Challenges in the Analysis and Validation of Software Systems

Motivation

Reactive systems appear everywhere, e.g., as web services, decision support systems, or logical controllers. Their validation techniques are as diverse as their appearance and structure. These comprise various forms of static analysis, model checking, symbolic execution and (model-based) testing, often tailored to quite extreme frame conditions. Thus it is almost impossible to compare these techniques, let alone to establish clear application profiles as a means for recommendation.

The RERS Challenges aim at overcoming this situation by providing a forum for experimental profile evaluation based on specifically designed Benchmark suites. These benchmarks are automatically synthesized to exhibit chosen properties, and then enhanced to include dedicated dimensions of difficulty, ranging from conceptual complexity of the properties (e.g. reachablity, full safety, liveness), over size of the reactive systems (a few hundred lines to tens of thousands of them), to exploited language features (arrays and arithmetic at index pointer).

Characteristic for RERS is its wide scope, which addresses not only source code analyzers (white-box problems), but also (model-based) testers and (test-based) modeler (black-box problems), and in particular free stylers. Currently, RERS focuses on functional properties only, but non-functional properties like time, performance, and stochastical behavior are envisaged.

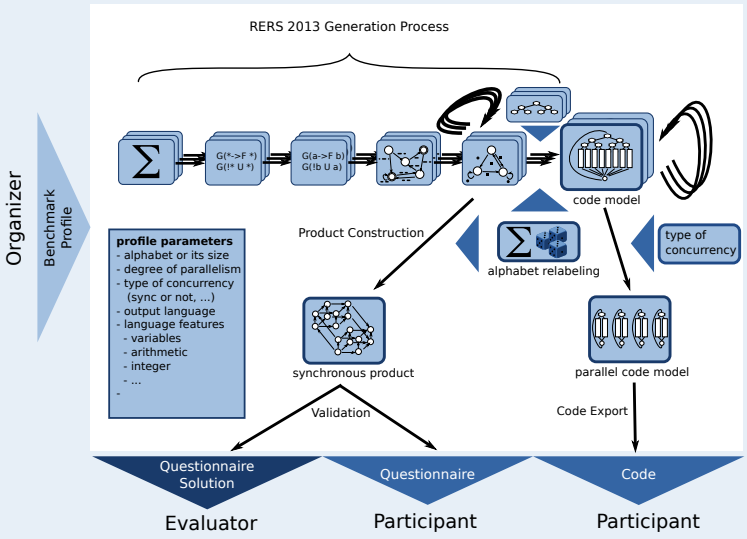
The Competition 2014

In 2014 we will address five independent categories of challenges. The problems of four of these categories will be generated automatically in a similar fashion as in 2013. The last category will consist of operational code, patched to exhibit certain reachability.

Categories for 2014 Competition:

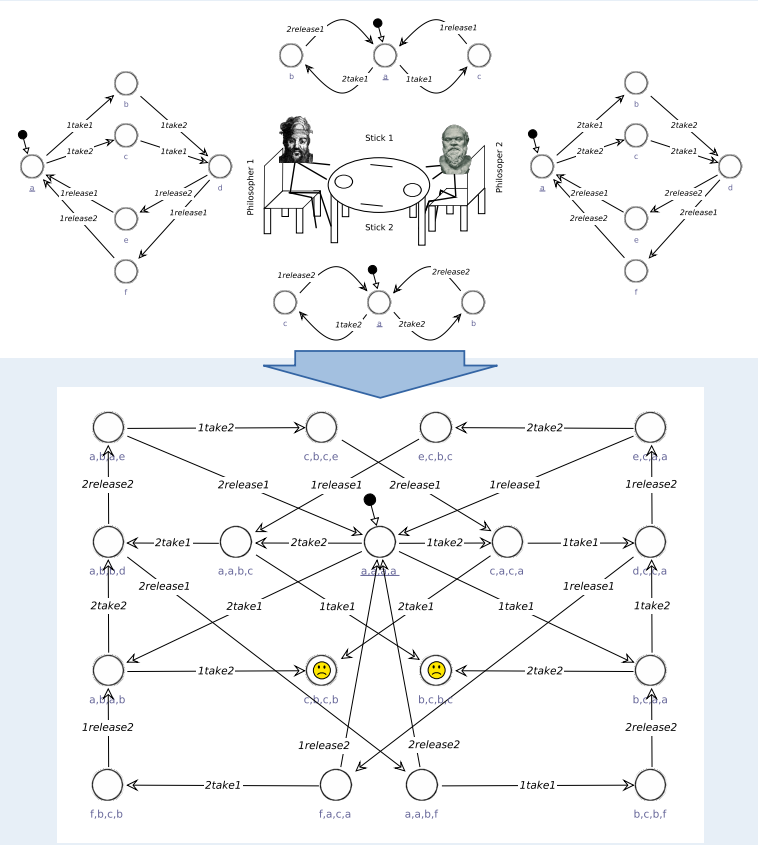
1) White-Box (basic, see 2013)	easy, middle, hard
2) White-Box (extended)	pointer arithmetics
3) Black-Box	execution based analysis
4) Protocols and Concurrent System	easy, middle, hard (Promela)
5) Real Problem	unrestricted

Benchmark Generation 2014



Generation of Concurrent Benchmark Programs

Concurrent benchmark code in Promela will be automatically generated via Property-driven [SPIN13] Benchmark Generation. The validation principle is illustrated along the well-known dining philosopher example below: the property profile (e.g., reachable deadlocks) is determined on the product automata for the concurrent systems. Independently, Promela code of varying complexity is generated from the concurrent systems in a sequence of property-preserving steps.



Previous Events

RERS 2013: co-located with ASE in Palo Alto, California
RERS 2012: co-located with ISoLA 2012 in Crete, Greece
RERS 2010: co-located with ISoLA 2010 in Crete, Greece

Organization / Contact

- ***April 1st - May 15th*** Training phase with problems suitable for classrooms
- ***May 15th - August 15th*** White-Box (basic), Black-Box
- ***June 15th - September 15th*** Protocols and Concurrent Systems
- ***July 15th - September 15th*** White-Box (extended), Real Problem
- ***September 15th*** Deadline for submissions of achievement contributions
- **October 8th - 11th**: RERS Event during ISoLA

ISoLA
Tiziana Margaria
Bernhard Steffen

RERS
Falk Howar
Jaco van de Pol
Markus Schordan
Malte Isberner
Theo Ruys
Bernhard Steffen

http://www.rers-challenge.org