Notes on “Analysis methods for Executable models”

Nondini Das, Julien Deantoni,

Suchita Ganesan, Jeff Gray,

Laith “Leo” Juwaidah, Domenik Pavletic,

Ernesto Posse, Amal Khalil, Marc Pantel,

Martin Schindler, Eugene Syriani

State of the art

- Translation to external formal language
  - Warning about resulting semantics
- Testing and debugging models
- Contract based methods
  - OCL Pre/post and invariant
- Symbolic execution
- Execution traces (real or simulated)
- Exhaustive simulation and model checking
- Execution semantics refinement

Open challenges

- Handling Abstraction Gap
  - Execution semantics refinement
  - Feedback from lower level checks (for instance by using real execution traces from implementation)
  - Mixing abstraction levels
- Explicit Handling of Semantics variation points
- Providing one golden/reference execution semantics
  - Guarantees/properties of executable models to check of code generation verifying optimizations, etc.
- Handling of Non determinism & concurrency
- Debugging through different execution paths / restoring state
- Give points on how to modify models/semantics to reject observed and undesired behaviors

Open challenges

- Handling Analysis scalability
- *Adoption in industry*
- Allowing Execution of partial models
  - Either under specified models
  - Or model with missing parts
- Handling heterogeneous execution
  - Different formalisms for different views
- FMI like common API / platform ?
- Analysis w.r.t model evolution (incremental analysis, property preservation, …)

Long term future work directions

- common API / virtual platform for model execution (*FMI like*)
- *Conceptual models* or *Ontology* to describe/classify solutions (in a meta-language independent manner)
- Modeling level optimizations

Exe 2016 wish list

- SoA survey on “Translation to external formal language”
- Open experimental benchmark to compare different approaches for model execution/analysis
- Industrial feedback on model behavior analysis