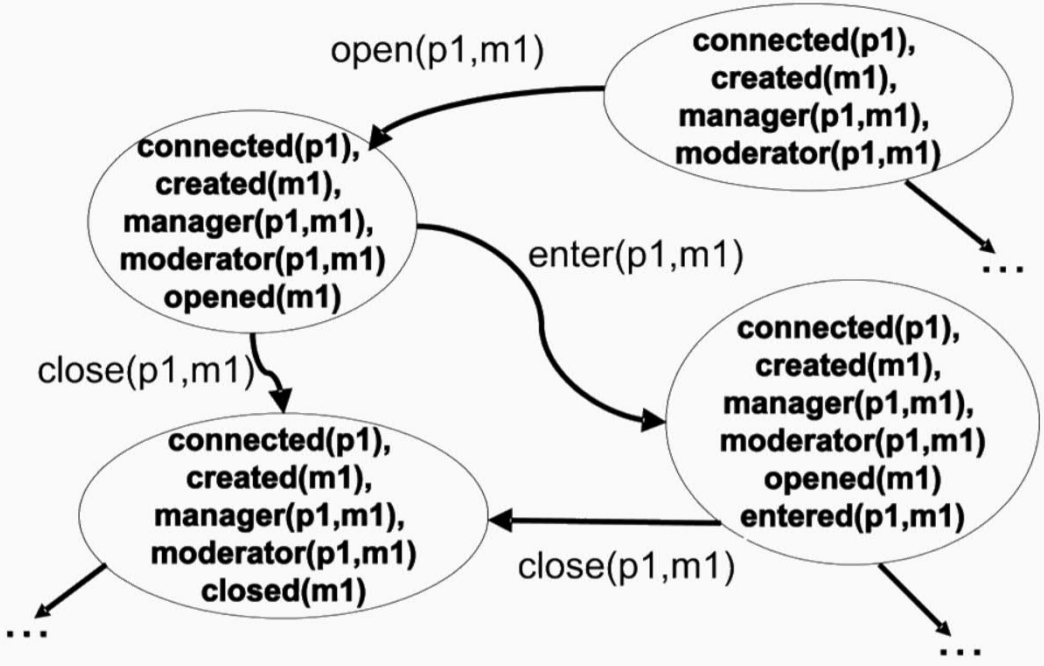
Generation of System Tests through Transition Systems

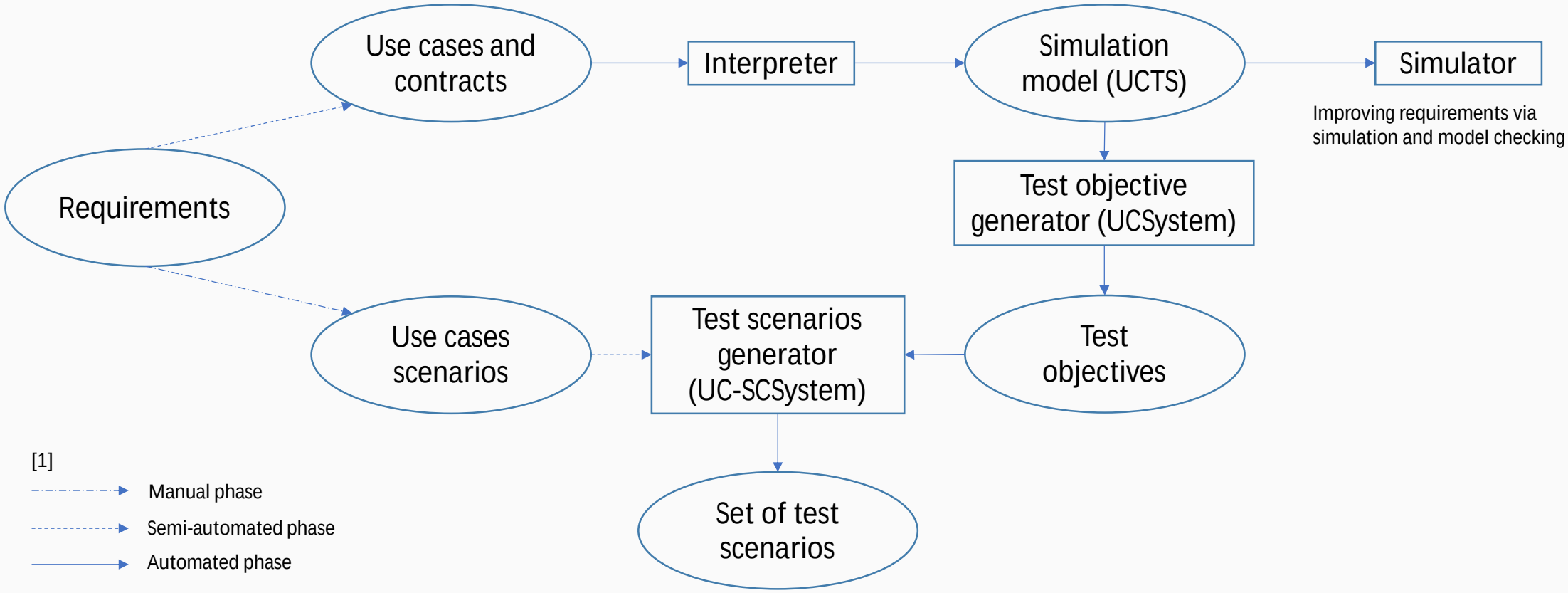
Mid-Term Results – Felix Hausberger

What is a Transition System?



[1]

Automatic Test Generation: A Use Case Driven Apporach



Research Planning

- Central Research Question

  „Which approaches for automatic generation of system tests exist that are using contract enriched use cases or other use case related means of the specification area within a transition system simulation model?“

- Search Terms:
  - System tests
  - Automatic generation
  - Transition system
  - Simulation model
  - Use cases
  - Contracts

Research Planning

- Research Sources: IEEE Xplore[1], ACM[2]

- Relevance Criteria
  - Does the method described in the article generate system tests automatically from use cases or other use case related means of the specification area?
  - Are test objectives generated using some kind of simulation model based on use case contracts (pre- and postconditions) or similar transition system approaches?

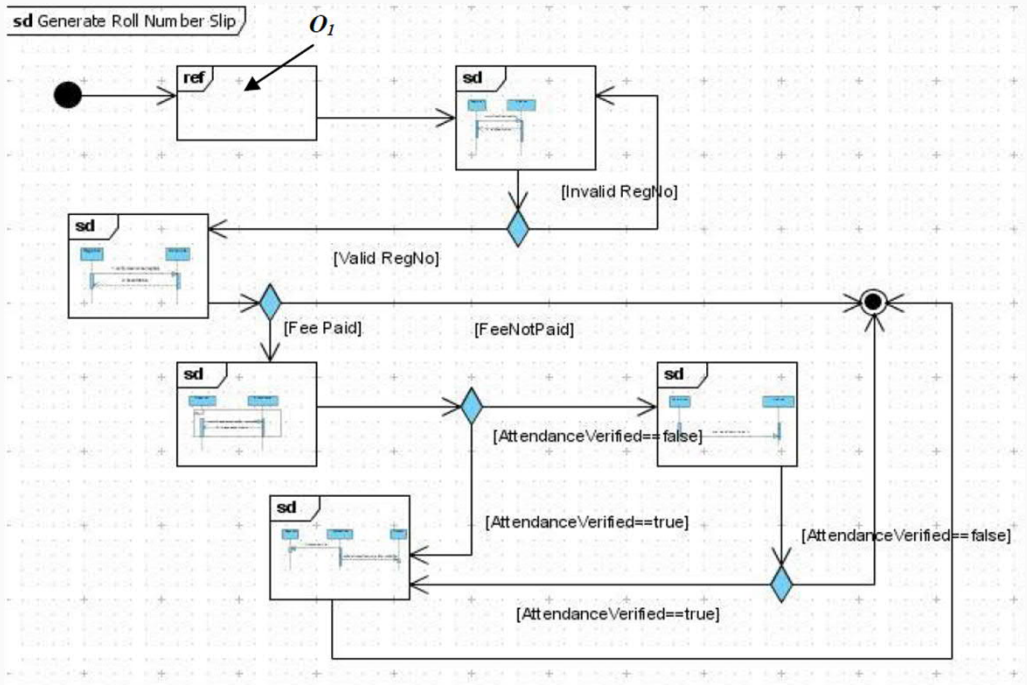
[1] https://ieeexplore.ieee.org/
[2] https://dl.acm.org/

Research Execution and Evaluation

https://github.com/fidsusj/SWE-Seminar

An Automated Approach to System Testing based on Scenarios and Operations Contracts

- Uses scenarios expressed as UML Interaction Overview Diagram (IOD)



[2]

An Automated Approach to System Testing based on Scenarios and Operations Contracts

- Contract Transition System (CTS) is built by enhancing the operations in the IOD with contracts written in Object Constraint Language (OCL)

```
context     RollNumSlip::issueRollNumSlip(regNo)
post:       self.regNo = regNo
            session = true

context     Accounts::verifyClearance(regNo:Integer) : Boolean
pre:        Student.regNumberStatus = true
post:       if self.feePaid = feeDue
                then Student.duesStatus = true
                result = true
            else Student.duesStatus = false
            result = false
            endif

context     Student::updateStudentRecord(regNo:Integer, rns:RollNumSlip)
pre:        self.attendanceVerified = true
            self.regNo = regNo
post:       self.rollNumSlip = rns
```

[2]

An Automated Approach to System Testing based on Scenarios and Operations Contracts

- Test paths are derived by path traversals from the inital node to the final node and by taking into account a specific coverage criterion
  - Transition coverage
  - State coverage
  - Transition pair coverage
- The original paper differs as the transition system is generated on system level and not on a concrete scenario level

Conclusion

- Both papers address the traceability problems between high-level views and concrete test case execution
- Using only means of the specification level (use cases, use case scenarios and UML Interaction Overview Diagrams) one can automatically generate test cases using a transition system
- Contracts are used to infer the correct partial ordering of functionalities that the system should offer
- Coverage criteria define the amount of test cases to be generated

Generation of system tests through transition systems

Thank you for your attention

Questions

Sources

[1] Clémentine Nebut, Franck Fleurey, Yves Le Traon, Jean-Marc Jézéquel, Automatic Test Generation: A Use Case Driven Approach, 2006. [Online]. Available: https://ieeexplore.ieee.org/document/1610607 (visited on 11/25/2020)

[2] Najla Raza, Aamer Nadeem, Muhammad Zohaib Z. Iqbal, An Automated Approach to System testing based on Scenarios and Operations Contracts, 2007. [Online]. Available: https://ieeexplore.ieee.org/document/4385504 (visited on 11/25/2020)