BISIMULATOR: A Modular Tool for On-the-Fly Equivalence Checking

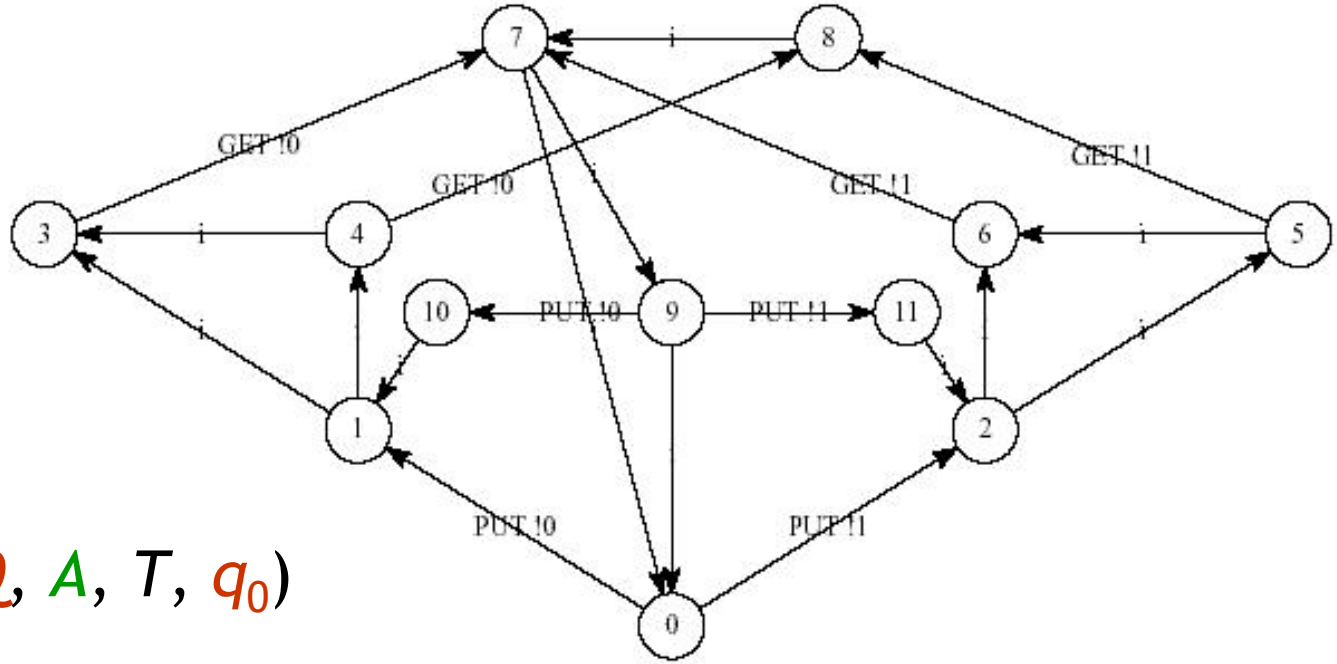
Damien Bergamini, Nicolas Descoubes, Christophe Joubert, and Radu Mateescu

INRIA Rhône-Alpes / VASY
655, avenue de l'Europe
F-38330 Montbonnot Saint Martin, France
http://www.inrialpes.fr/vasy

Outline

- Introduction
- Boolean equation systems
- Equivalence relations
- Tool architecture
- Demo
- Conclusion and future work

Labelled Transition Systems

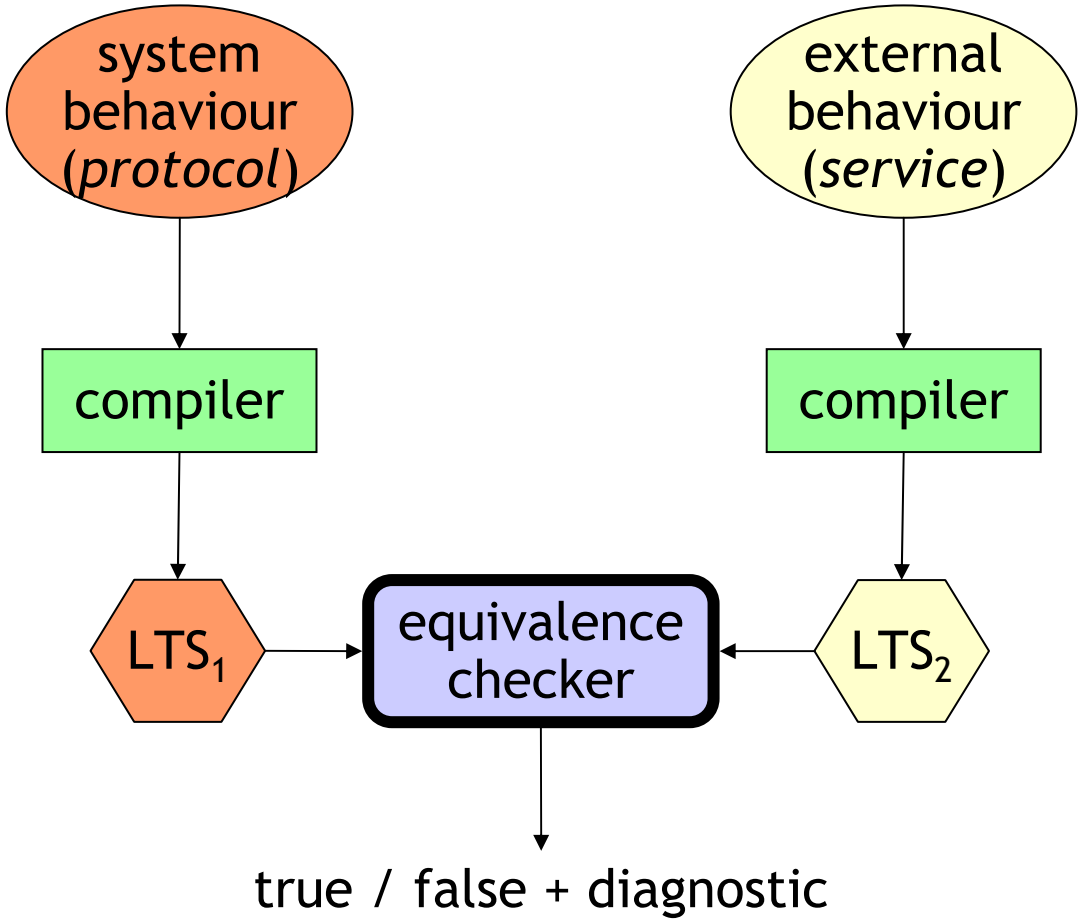


$M = (Q, A, T, q_0)$

CADP toolbox (http://www.inrialpes.fr/vasy/cadp)

- Explicit representation (succ/pred function)
  - BCG (Binary Coded Graphs)
- Implicit representation (successor function)
  - OPEN/CAESAR [Garavel-98]

Equivalence checking



Global

- LTS built *before* check
- Partition refinement
- Better when check OK

On-the-fly

- LTS built *during* check
- Synchronous product
- Better when check KO

On-the-fly equivalence checking

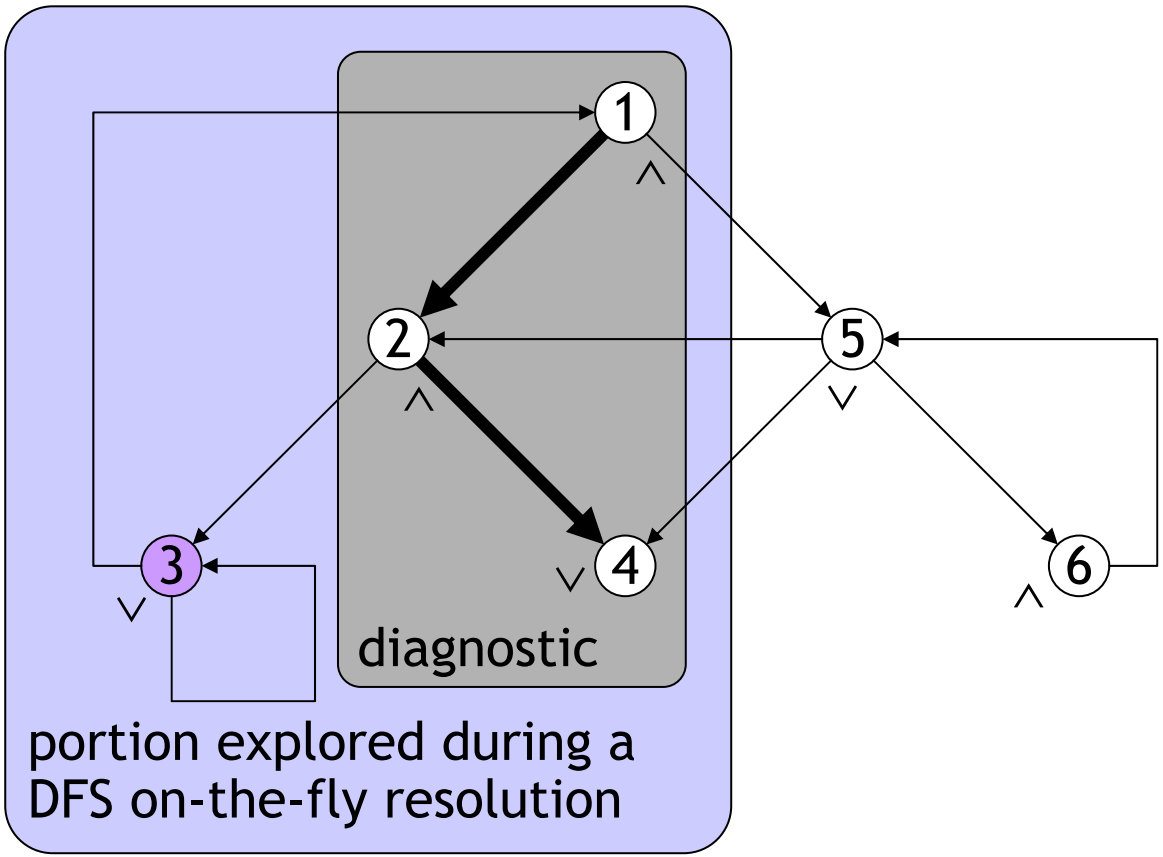
- Direct approaches
  - [Fernandez-Mounier-91,Cleaveland-Sokolsky-01]
    - On-the-fly equivalence checking algorithms
- Temporal-logic based approaches
  - [Cleaveland-Steffen-91]
    - Modal μ-calculus encoding of strong & observational equivalences
  - [Ingolfsdottir-Steffen-91,Fantechi-Gnesi-et-al-92]
    - Characteristic μ-calculus / ACTL formulas for strong & observational equivalences
- Boolean equation system / game graph based approaches
  - [Andersen-Vergauwen-95]
    - Branching equivalence encoding using BESs of alternation depth 2
  - [Stevens-Stirling-97,Bollig-Leucker-Weber-01]
    - Game graphs (mainly used for model checking of modal μ-calculus)

Alternation-free Boolean Equation Systems

BES

$$\begin{cases} X_1 =_\nu X_2 \wedge X_5 \\ X_2 =_\nu X_3 \wedge X_4 \\ X_3 =_\nu X_1 \vee X_3 \\ X_4 =_\nu F \\ X_5 =_\nu X_2 \vee X_4 \vee X_6 \\ X_6 =_\nu X_5 \end{cases}$$

boolean graph



[Andersen-94]
[Mader-97]

Strong equivalence

- $M_1 = (Q_1, A, T_1, q_{01})$, $M_2 = (Q_2, A, T_2, q_{02})$
  $\approx \subseteq Q_1 \times Q_2$ is the maximal relation s.t. $p \approx q$ iff

  $\forall a \in A. \forall p \rightarrow_a p' \in T_1. \exists q \rightarrow_a q' \in T_2. p' \approx q'$

  and

  $\forall a \in A. \forall q \rightarrow_a q' \in T_2. \exists p \rightarrow_a p' \in T_1. p' \approx q'$

- $M_1 \approx M_2$ iff $q_{01} \approx q_{02}$

Translation to BES

- Principle: $p \approx q$ iff $X_{p,q}$ is true
- General BES:

$$\begin{cases} X_{p,q} =_\nu (\wedge_{p \to a\, p'} \vee_{q \to a\, q'} X_{p',q'}) \\ \qquad\qquad \wedge \\ \qquad\qquad (\wedge_{q \to a\, q'} \vee_{p \to a\, p'} X_{p',q'}) \end{cases}$$

- Simple BES:

$$\begin{cases} X_{p,q} =_\nu (\wedge_{p \to a\, p'} Y_{a,p',q}) \wedge (\wedge_{q \to a\, q'} Z_{a,p,q'}) \\ Y_{a,p',q} =_\nu \vee_{q \to a\, q'} X_{p',q'} \\ Z_{a,p,q'} =_\nu \vee_{p \to a\, p'} X_{p',q'} \end{cases}$$

$p \leq q$ (preorder)

Tau*.a and safety equivalences

- $M_1 = (Q_1, A_\tau, T_1, q_{01})$, $M_2 = (Q_2, A_\tau, T_2, q_{02})$
  $A_\tau = A \cup \{ \tau \}$
- $\tau^*.a$ equivalence:

$$\begin{cases} X_{p,q} =_\nu (\wedge_{p \rightarrow \tau^*.a\ p'} \vee_{q \rightarrow \tau^*.a\ q'} X_{p',q'}) \\ \qquad\qquad \wedge \\ \qquad\qquad (\wedge_{q \rightarrow \tau^*.a\ q'} \vee_{p \rightarrow \tau^*.a\ p'} X_{p',q'}) \end{cases}$$

- Safety equivalence:

$$\begin{cases} X_{p,q} =_\nu Y_{p,q} \wedge Y_{q,p} \\ Y_{p,q} =_\nu \wedge_{p \rightarrow \tau^*.a\ p'} \vee_{q \rightarrow \tau^*.a\ q'} Y_{p',q'} \end{cases}$$

Observational and branching equivalences

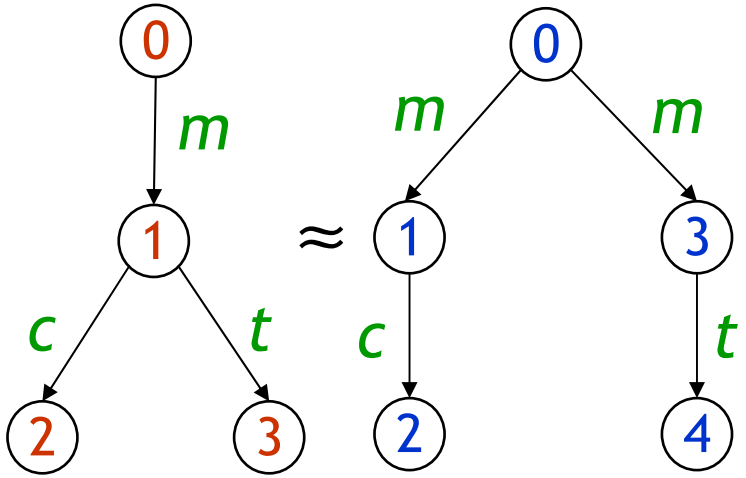
- Observational equivalence:

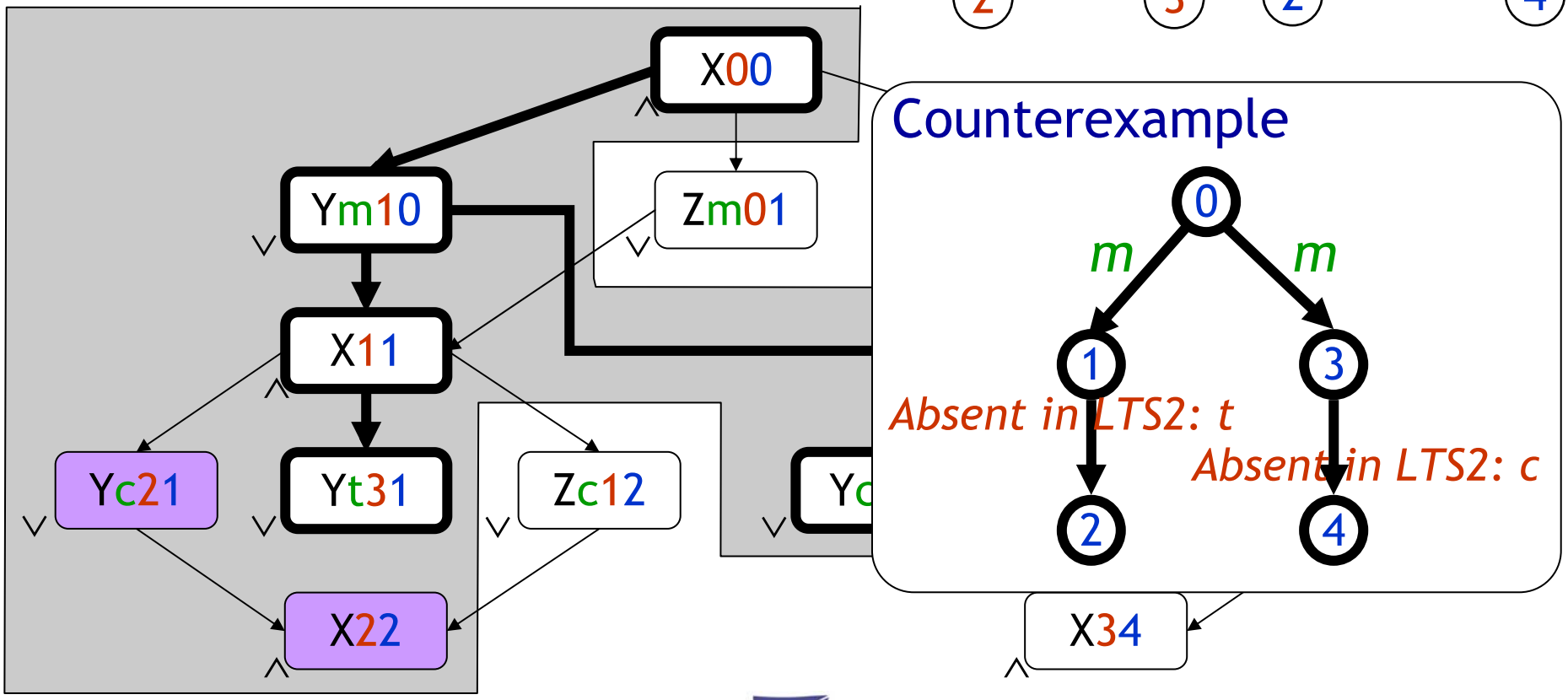
$$\begin{cases} X_{p,q} =_\nu (\wedge_{p \rightarrow\tau p'} \vee_{q \rightarrow\tau^* q'} X_{p',q'}) \wedge (\wedge_{p \rightarrow a p'} \vee_{q \rightarrow\tau^*.a.\tau^* q'} X_{p',q'}) \\ \qquad \wedge \\ \qquad (\wedge_{q \rightarrow\tau q'} \vee_{p \rightarrow\tau^* p'} X_{p',q'}) \wedge (\wedge_{q \rightarrow a q'} \vee_{p \rightarrow\tau^*.a.\tau^* p'} X_{p',q'}) \end{cases}$$

- Branching equivalence:

$$\begin{cases} X_{p,q} =_\nu \wedge_{p \rightarrow b p'} ((b{=}\tau \wedge X_{p',q}) \vee \vee_{q \rightarrow\tau^* q' \rightarrow b q''} (X_{p,q'} \wedge X_{p',q''}) \\ \qquad \wedge \\ \qquad \wedge_{q \rightarrow b q'} ((b{=}\tau \wedge X_{p,q'}) \vee \vee_{p \rightarrow\tau^* p' \rightarrow b p''} (X_{p',q} \wedge X_{p'',q'}) \end{cases}$$

Verification (strong equivalence)

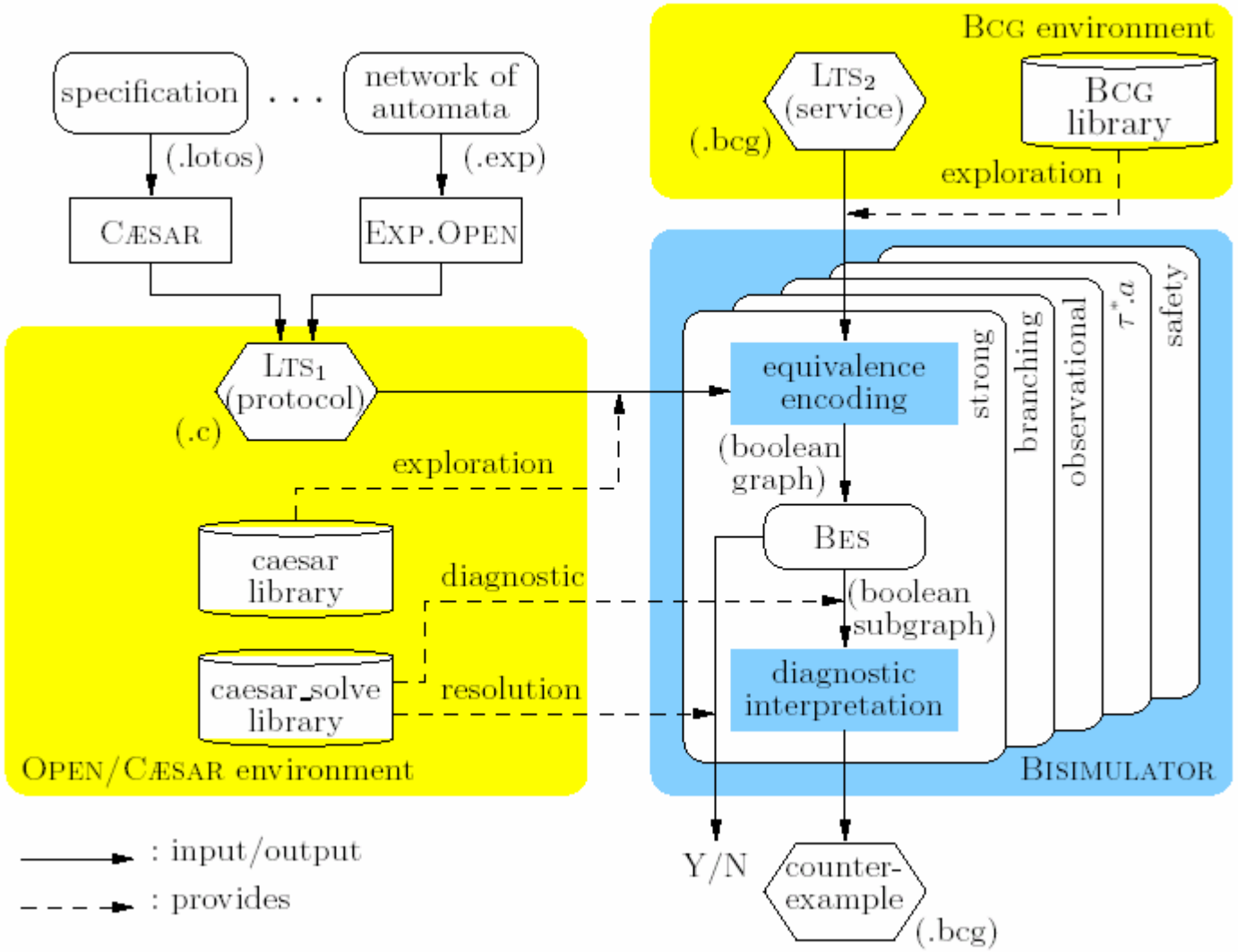




CAESAR_SOLVE library

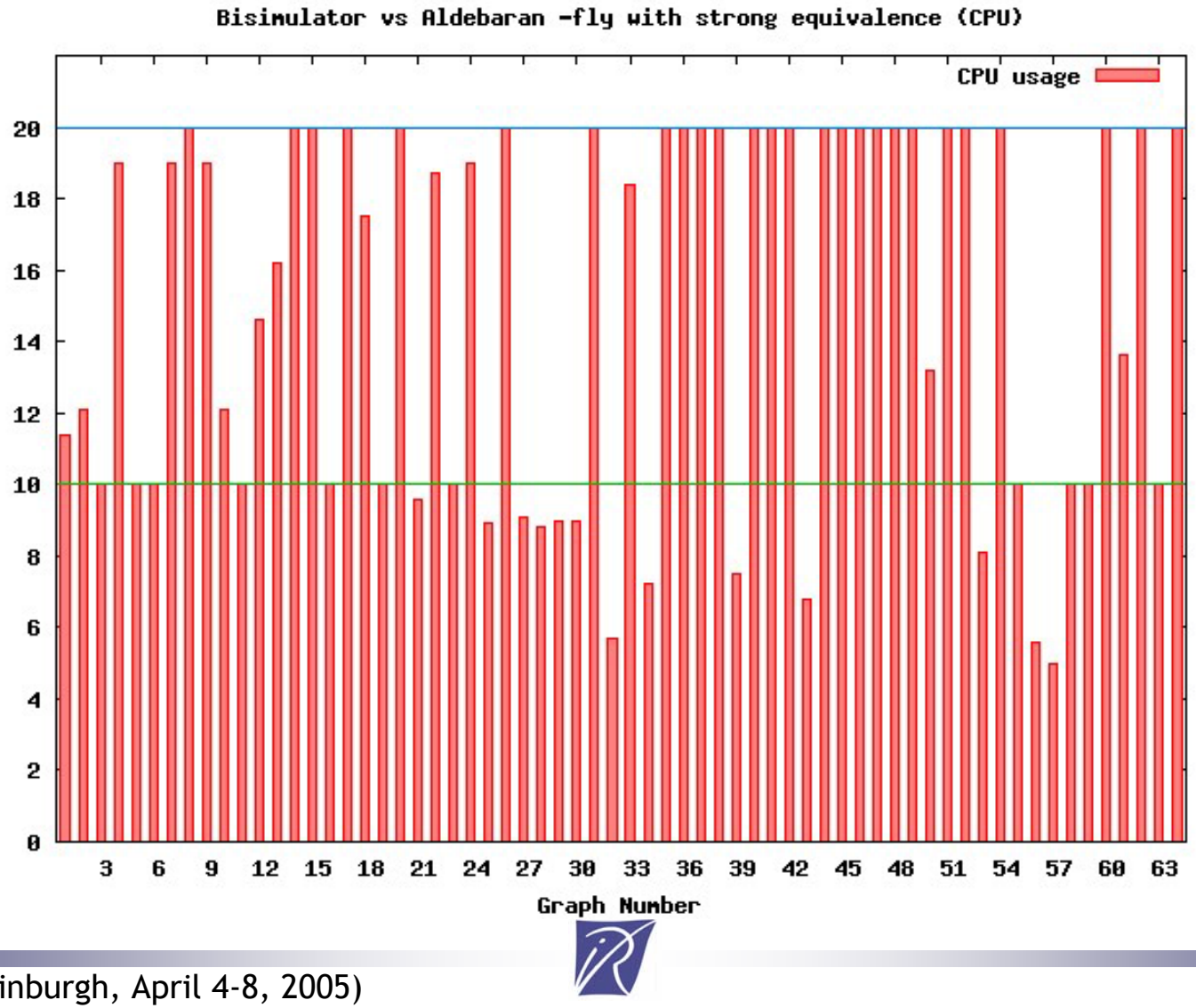
- On-the-fly resolution of alternation-free BESs [Mateescu-03]
- Developed in CADP using OPEN/CAESAR
- 4 linear-time sequential algorithms (10,000 lines of C)
  - DFS and BFS for general BESs
  - DFS memory-efficient for acyclic or conjunctive/disjunctive BESs
- 1 linear-time distributed algorithm (10,000 lines of C) [Joubert-Mateescu-04]
- Diagnostics (boolean subgraphs) [Mateescu-00]
- Generic, application-independent

Architecture of BISIMULATOR



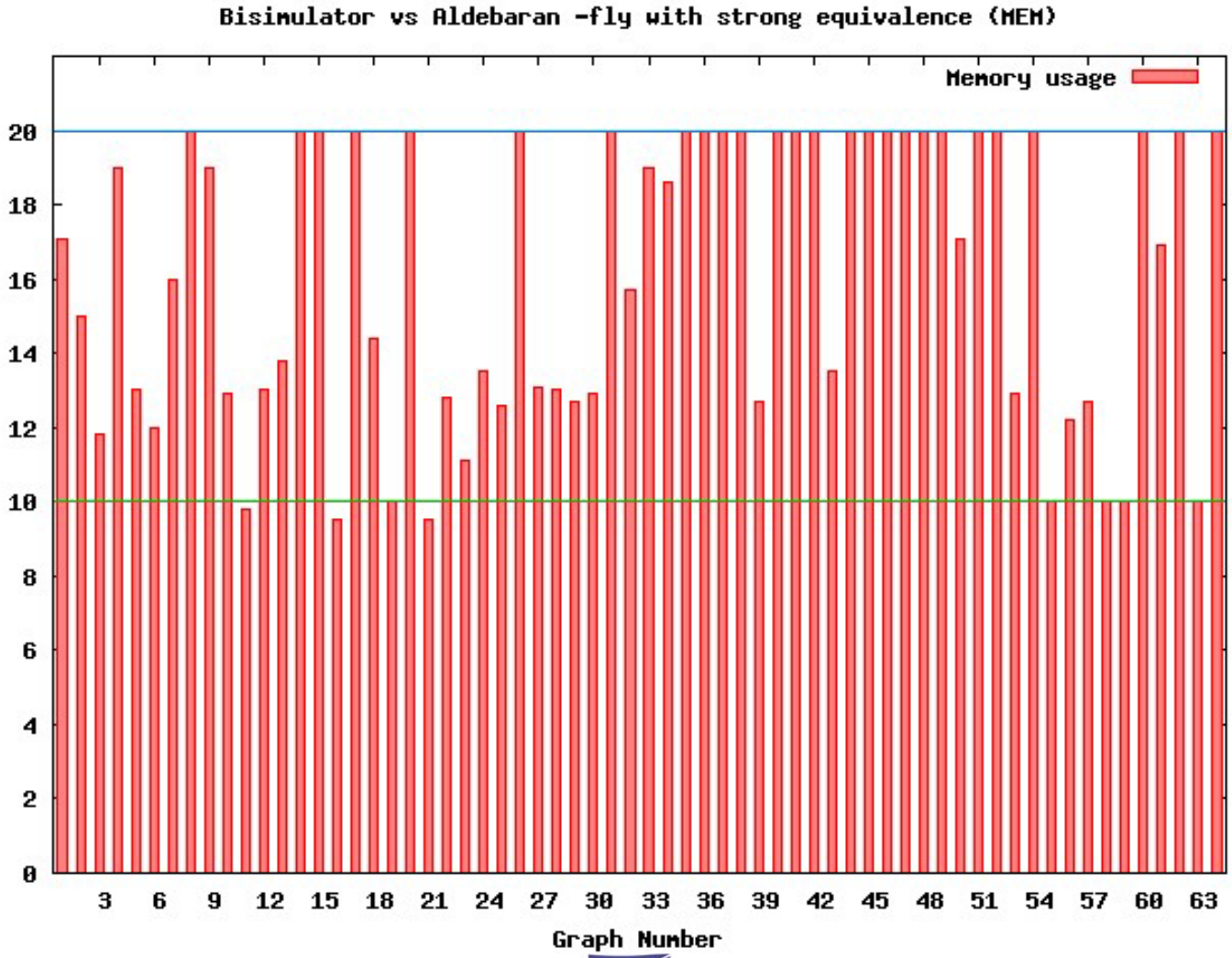
BISIMULATOR vs Aldébaran (1/2)

(VLTS benchmark suite)

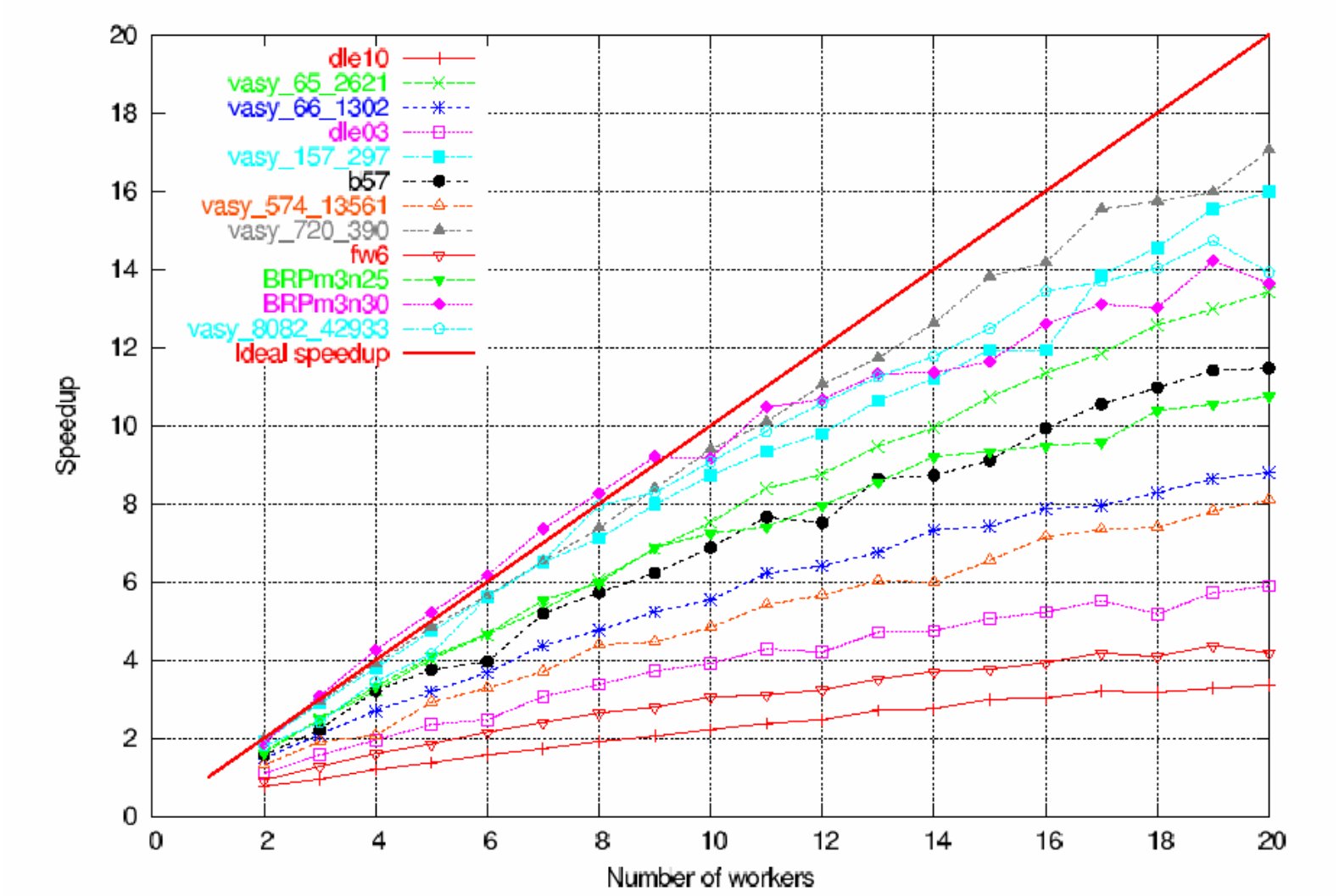


BISIMULATOR vs Aldébaran (2/2)

(VLTS benchmark suite)



Distributed vs sequential BISIMULATOR



Demo

Conclusion and future work

- Already done
  - Technology for on-the-fly equivalence checking
    - Highly modular (one module / equivalence)
    - Natural optimizations based on BES manipulation
    - Generic BES library CAESAR_SOLVE [Mateescu-03]
    - Distributed resolution [Joubert-Mateescu-04]
  - BISIMULATOR
    - Integrated in CADP
    - Language-independent (OPEN/CAESAR)
- Ongoing work
  - Encoding of other equivalences
    - Markovian bisimulation [Hermanns-Siegle-99]
  - Study of other BES resolution strategies