Model Checking Infinite-state Systems in SAL

Bruno Dutertre, SRI International

Automated Formal Methods

FLoC Workshop

Seattle, August 21st 2006.

Outline

The DRT Example

Counter-based Model

- SAL model
- Property specification
- Verification

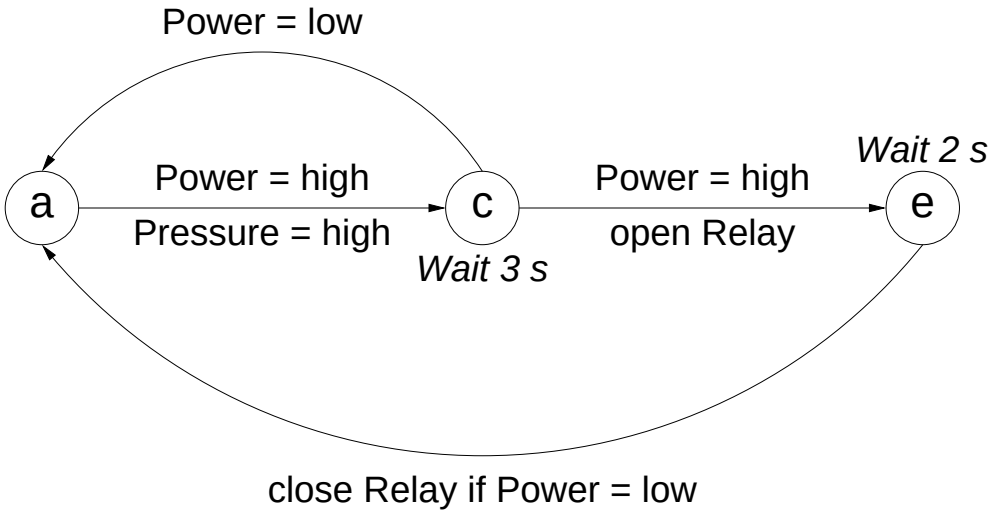
Timeout Automata Model

- Definition
- Application to DRT

References

DRT

Simplified Delayed Trip Reactor (inspired by Lawford and Zhang, *Equivalence Verification of Timed Transition Systems*, ACSD 2004)



Safety Property: if power and pressure at high at time t, and power is still high at $t + 30$ then the relay must be open for at least 20 time units, starting at some time in $[t + 30, t + 31]$. (time unit is 0.1 s)

Counter-Based Model

Need: model real-time delays

Approach:

- Discrete time and synchronouus composition
- One transition = one discrete time step = one time unit
- Integer-valued counters to model delays
- Finite model if all delays are bounded.

Application to DRT

Controller Model

Safety Property

- Specifying the property
- Analysis: smc, bmc, inf-bmc

A Weaker Property

Variant

- Reactor model and verification

k-induction

To show that a transition system $M = (X, I, T)$ satisfies $\Box P$

Usual induction

- Base case: $I(x) \Rightarrow P(x)$
- Induction step: $P(x) \wedge T(x, x') \Rightarrow P(x')$

k-induction

- Base case:

$$I(x_0) \wedge T(x_0, x_1) \wedge \ldots \wedge T(x_{k-2}, x_{k-1}) \Rightarrow P(x_0) \wedge \ldots \wedge P(x_{k-1})$$

- Induction step:

$$P(x_0) \wedge T(x_0, x_1) \wedge \ldots \wedge T(x_{k-2}, x_{k-1}) \wedge P(x_{k-1}) \wedge T(x_{k-1}, x_k) \Rightarrow P(x_k)$$

Usual induction is k-induction with $k = 1$

Proving $\Box P$ by k-induction is the same as proving $\Box(P \wedge \circ P \wedge \ldots \wedge \circ^{k-1} P)$ by induction

Limits of Counter Models

Expressiveness

- Not applicable to dense time

Verification Issues

- Lots of intermediate states where nothing happens (just counters get increased or decreased)
- BMC or induction depth depends on constants in the model (large depth for simple system may make SMC or BMC blow up)

Timeout-Based Model

State variables

- global time t and timeouts $\tau_1, \ldots, \tau_n$ (real-valued)
- discrete variables

τ_i stores a time in the future, where a discrete transition is scheduled to happen

$t \leqslant \tau_i$ is an invariant

Discrete Transitions

- Enabled when $t = \tau_i$ for some i
- Do not change t and must update τ_i to a value larger than t

Time-progress transitions

- Enabled when $t < \min(\tau_1, \ldots, \tau_n)$
- Increase t to $\min(\tau_1, \ldots, \tau_n)$

Application to DRT

SAL Model

- Controller
- Reactor
- Clock

Verification

- BMC: search for counterexamples
- k-induction: proof
- discovering auxiliary lemmas

To Get More Information

SAL and Yices

- http://sal.csl.sri.com & http://sal-wiki.csl.sri.com
- http://yices.csl.sri.com & http://yices-wiki.csl.sri.com

Infinite & Timed Systems in SAL

- B. Dutertre and M. Sorea, *Modeling and Verification of a Fault-Tolerant Real-time Startup Protocol using Calendar Automata*, FORMATS/FTRTFT 2004 (http://www.csl.sri.com/~bruno/publis/startup.pdf)
- B. Dutertre and M. Sorea, *Timed Systems in SAL*, Technical Report, SRI-SDL-04-03, July 2004. (http://www.csl.sri.com/~bruno/publis/sri-sdl-04-03.pdf)
- L. Pike and S. Johnson, *The Formal Verificaiton of a Reintegration Protocol*, EMSOFT'05, (http://www.cs.indiana.edu/~lepike/pub_pages/emsoft.html)
- G. Brown and L. Pike. *Easy parameterized verification of biphase and 8N1 protocols*, TACAS'06, (http://www.cs.indiana.edu/~lepike/pub_pages/bmp.html)
- G. Brown and L. Pike. *"Easy" parameterized verification of cross clock domain protocol*, DCC'06, (http://www.cs.indiana.edu/~lepike/pub_pages/dcc.html)