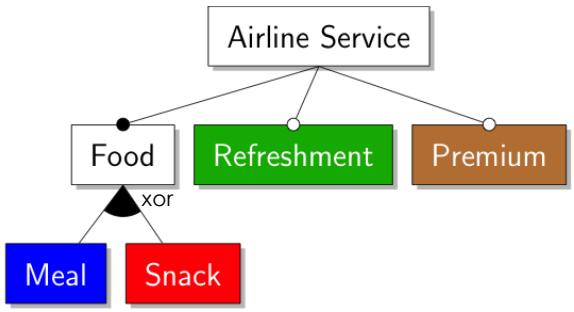
Family-Based Model Checking using Off-the-Shelf Model Checkers

by Aleksandar S. Dimovski, **Ahmad Salim Al-Sibahi**, Claus Brabrand & Andrzej Wasowski

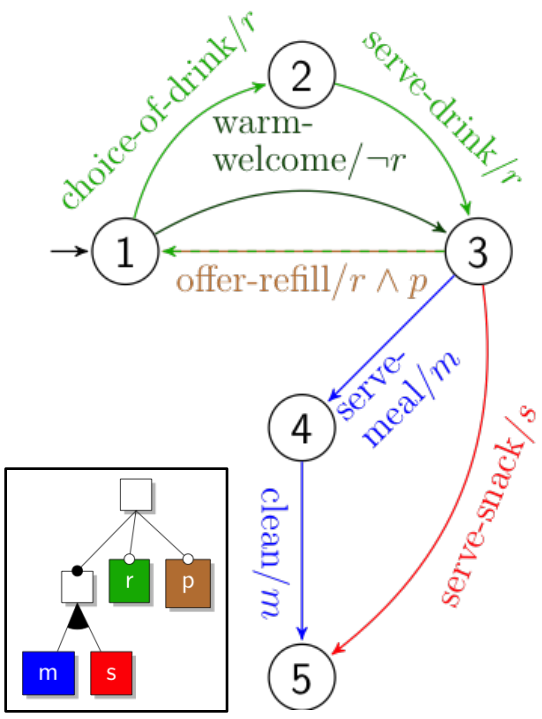
IT University of Copenhagen

Airline

- Service received on-board depends on the airline
- Ensure correctness of instruction manual for flight attendants

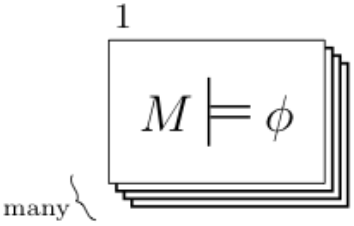


Flight attendant process



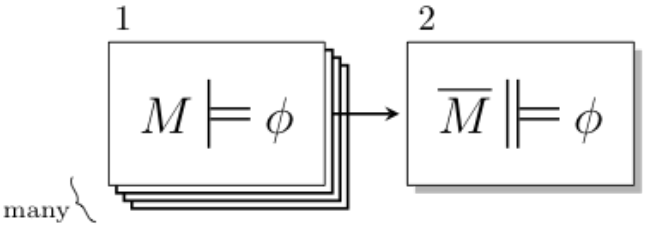
- Flight attendants follow a fixed set of instructions
- Instructions must meet the passengers expectations

State of the art



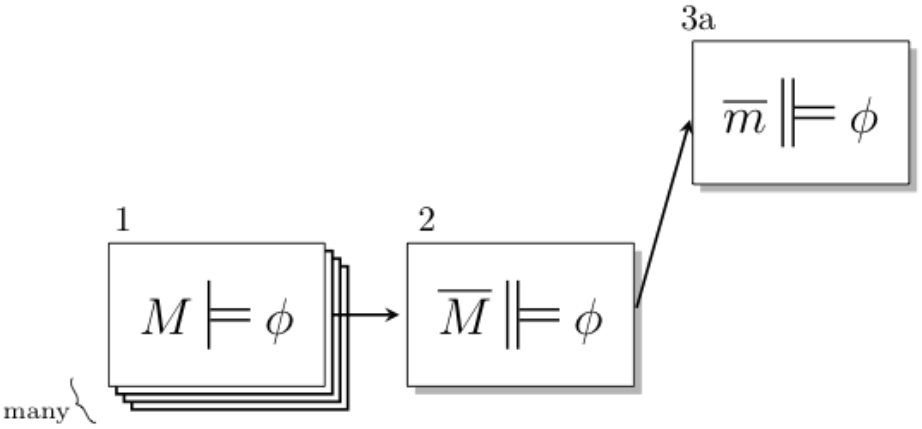
Holzmann, G. J. The SPIN Model Checker - primer and reference manual. Addison-Wesley, 2004.

State of the art

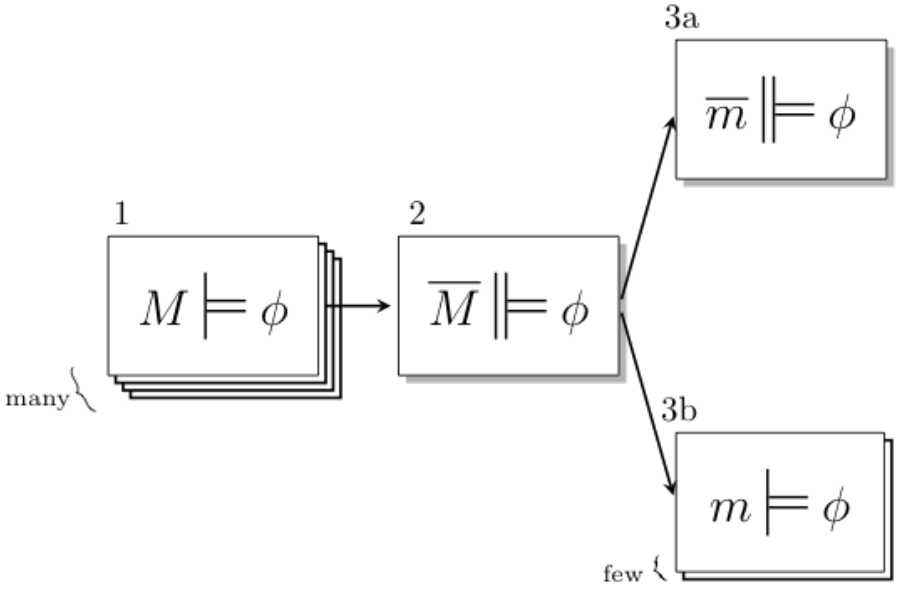


Classen, A., Cordy, M., Heymans, P., Legay, A., and Schobbens, P.
Model checking software product lines with SNIP. STTT 14, 5 (2012), 589–612.

State of the art



State of the art



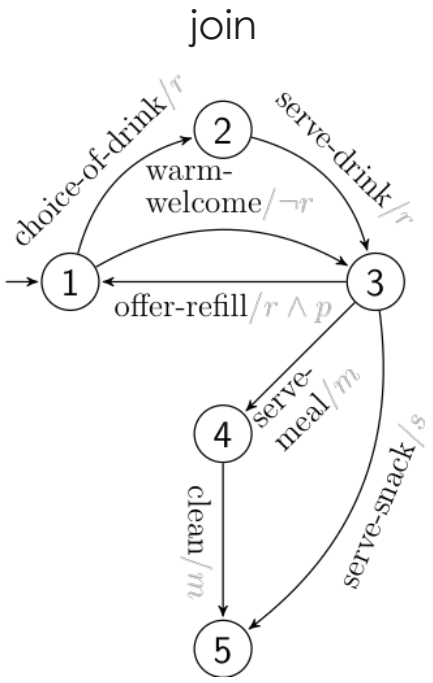
Objective

"*Improving the **scalability** of model checking software product lines by introducing **variability abstractions***

a ::= join | ignore(f) | project(ϕ) | $a_1 \circ a_2$

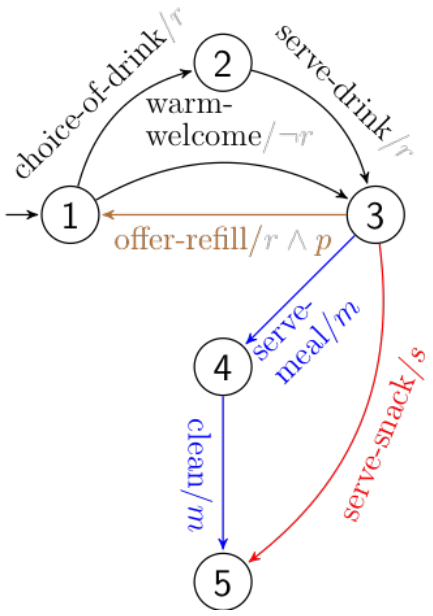
VARIABILITY ABSTRACTIONS

Join



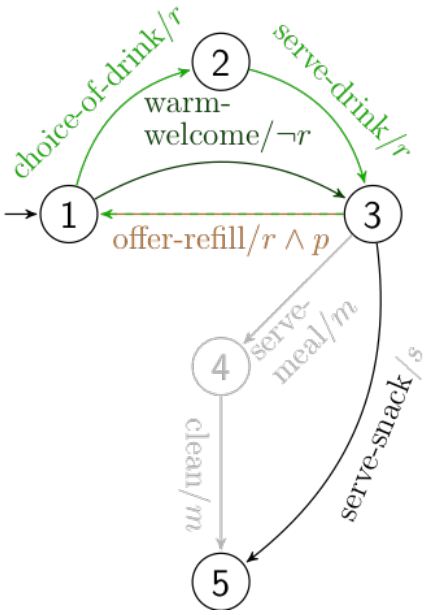
Ignoring Features

ignore(r)

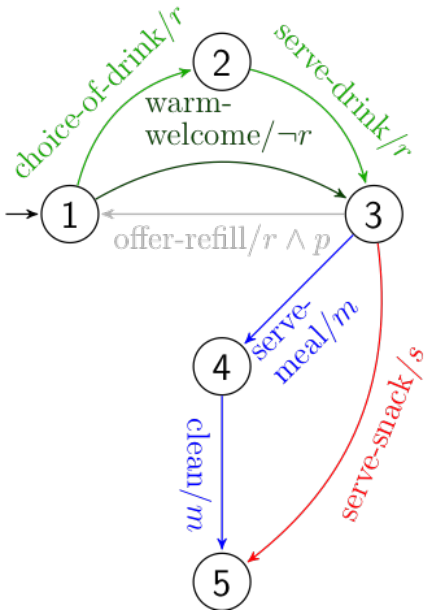


Projection

project(s)

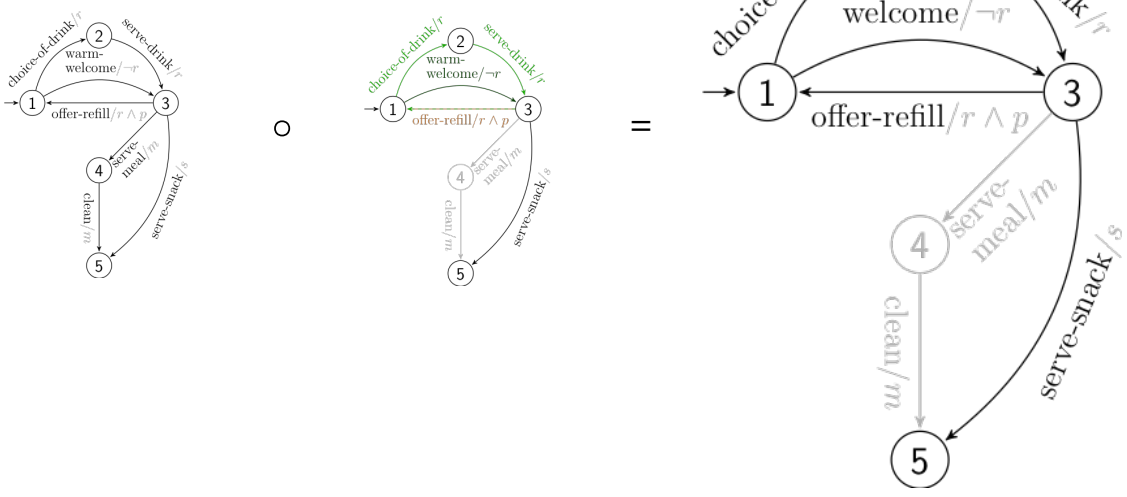


project($\neg$p)



Composition

join ◦ project(s)

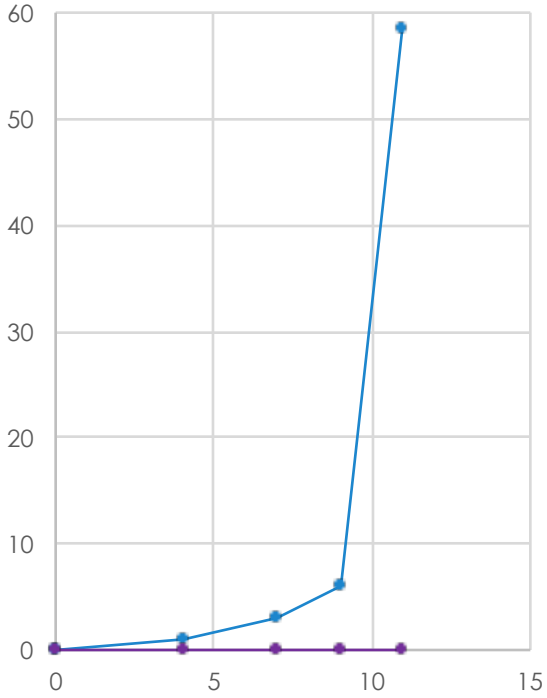


Property Preservation

- Abstractions form Galois connections
- Abstractions soundness
- Abstraction decomposition

Evaluation

- Modified version of MinePump*
- Infeasible → Feasible model checking
- Considerable improvement in both time and space use

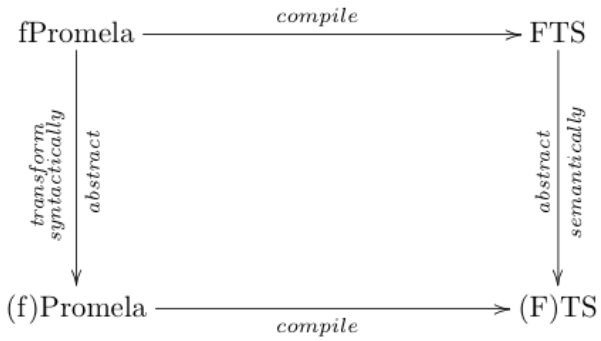


Mahony, B. P., and Hayes, I. J.
A case-study in timed refinement: A mine pump. IEEE Trans. Software Eng. 18, 9 (1992), 817–826.

Future directions

1. Automation of abstraction application
   - Property-sensitive Heuristics
   - Counter-example guided abstraction refinement (CEGAR)
2. Supporting more complex features beyond Boolean values
   - Enumerations
   - Integers
   - Strings
3. Integration with software model checking
   - Java Pathfinder for Java
   - CPAChecker for C
   - Etc.

Tool



- Syntactically transforms fPromela programs to abstracted fPromela (or Promela) programs
- Available for installation

Summary

- A calculus of variability abstractions that works symbiotically with both off-the-shelf and lifted model checkers
- Implemented in a tool using syntactic source-to-source transformations
- Evaluation shows considerably improved feasibility, and that many properties did not require full precision to be proven correctly

Information

- Tool available from: https://github.com/ahmadsalim/p3-tool

- Main paper:

A. S. Dimovski, A. S. Al-Sibahi, C. Brabrand, and A. Wąsowski.

Family-based model checking without a family-based model checker. In 22nd International SPIN Symposium on Model Checking of Software, Stellenbosch, South Africa, August 24 - 26, 2015, 2015. To Appear.