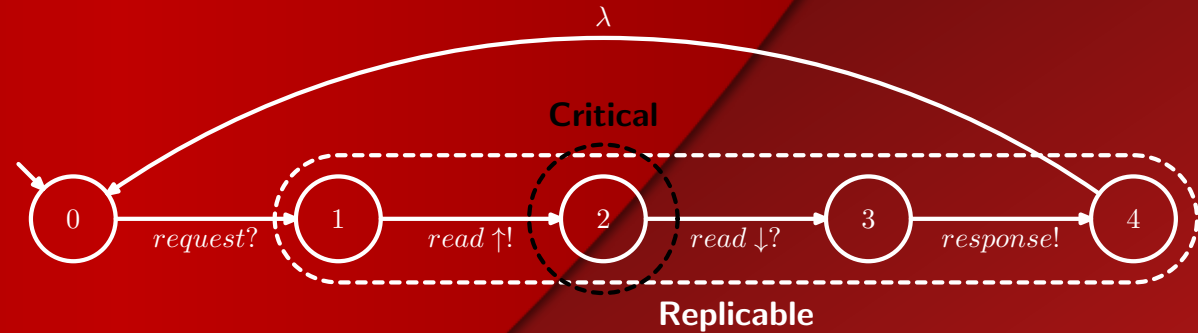# Analysis of *Interface Automata* with On-Demand Replication

Author: Mgr. Jakub Daniel   **Supervisor: RNDr. Jan Kofroň Ph.D.**

Institution
Faculty of Mathematics and Physics
Charles University in Prague

Motivation

Interface Automata are a formalism for modelling behaviour of components through combination of required and provided interfaces. Some of their notable properties are

+ well-established formalism,
+ straightforward use in model checking,
+ composition,
+ refinement,
+ orientation toward open systems,
− lack of support for unbounded threading.

Component behaviour with potentially unbounded degree of parallelism tends to result in an infinite model when subjected to explicit-state model checking.

The goal of the thesis was to propose a method based on interface automata allowing one to capture behaviour of common components and to

▶ allow their replication to an arbitrary degree of parallelism,
▶ allow compositions of the resulting models with compatible counterparts.

Challenges

▶ Separation of replicable and non-replicable parts of models
▶ Extension of the standard formalism to support repeated application of operations

Solution

▶ Relaxed *Interface Automaton* definition for intermediate results
▶ Annotations dividing model into non-replicable, replicable, and critical sections
▶ Replication operation
▶ Revised composition operation ensuring compliance with standard *Interface Automaton* definition

Properties

▶ Finiteness of models
▶ Repeatability of operations
▶ Selective replication
▶ Basic synchronisation awareness
▶ Running composition avoiding message ambiguity and unnecessary state space growth

References

1. L. de Alfaro, T. A. Henzinger. "Interface Automata". In Proceedings of FSE 2001, ACM.
2. J. Adámek. "Verification of Software Components: Addressing Unbounded Parallelism". IJCIS, 8(2), 2007.
3. F. Plášil, S. Višňovský. "Behavior Protocols for Software Components". IEEE TSE, 28(11), 2002.
4. T. Poch et al. "Threaded Behavior Protocols". Formal Aspects of Computing, 25(4), 2011.