INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE | INRIA | centre de recherche GRENOBLE - RHÔNE-ALPES

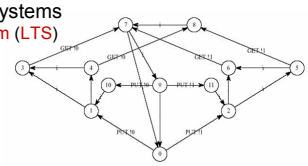
VASY

Bisimulator 2.0: An On-the-Fly Equivalence Checker based on Boolean Equation Systems

Radu Mateescu and Emilie Oudot
INRIA / VASY project-team, Faculté des Sciences Mirande, bât. LE2I, F-21000 Dijon, France
{Radu.Mateescu,Emilie.Oudot}@inria.fr

Context

- *Equivalence checking* of concurrent finite-state systems
  - Behaviour represented as a Labeled Transition System (LTS)
  - Compare the LTSs of a *protocol* and of its *service* modulo some equivalence relation



- *On-the-fly verification*
  - Incremental, demand-driven exploration of LTSs
  - Fast detection of errors in complex systems
  - Problem encoded as the local resolution of a Boolean Equation System (BES)

- **CADP** (*Construction and Analysis of Distributed Processes*) toolbox
  - Process algebraic input languages (LOTOS, EXP, FSP, CHP, …)
  - Compilation and rapid prototyping
  - Interactive and guided simulation
  - Model checking (modal μ-calculus) and equivalence checking (bisimulations)
  - Test generation
  - Performance evaluation (Interactive Markov Chains)
- http://www.inrialpes.fr/vasy/cadp

Two enhancements

1. New BES encodings of weak equivalence relations

BES encoding the comparison of two LTSs $M_1 = \langle Q_1, A_1, T_1, q_{01} \rangle$ and $M_2 = \langle Q_2, A_2, T_2, q_{02} \rangle$ modulo branching bisimulation

Basic variant:

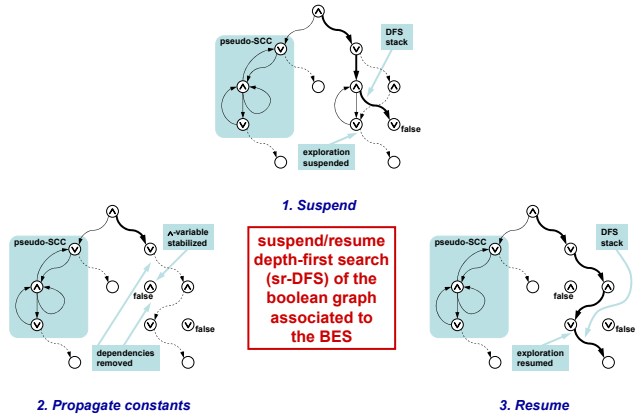
$$\left\{ \begin{array}{l} X_{pq} \stackrel{\nu}{=} \bigwedge_{p \stackrel{a}{\rightarrow} p'} ((a = \tau \wedge X_{p'q}) \vee \bigvee_{q \stackrel{\tau^*}{\rightarrow} q' \stackrel{a}{\rightarrow} q''} (X_{pq'} \wedge X_{p'q''})) \\ \qquad \wedge \\ \qquad \bigwedge_{q \stackrel{a}{\rightarrow} q'} ((a = \tau \wedge X_{pq'}) \vee \bigvee_{p \stackrel{\tau^*}{\rightarrow} p' \stackrel{a}{\rightarrow} p''} (X_{p'q} \wedge X_{p'q'})) \end{array} \right\}_{\substack{p,p',p'' \in Q_1, \\ q,q',q'' \in Q_2, \\ a \in A_1 \cup A_2}}$$

Enhanced variant:

$$\left\{ \begin{array}{l} X_{pq} \stackrel{\nu}{=} \bigwedge_{p \stackrel{a}{\rightarrow} p'} Y_{pp'qa} \wedge \bigwedge_{q \stackrel{a}{\rightarrow} q'} Z_{pqq'a} \\ Y_{pp'qa} \stackrel{\nu}{=} (a = \tau \wedge X_{p'q}) \vee U_{pp'qa} \\ Z_{pqq'a} \stackrel{\nu}{=} (a = \tau \wedge X_{pq'}) \vee V_{pqq'a} \\ U_{pp'qa} \stackrel{\nu}{=} \bigvee_{q \stackrel{a}{\rightarrow} q'} W_{pp'qq'} \vee \bigvee_{q \stackrel{\tau}{\rightarrow} q'} U_{pp'q'a} \\ V_{pqq'a} \stackrel{\nu}{=} \bigvee_{p \stackrel{a}{\rightarrow} p'} W_{pp'qq'} \vee \bigvee_{p \stackrel{\tau}{\rightarrow} p'} V_{p'qq'a} \\ W_{pp'qq'} \stackrel{\nu}{=} X_{pq} \wedge X_{p'q'} \end{array} \right\}_{\substack{p,p',p'' \in Q_1, \\ q,q',q'' \in Q_2, \\ a \in A_1 \cup A_2}}$$

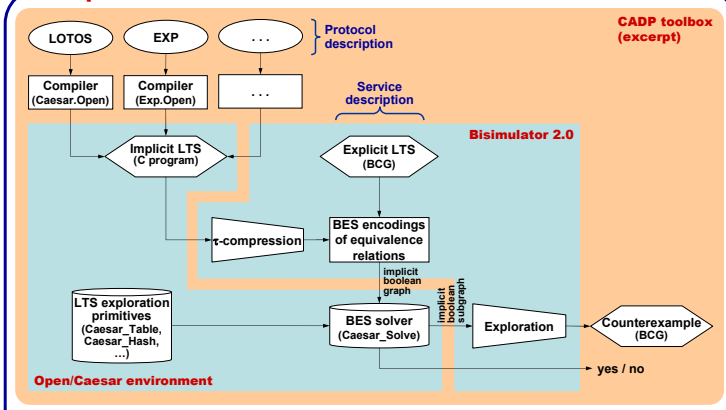
- *Alternation-free* BES (one block of maximal fixed point equations)
- Transitive closures over τ-transitions computed using BES equations
- On-the-fly LTS reduction using *τ-compression* (elimination of τ-cycles)

2. New local BES resolution algorithm based on suspend/resume DFS



- Detection of *counterexamples* by propagating false constants
- Detection of *examples* by suspending the DFS in search of pseudo-SCCs
- Stops as soon as an example or a counterexample was encountered

Implementation: Bisimulator 2.0

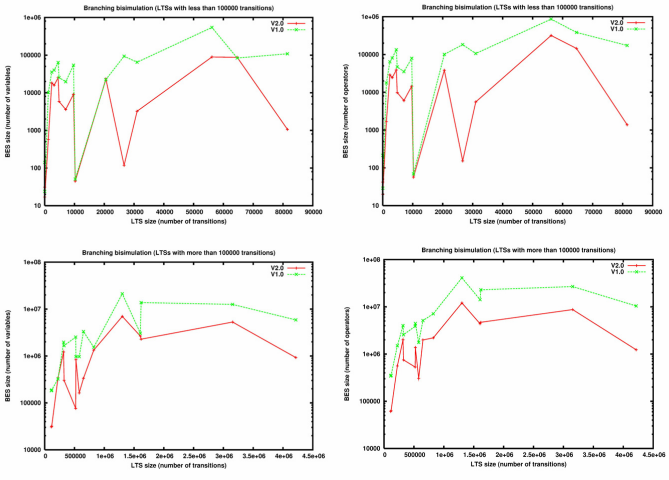


- Tool developed using the generic **Open/Caesar** environment for on-the-fly LTS exploration
- Seven equivalence relations and their preorders:
  - Strong, observational, branching, τ*.*a*
  - Safety, trace, weak trace
- On-the-fly LTS reduction by *τ-confluence*
- Verification engine: the generic **Caesar_Solve** library for on-the-fly BES resolution

Alg.	Type of BES	Strategy
A0	general	DFS
A1		BFS
A2	acyclic	DFS
A3	disjunctive	
A4	conjunctive	
A5	general	
A6	disjunctive, unique resolution	BFS
A7	conjunctive, unique resolution	
A8	general	sr-DFS

Experiments

- LTSs taken from the CADP demo examples and the **VLTS** benchmarks (http://www.inrialpes.fr/vasy/cadp/resources/benchmark.html)
- Reductions in BES size (number of variables and operators) induced by the new encoding and the sr-DFS algorithm
- Significant speed improvements for branching bisimulation (about one order of magnitude)



Further applications

- Extend Bisimulator 2.0 with other weak equivalence relations (*testing equivalence*, *CFFD-equivalence*) encoded as BESs
- Employ the sr-DFS algorithm for on-the-fly LTS reductions modulo *weak τ-confluence*, encoded using BES resolution
- Apply the BES technology to specific reductions of automata describing the dynamic behaviour of *Genetic Regulatory Networks* (European project **EC-MOAN**)

ec-moan