [a]s^\sigma \in C \\
C \rhd_\Box [\alpha^*]s^\sigma &:= C \rhd_\Box [\alpha][\alpha^*]s^\sigma \\
C \rhd_\Box [\alpha;\beta]s^+ &:= C \rhd_\Box [\alpha][\beta]s^+ \;\&\&\; (\mathit{if}\, \varepsilon \in \mathcal{L}\,\alpha \;\mathit{then}\; C \rhd_\Box [\beta]s^+ \;\mathit{else\ true}) \\
C \rhd_\Box [\alpha+\beta]s^+ &:= C \rhd_\Box [\alpha]s^+ \;\&\&\; C \rhd_\Box [\beta]s^+
\end{aligned}$$

Subformula Closure

Definition

$$
\begin{aligned}
\mathit{sfc}\; s^{\sigma} &:= \{s^{\sigma}\} \\
\mathit{sfc}\; s \to t^{\sigma} &:= \{s \to t^{\sigma}\} \cup \mathit{sfc}\; s^{\overline{\sigma}} \cup \mathit{sfc}\; t^{\sigma} \\
\mathit{sfc}\; [\alpha]s^{\sigma} &:= \{[\alpha]s^{\sigma}\} \cup \mathit{sfc}_{\Box}\; [\alpha]s^{\sigma}
\end{aligned}
$$

Definition

$$
\begin{aligned}
\mathit{sfc}_{\Box}\; [a]s^{\sigma} &:= \{[a]s^{\sigma}\} \\
\mathit{sfc}_{\Box}\; [\alpha^*]s^{\sigma} &:= \{[\alpha^*]s^{\sigma}\} \cup \mathit{sfc}_{\Box}\; [\alpha][\alpha^*]s^{\sigma} \\
\mathit{sfc}_{\Box}\; [\alpha;\beta]s^{\sigma} &:= \{[\alpha;\beta]s^{\sigma}\} \cup \mathit{sfc}_{\Box}\; [\alpha][\beta]s^{\sigma} \cup \mathit{sfc}_{\Box}\; [\beta]s^{\sigma} \\
\mathit{sfc}_{\Box}\; [\alpha+\beta]s^{\sigma} &:= \{[\alpha+\beta]s^{\sigma}\} \cup \mathit{sfc}_{\Box}\; [\alpha]s^{\sigma} \cup \mathit{sfc}_{\Box}\; [\beta]s^{\sigma}
\end{aligned}
$$