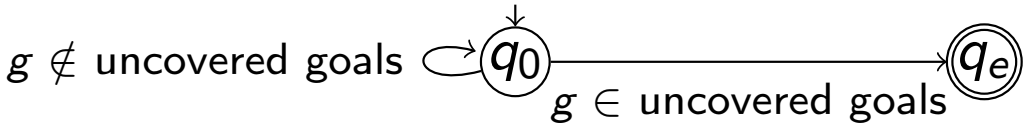
CoVeriTest
Test-Comp'24

Marie-Christine Jakobs

LMU Munich

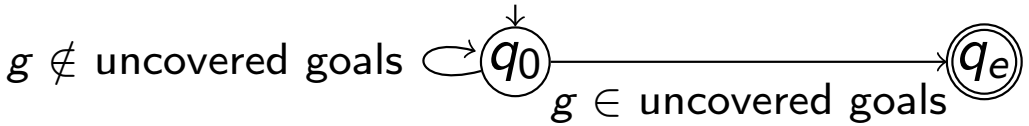
Cooperative **Veri**fier-Based **Test**ing in a Nutshell

- Verifier-based testing
  - Reaching a test goals = specification violation



Cooperative **Veri**fier-Based **Test**ing in a Nutshell

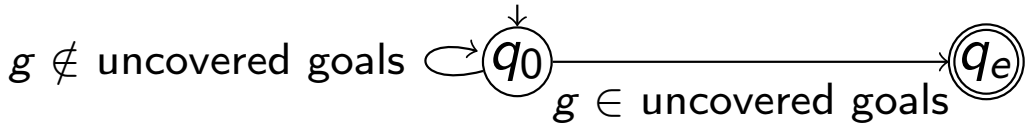
- Verifier-based testing
  - Reaching a test goals = specification violation



  - Use verifiers to check specification

Cooperative **Veri**fier-Based **Test**ing in a Nutshell

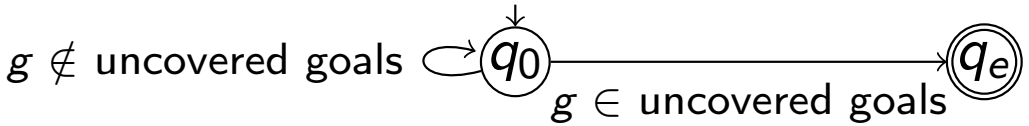
- Verifier-based testing
  - Reaching a test goals = specification violation



  - Use verifiers to check specification
  - Generate test cases from counterexamples

Cooperative **Veri**fier-Based **Test**ing in a Nutshell

- Verifier-based testing
  - Reaching a test goals = specification violation

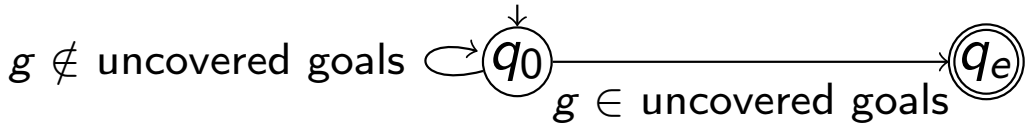


  - Use verifiers to check specification
  - Generate test cases from counterexamples

- Cooperation
  - Interleave different analyses

Cooperative Verifier-Based Testing in a Nutshell

- Verifier-based testing
  - Reaching a test goals = specification violation



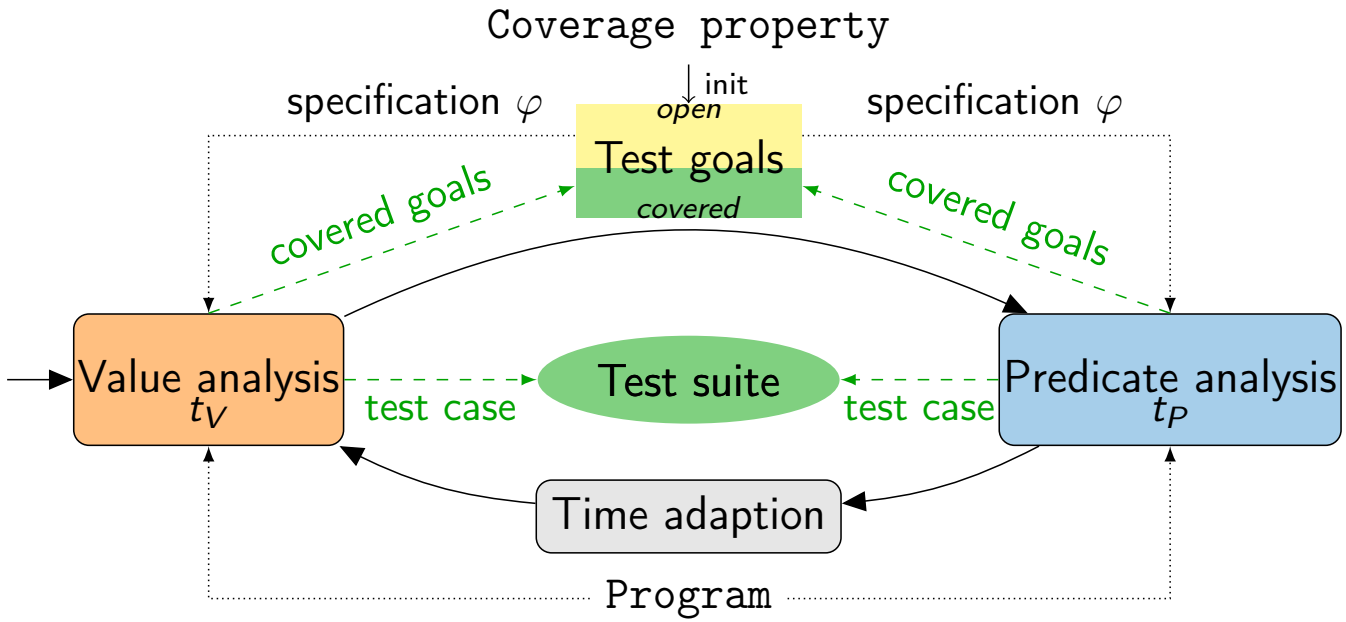
  - Use verifiers to check specification
  - Generate test cases from counterexamples

- Cooperation
  - Interleave different analyses
  - Analyses may reuse previous analysis results, i.e.,
    - Use ARG returned by last run of this analysis
    - Continue exploration

CoVeriTest Overview

Integrated in CPAchecker

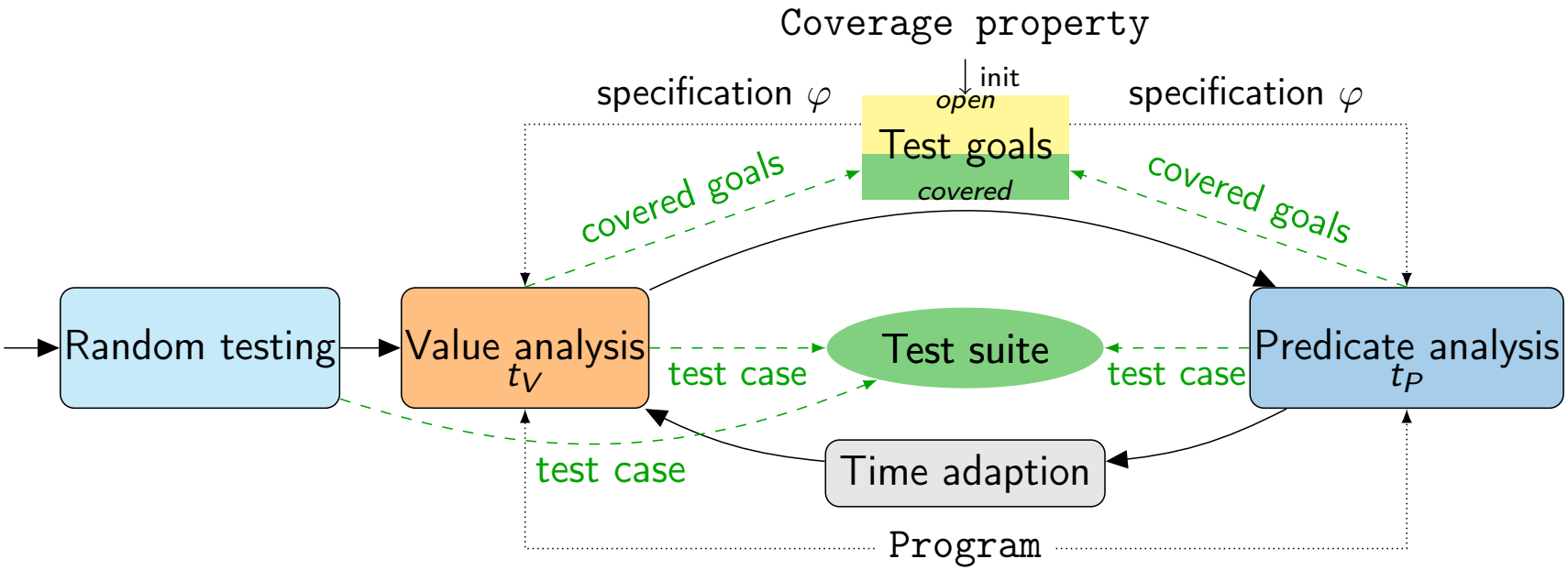
`https://cpachecker.sosy-lab.org`



CoVeriTest Overview

Integrated in CPAchecker

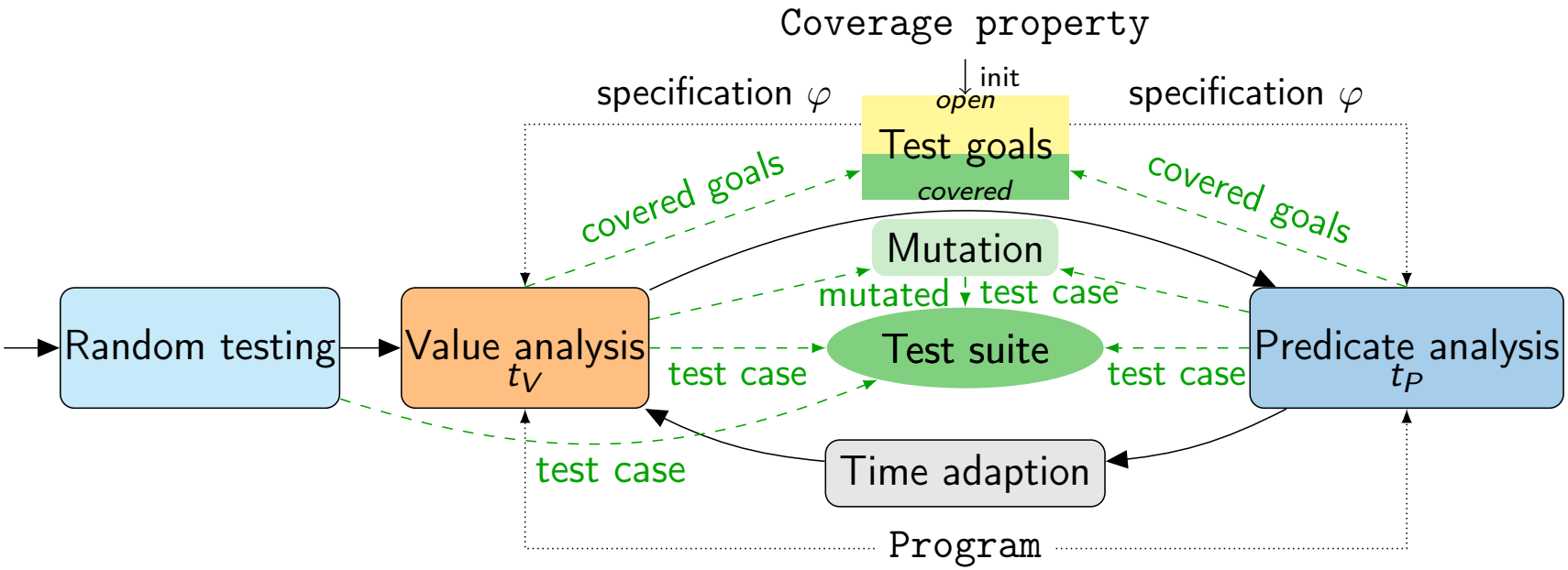
`https://cpachecker.sosy-lab.org`



CoVeriTest Overview

Integrated in CPAchecker

`https://cpachecker.sosy-lab.org`



Time Limit Adaption

- Initially, $t_V = 20\,s$, $t_P = 80\,s$
- Adaption rewards *past* behavior
  - Only if new goals covered
  - Based on relative progress p_i

$$\text{limit}_i^{\text{new}} = 10\,s + \frac{\frac{p_i}{\text{limit}_i}}{\frac{p_V}{\text{limit}_V} + \frac{p_P}{\text{limit}_P}} * 80\,s$$

Random Testing and Mutation

Random Testing

- 10 test cases
- Each random length in [0,20]
- Random integer values in [0,20]

Random Testing and Mutation

Random Testing

- 10 test cases
- Each random length in [0,20]
- Random integer values in [0,20]

Mutation of Test Cases

- 5 mutated test cases per generated test case
- Randomly mutate input values
  - Minimum integer value (2%) or maximum integer value (2%)
  - Zero value (15%)
  - Negated value (15%)
  - Random integer value (64%)

Brief Assessment of Results

- Place 11 in cover-error
- Major weakness cover-error category
  - Runs out of resources in roughly 50% of the cases
  - Test-cases do not always reveal violation

Brief Assessment of Results

- Place 11 in cover-error
- Major weakness cover-error category
  - Runs out of resources in roughly 50% of the cases
  - Test-cases do not always reveal violation

- Place 6 in cover-branches
- Better than winner FuSeBMC in following cover-branches subcategories
  - ReachSafety-Hardware
  - ReachSafety-XCSP