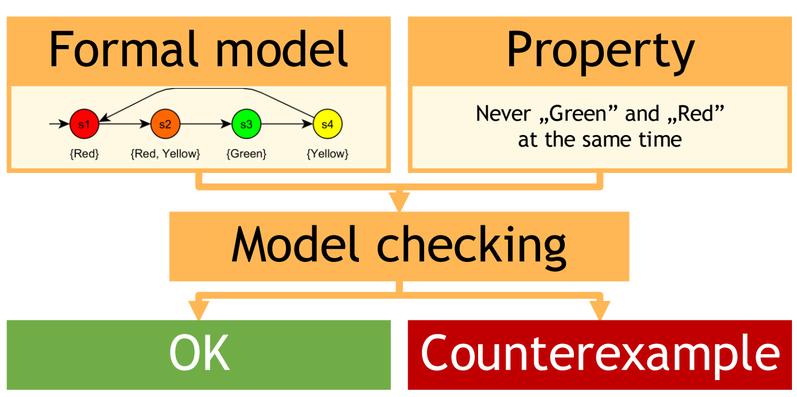
A SAT/SMT-based CEGAR Framework

Á. Hajdu, T. Tóth, A. Vörös, Z. Micskei, I. Majzik

Motivation

Formal verification



Challenges

- Large set of states and behaviors
- Possible solution is abstraction
  - CEGAR method: start with coarse abstraction and refine until sufficient precision is reached
  - Various abstract domains and refinement strategies exist

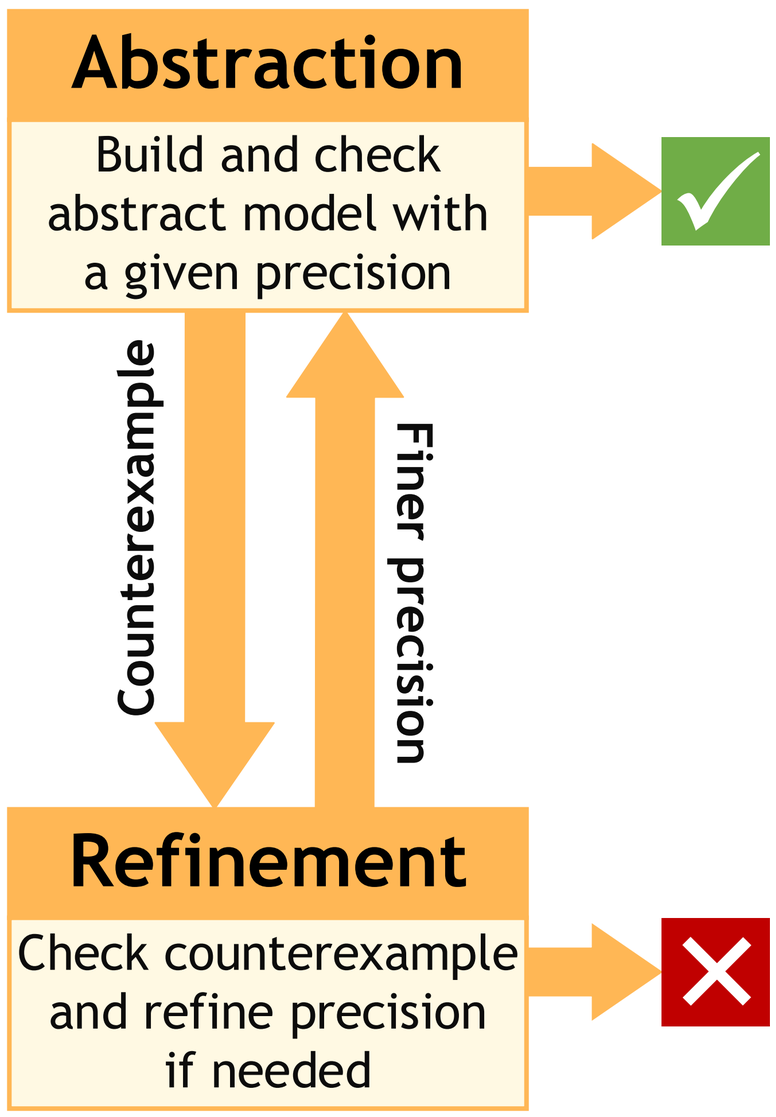
Goal

- Finding efficient combinations of abstractions and refinements for analyzing various models
- Evaluating heuristics for selecting the most efficient combination

→ Configurable framework required

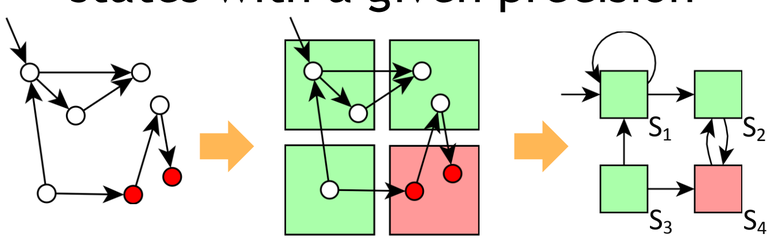
Counterexample-Guided Abstraction Refinement (CEGAR)

Overview



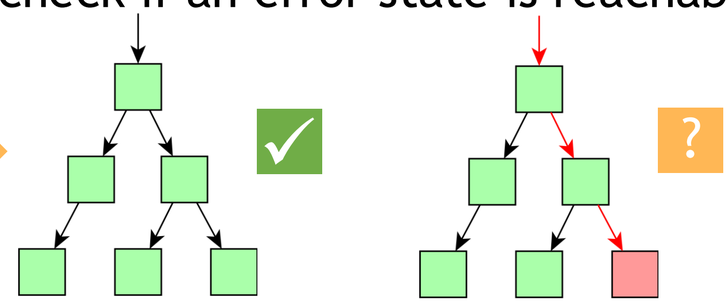
Abstraction

Map concrete states to abstract states with a given precision



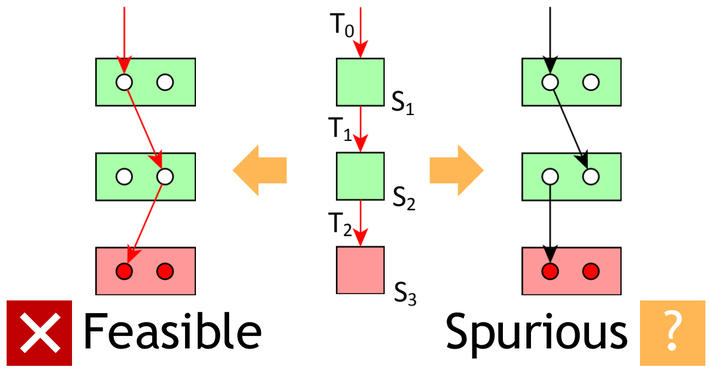
- Concrete state: variable assignment
- Abstract state: set of concrete states described by a FOL formula

Explore abstract state space and check if an error state is reachable



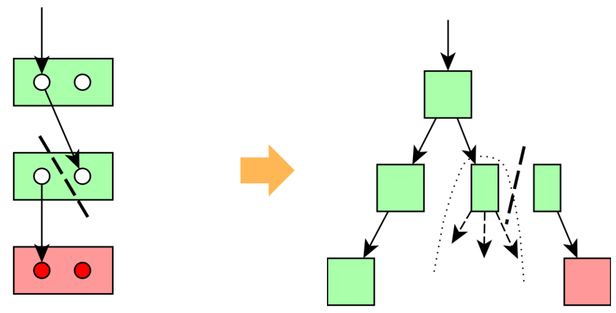
Refinement

Check if counterexample is feasible in the concrete model



Use SMT solver: counterexample feasible iff $(T_0 \wedge S_1 \wedge T_1 \wedge \ldots \wedge T_{n-1} \wedge S_n)$ is SAT

Refine precision and continue exploration



Use SMT solver: extract refutation from UNSAT formula (interpolant, unsat core, etc.), and refine precision based on the refutation

A Generic CEGAR Framework

Formalisms

- Transition Systems
- Control Flow Automata
- Timed Automata
- Statechart

Analysis modules and layers

- Abstraction
  - Domain: Predicate, Value, Zone, Location
  - Exploration strategy
  - Formalism interpreter
- Refinement
  - Concretizer: Craig itp., Sequence itp., Unsat core
  - Refinement strategy

Core

- FOL expressions

Solver interface

SMT solvers

Advantages

- Configurable
  - Formalisms
  - Abstract domains
  - Refinement strategies
- Novel combinations of abstractions and refinements
- Allows using heuristics for selecting the most efficient configuration based on input

Reference

Ákos Hajdu, Tamás Tóth, András Vörös, István Majzik. *A Configurable CEGAR Framework with Interpolation-Based Refinements*. In Formal Techniques for Distributed Objects, Components and Systems, volume 9688 of LNCS, pages 158–174. Springer, 2016.

Budapest University of Technology and Economics
Fault Tolerant Systems Research Group
http://inf.mit.bme.hu/en

MŰEGYETEM 1782