Verification of Distributed Systems with the toolkit VerICS

Wojciech Penczek

a joint work with VERICS team:

M. Kacprzak, W. Nabiałek, A. Niewiadomski, A. Półrola, M. Szreter, B. Wozna and A. Zbrzezny

ICS, Polish Academy of Sciences

Advanced Course on Petri Nets, Rostock, September 2010
We have to verify software

On June 4, 1996 ...

... an unmanned Ariane 5 rocket launched by the European Space Agency exploded just forty seconds after its lift-off. The rocket was on its first voyage, after a decade of development costing $7 billion. The destroyed rocket and its cargo were valued at $500 million.
How expensive is NOT TO VERIFY software

- Bug in the division module of Pentium II - $475 millions,

- Bug in the luggage service system in Denver (9 months delay in the opening of the airport) - $1.1 million a day,

- Bug in the radiotherapy system THERAC-25 caused death of 6 patients in 1985-87.
Why verification of distributed systems is so difficult?

- Hand verification is impossible in practice due to complexity of systems,
- Model checking is NP-hard or more difficult,
- The high complexity causes the state explosion problem - state spaces of distributed systems grow exponentially with the number of processes.

Solution: Symbolic model checking over a part of the state space of an abstracted system.
The toolkit VerICS

**Introduction**

- **VerICS** - a model checker for high-level languages, real-time distributed and multi-agent systems,

- **Input languages:** Time Petri Nets, Timed Automata, subsets of Estelle, UML, Java, and Promela.

- Various classes of properties can be verified: reachability, CTL, TCTL, TCTLK,

- **SAT-based** and abstraction-based enumerative model checking methods are exploited.
Main features of VerICS

- SAT-based BMC for **branching time properties** of Petri nets and (timed) automata,
- SAT-based verification of Java, UML, and Promella via translation to timed automata or directly to SAT,
- SAT-based verification of temporal-epistemic properties of multi-agent systems,
- SAT-based parametric reachability verification.
Plan

- Presentation of Verics main functionalities,
- Parametric verification of Mutex,
- Parametric reachability for timed Mutex.
DEMO of VERICS