Topics in Concurrency

Lecture 5

Jonathan Hayman

23 February 2015

Specification logics

Logics for specifying correctness properties.
We'll look at:

- Basic logics and bisimilarity
- Fixed points and logic
- CTL
- Model checking

The model

A *transition system* is:

initial state   transition relation, $tran \subseteq S \times L \times S$

$$(S, i, L, tran)$$

set of states   set of labels

The model

A *transition system* is:

transition relation, $tran \subseteq S \times L \times S$

$$(S, \quad L, tran)$$

set of states

set of labels

A CCS term / process / state is finite state if the set of states reachable from it is finite.

Finitary Hennessy-Milner Logic

Assertions:

$$A ::= T \mid F \mid A_0 \wedge A_1 \mid A_0 \vee A_1 \mid \neg A \mid \langle \lambda \rangle A \mid \langle - \rangle A$$

Satisfaction: $s \models A$

Finitary Hennessy-Milner Logic

Assertions:

$$A ::= T \mid F \mid A_0 \wedge A_1 \mid A_0 \vee A_1 \mid \neg A \mid \langle \lambda \rangle A \mid \langle - \rangle A$$

Satisfaction: $s \vDash A$

$$\begin{array}{rl}
s \vDash T & \text{always} \\
s \vDash F & \text{never} \\
s \vDash A_0 \wedge A_1 & \text{if} \quad s \vDash A_0 \quad \text{and} \quad s \vDash A_1 \\
s \vDash A_0 \vee A_1 & \text{if} \quad s \vDash A_0 \quad \text{or} \quad s \vDash A_1 \\
s \vDash \neg A & \text{if} \quad \text{not} \quad s \vDash A \\
s \vDash \langle \lambda \rangle A & \text{if} \quad \text{there exists } s' \text{ s.t. } s \xrightarrow{\lambda} s' \text{ and } s' \vDash A \\
s \vDash \langle - \rangle A & \text{if} \quad \text{there exist } s', \lambda \text{ s.t. } s \xrightarrow{\lambda} s' \text{ and } s' \vDash A
\end{array}$$

Derived assertions

$$[\lambda]A \equiv \neg\langle \lambda \rangle \neg A \qquad [-]A \equiv \neg\langle - \rangle \neg A$$

Finitary Hennessy-Milner Logic

Assertions:

$$A ::= T \mid F \mid A_0 \wedge A_1 \mid A_0 \vee A_1 \mid \neg A \mid \langle \lambda \rangle A \mid \langle - \rangle A$$

Satisfaction: $s \models A$

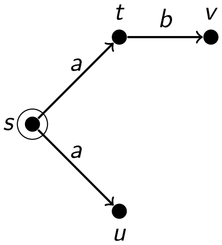
$$\begin{array}{rl}
s \models T & \text{always} \\
s \models F & \text{never} \\
s \models A_0 \wedge A_1 & \text{if} \quad s \models A_0 \quad \text{and} \quad s \models A_1 \\
s \models A_0 \vee A_1 & \text{if} \quad s \models A_0 \quad \text{or} \quad s \models A_1 \\
s \models \neg A & \text{if} \quad \text{not} \quad s \models A \\
s \models \langle \lambda \rangle A & \text{if} \quad \text{there exists } s' \text{ s.t. } s \xrightarrow{\lambda} s' \text{ and } s' \models A \\
s \models \langle - \rangle A & \text{if} \quad \text{there exist } s', \lambda \text{ s.t. } s \xrightarrow{\lambda} s' \text{ and } s' \models A
\end{array}$$

Derived assertions

$$[\lambda]A \equiv \neg\langle \lambda \rangle \neg A \qquad [-]A \equiv \neg\langle - \rangle \neg A$$

$$s \models [\lambda]A \text{ iff } \text{ for all } s' \text{ s.t. } s \xrightarrow{\lambda} s' \text{ have } s' \models A$$

Examples



$$? \; s \models \langle a \rangle T \; ?$$

$$? \; s \models [a] T \; ?$$

$$? \; u \models [-] F \; ?$$

$$? \; s \models \langle a \rangle \langle b \rangle T \; ?$$

$$? \; s \models [a] \langle b \rangle T \; ?$$

Examples

Generally:

- $\langle a \rangle T$
- $[a]F$
- $\langle - \rangle F$
- $\langle - \rangle T$
- $[-]T$
- $[-]F$

Give a transition system with initial state satisfying:

$$\langle - \rangle [a]F \wedge [a] < a > T$$

(Strong) bisimilarity and logic

A non-finitary Hennessy-Milner logic allows an infinite conjunction

$$A ::= \bigwedge_{i \in I} A_i \mid \neg A \mid \langle \lambda \rangle A$$

with semantics

$$s \vDash \bigwedge_{i \in A} A_i \text{ iff } s \vDash A_i \text{ for all } i \in I$$

Define

$$p \asymp q \quad \text{iff} \quad \text{for all assertions } A \text{ of H-M logic } p \vDash A \text{ iff } q \vDash A$$

Theorem

$$\asymp \quad = \quad \sim$$

This gives a way to demonstrate non-bisimilarity of states