Model-based, mutation-driven test case generation via heuristic-guided branching search

Andreas Fellner (AIT/TUW)
Willibald Krenn (AIT), Rupert Schlick (AIT), Thorsten Tarrach (AIT), Georg Weissenbacher (TUW)

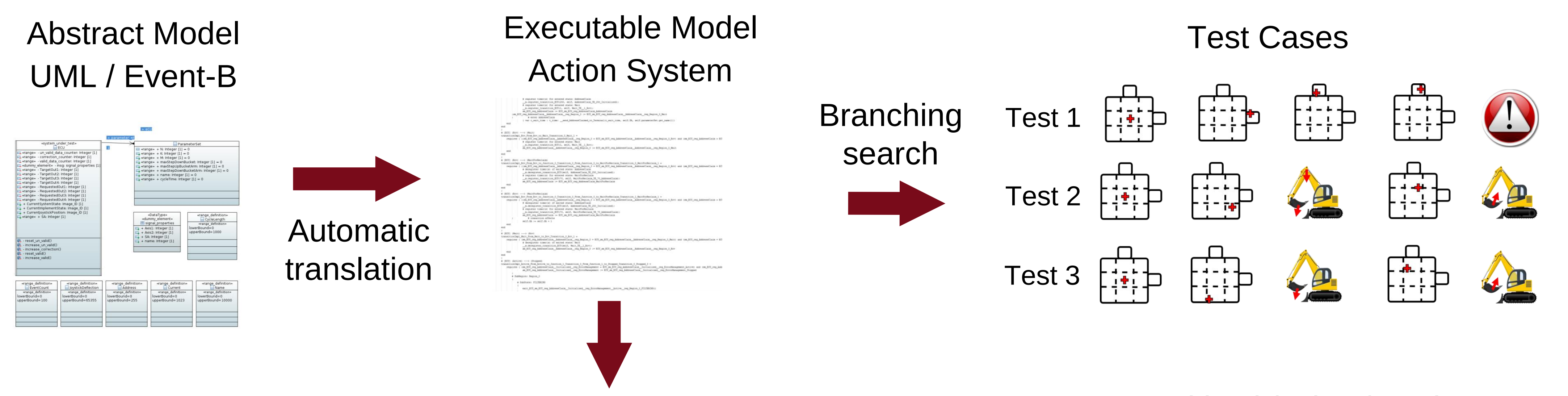
Abstract

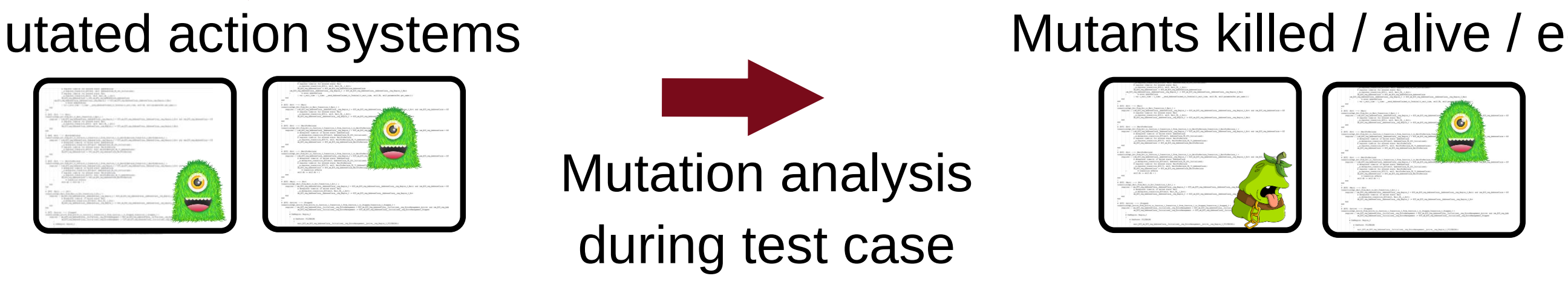
This work introduces a heuristic-guided branching search algorithm for model-based, mutation-driven test case generation. The algorithm is designed towards the efficient and computationally tractable exploration of discrete, non-deterministic models with huge state spaces. Asynchronous parallel processing is a key feature of the algorithm. The algorithm is inspired by the successful path planning algorithm Rapidly exploring Random Trees (RRT). We adapt RRT towards test case generation by introducing parametrized heuristics for start and successor state selection, as well as a mechanism to construct test cases from the data produced during search. With our new algorithm, we are now able to produce test cases for models consisting of over 2300 concurrent objects.

Contributions

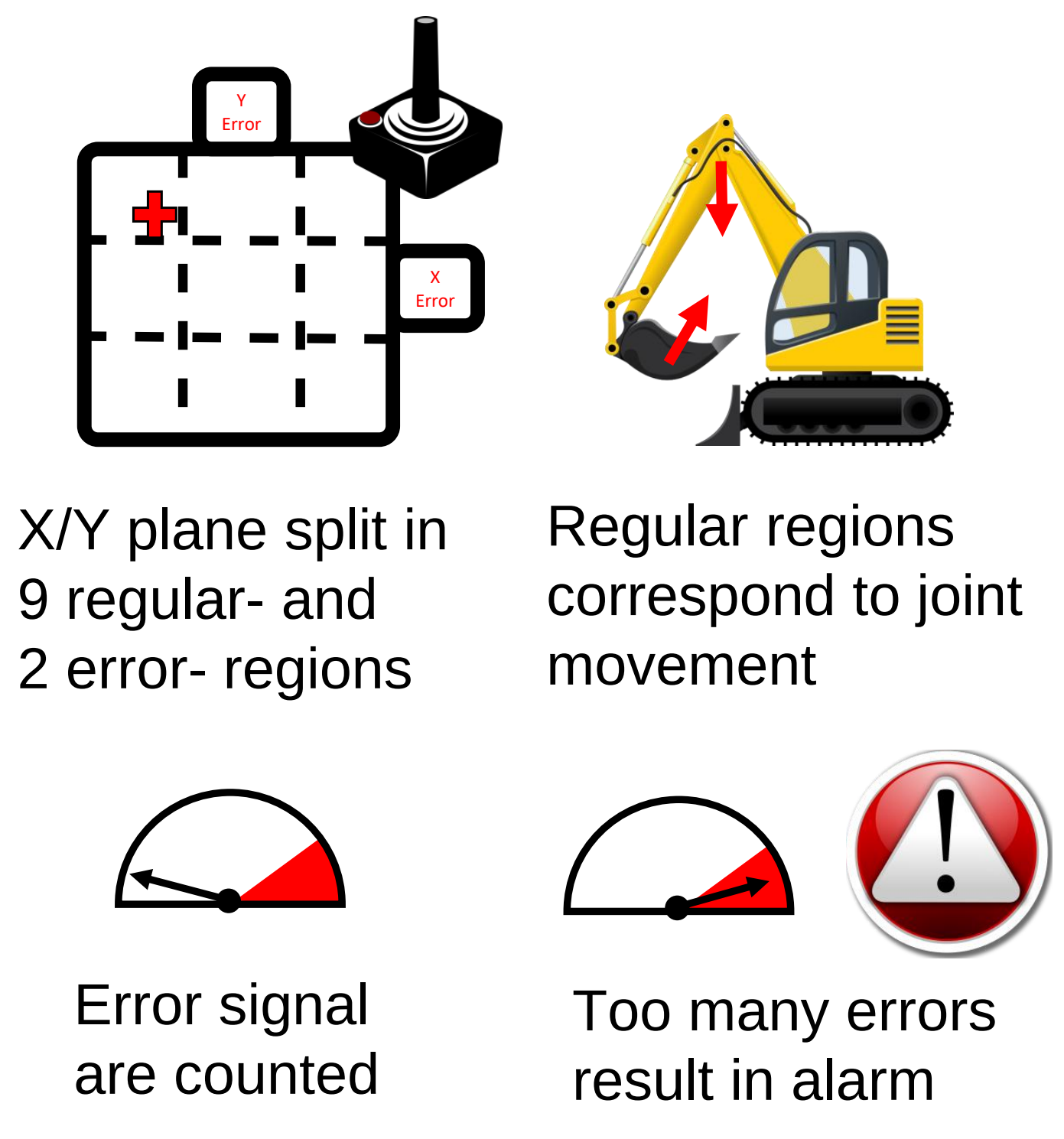
- Branching Search For Test Case Generation
  - Fully leverage parallelism
  - Flexibility through set of heuristics
  - Shorter and more effective tests
- Extensive evaluation on large models

Model Based Testing with MoMuT




Mutated action systems
Mutation analysis during test case generation
Mutants killed / alive / equivalent?

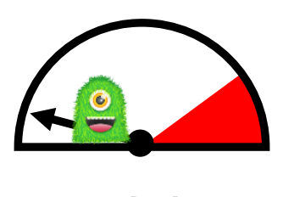
Bucket Loader Example


X/Y plane split in 9 regular- and 2 error- regions
Regular regions correspond to joint movement
Error signal are counted
Too many errors result in alarm

Mutation

- Small change of the system
- Test case kills a mutant if
  - Original system has output X
  - Mutated system has output X != Y
- Mutation coverage criterion
  - Kill x% of produced mutants
- Equivalent mutants
  - Can not be killed by any test

Example Mutations



Disfunctional error counter

Original	Mutant

`if error_signal`
`c := c + 1`
	`if error_signal`
	`c := c`

Disfunctional bucket movement

Original	Mutant

`if x_signal > 100`
`bucket_up()`
	`if false`
	`bucket_up()`

MoMuT

- Automated and Model Based Test Case Generation Tool
- Developed at AIT and TU Graz
- www.momut.org

Supported modelling formalisms

- UML
- Event-B
- Object Oriented Action Systems (OOAS)

Why Model Based Testing?

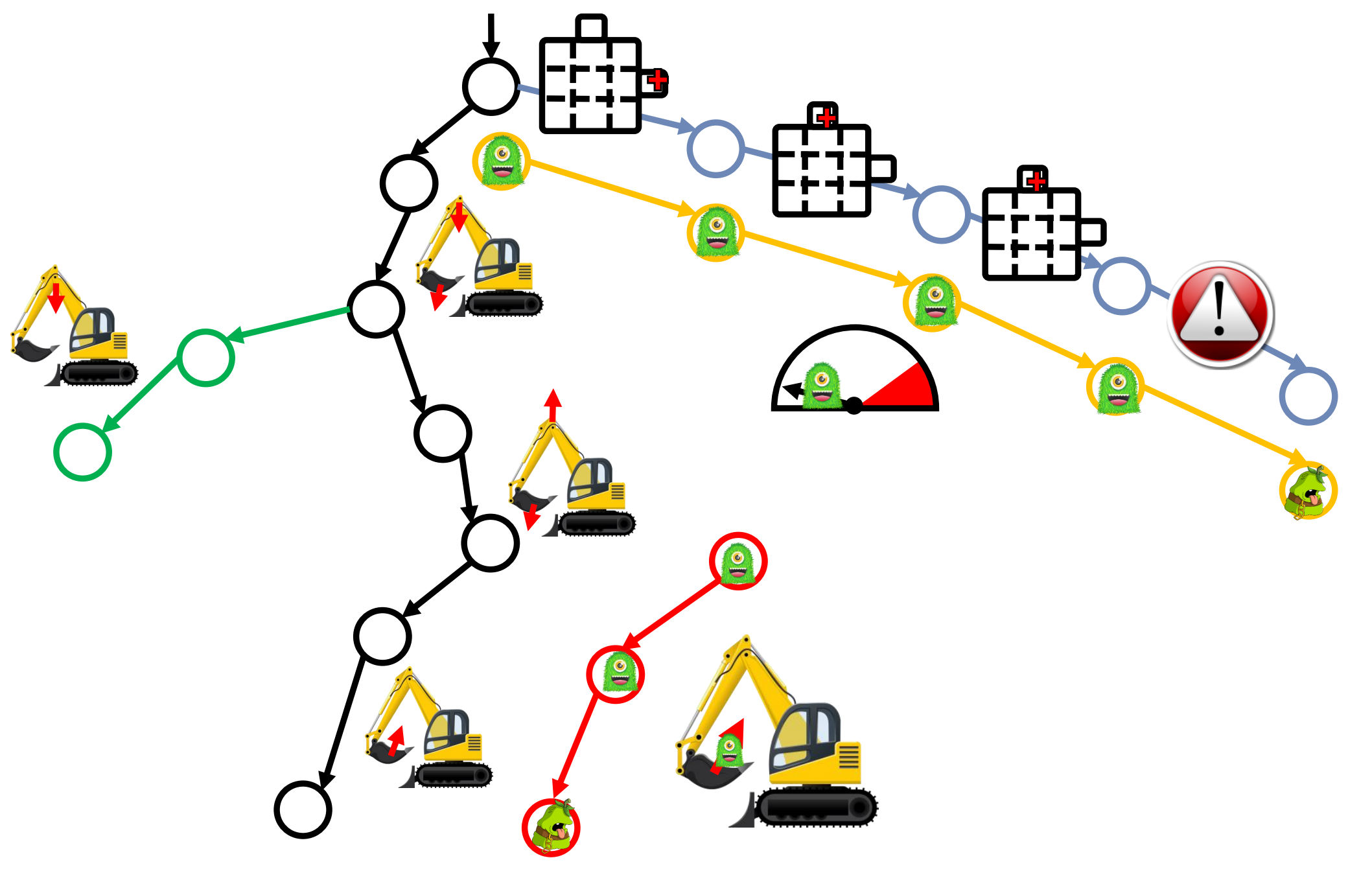
- Model reflects requirements
  - Verify high level correctness instead of properties of code
  - Split roles of test- and the system- designer
- Domain independence
- Essential during model-driven development

Why Mutation Coverage

- Tests directly relate to implementation faults
- Cover much high level behaviour per test case
- Prune irrelevant test steps
- Fine tunable due to choice of mutants

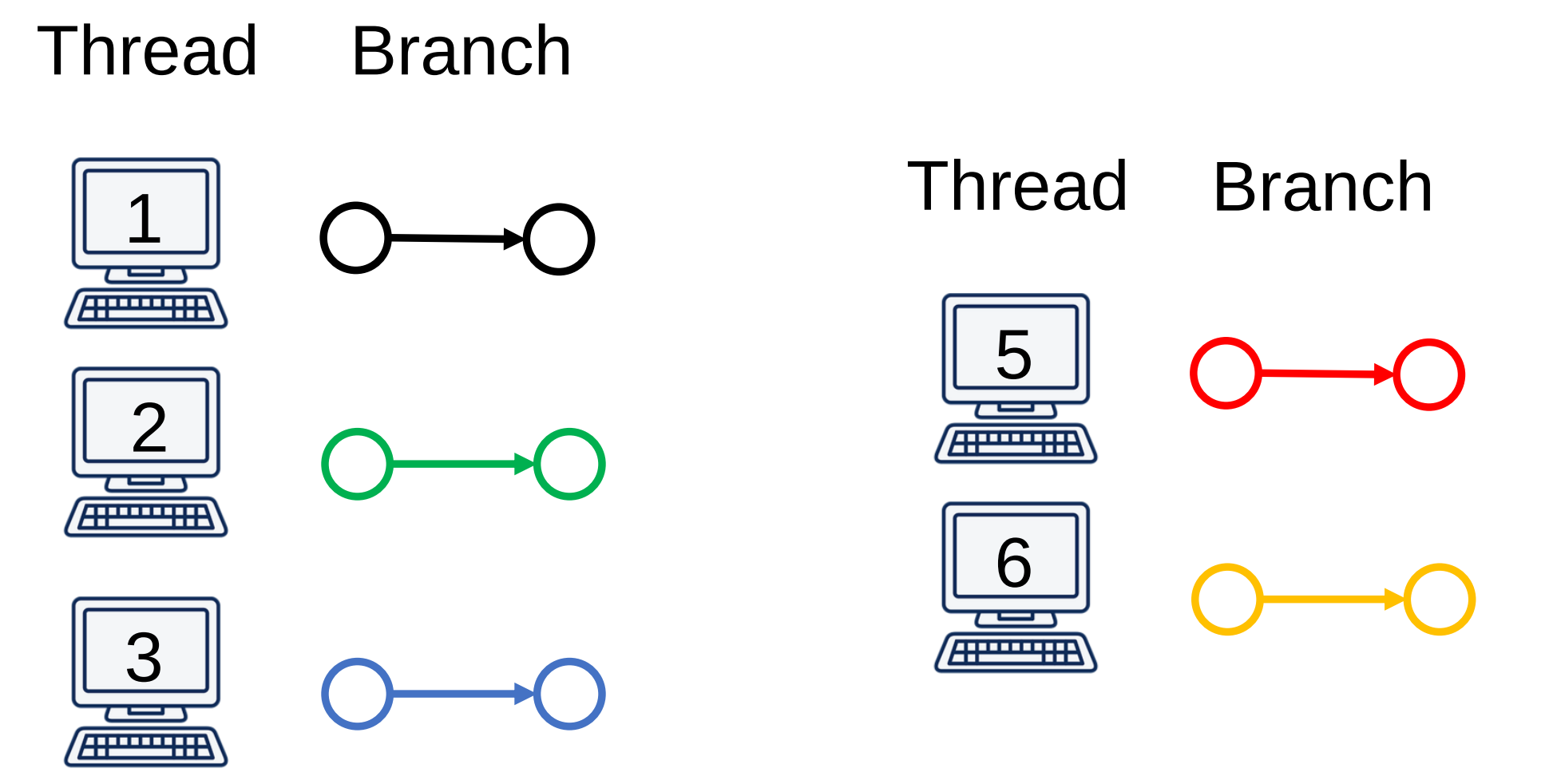
Branching Search for Test Case Generation

Algorithm Sketch & Example



Explore model in parallel

Analyse mutants in parallel



Different branches for different test purposes

Execute mutated model only where necessary

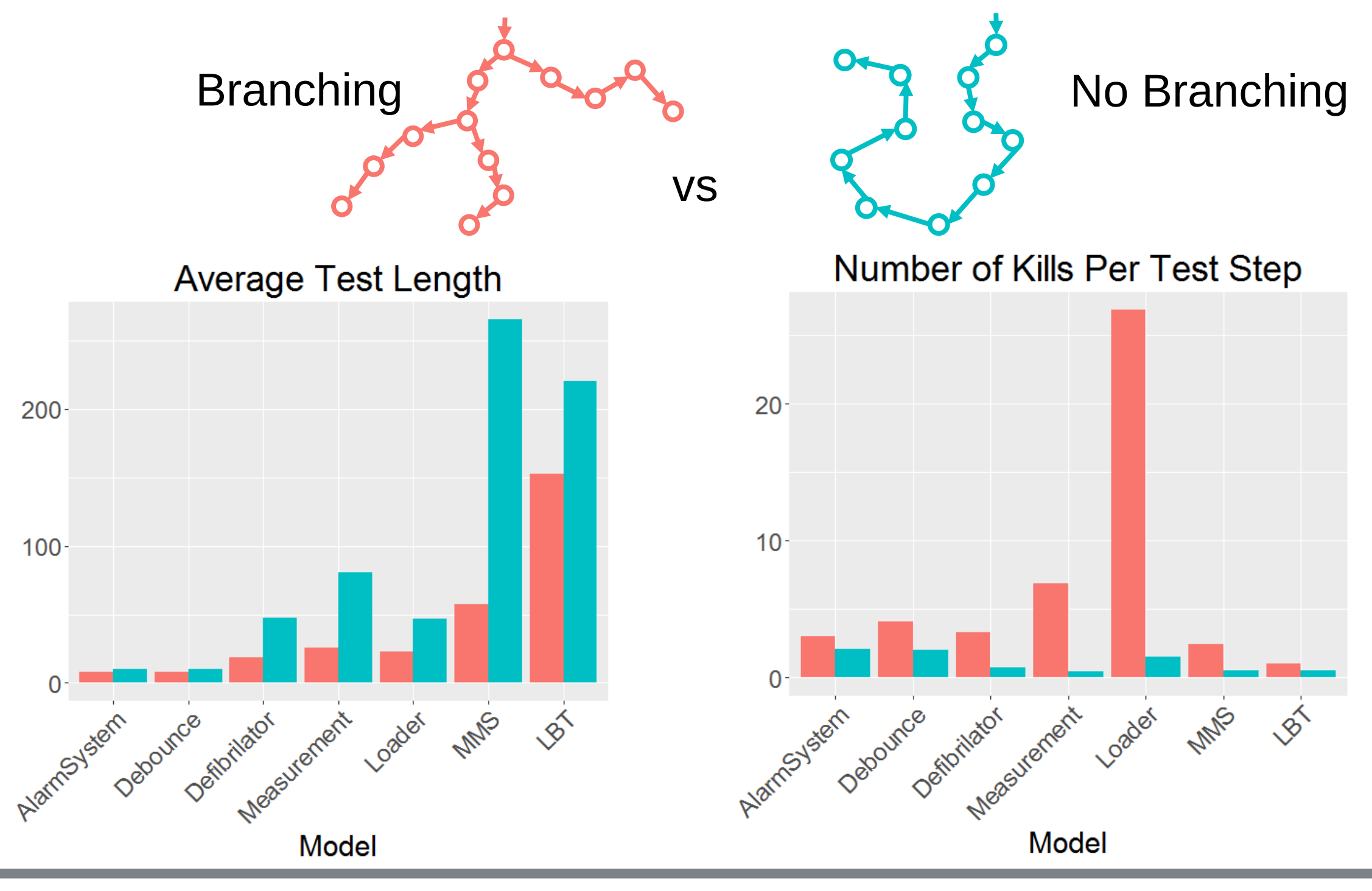
Where to Start New Branches?

- Initial state
- Random state
- Rare state
- Distance based
- Round robin

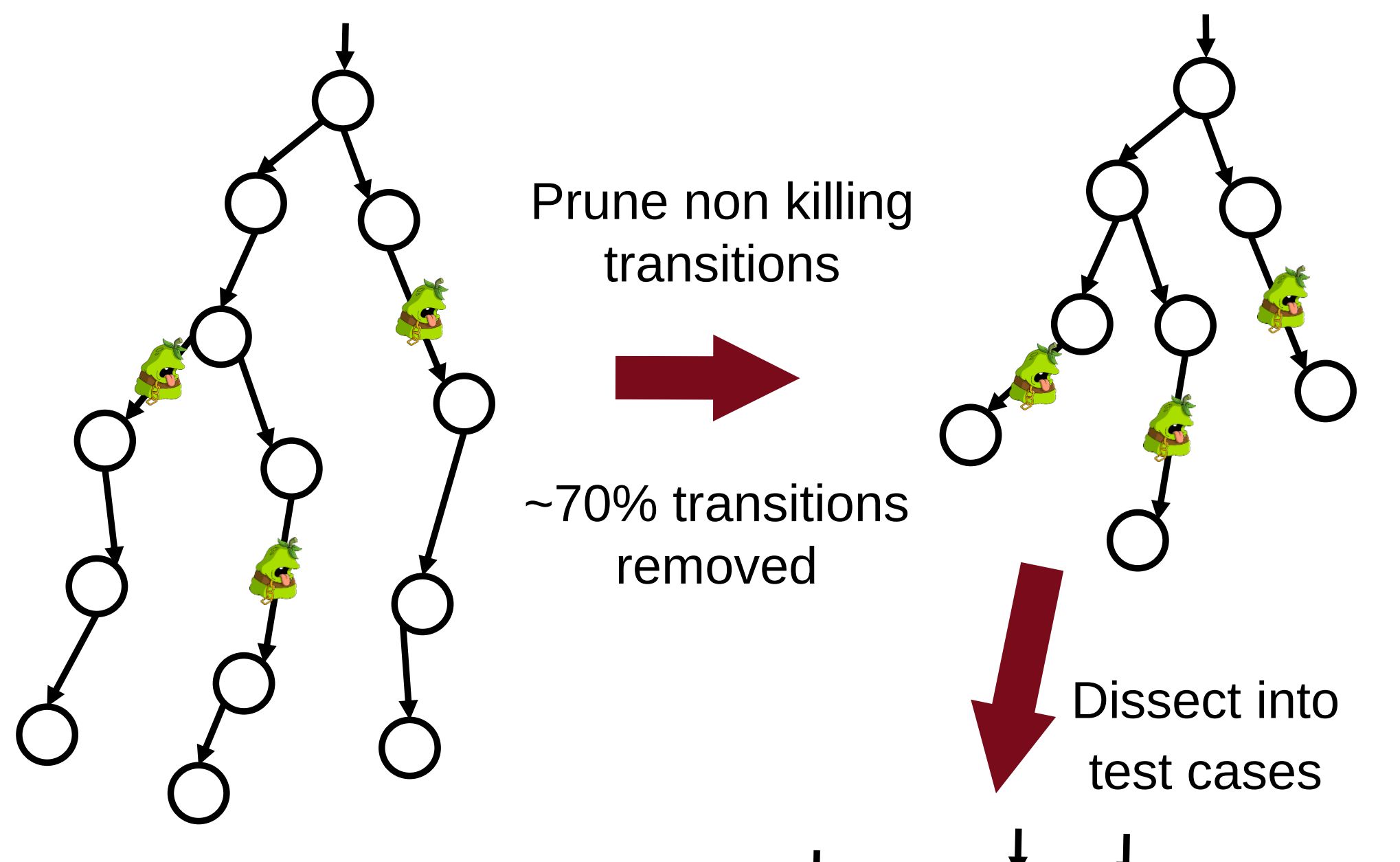
Heuristic / Model	Dist	Rand	Rare	Init	RoRo
AlarmSystem	🟨	🟨	🟩	🟨	🟨
Debounce	🟩	🟨	🟨	🟨	🟨
Defibrilator	🟩	🟥	🟨	🟥	🟨
Measurement	🟨	🟩	🟨	🟥	🟥
LoaderBucket	🟨	🟨	🟥	🟩	🟨
MMS	🟨	🟩	🟥	🟨	🟥
LBT	🟨	🟩	🟨	🟨	🟨

🟩 ... most mutants found
🟥 ... significantly less mutants found
🟨 ... others

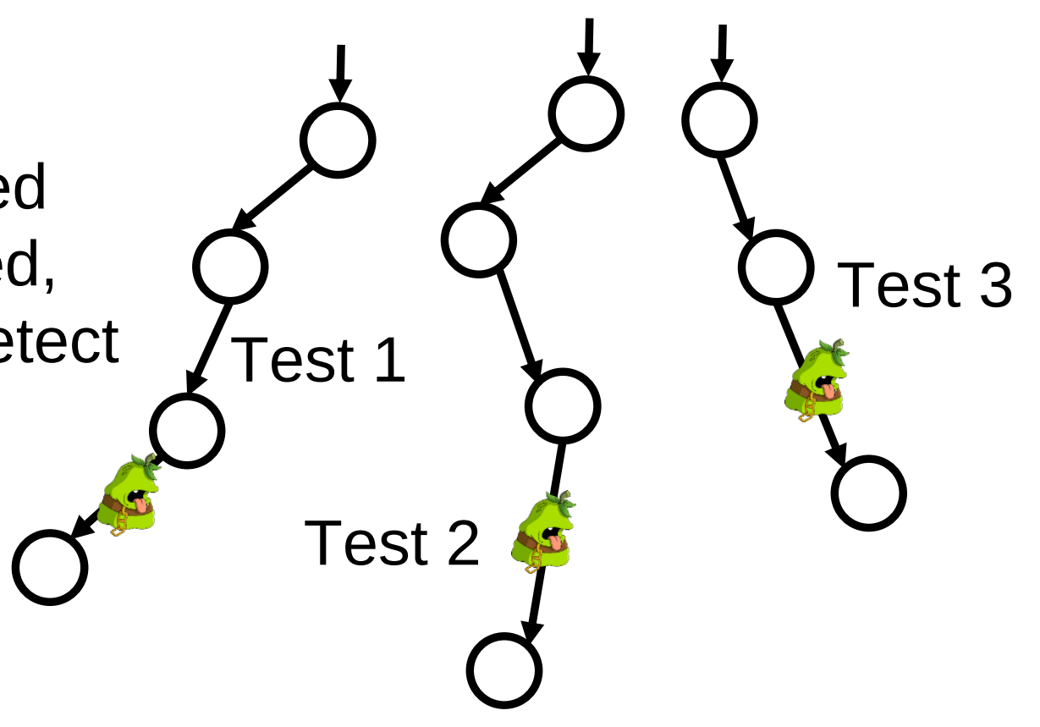
Branching Search Test Case Quality



Branching Search and Test Case Generation



Each test is associated with the mutant it killed, i.e. the faults it can detect



Future Work

- Semi-symbolic exploration techniques
  - Concolic execution
  - Distance based on mutant constraints
- Static analysis to improve heuristics
- Translate OOAS to Petri Nets

Publication

[1] A. Fellner, W. Krenn, R. Schlick, T. Tarrach, and G. Weissenbacher, “Model-based, mutation-driven test case generation via heuristic-guided branching search” in MEMOCODE, 2017

Related Publications

[2] B. Aichernig, H. Brandl, E. Jöbstl, W. Krenn, R. Schlick, and S. Tiran, “MoMuT::UML Model-Based Mutation Testing for UML” in *Software Testing, Verification and Validation (ICST), 2015*

[3] B. Aichernig, J. Auer, E. Jöbstl, R. Korošec, W. Krenn, R. Schlick, and B. Schmidt, “Model-Based Mutation Testing of an Industrial Measurement Device” in *Tests and Proofs (TAP), 2014*

Contact

ANDREAS FELLNER
andreas.fellner@ait.ac.at
Dependable Systems Engineering
Center for Digital Safety & Security
AIT Austrian Institute of Technology GmbH
Donau-City-Straße 1
1220 Vienna
www.ait.ac.at

www.momut.org

This work has been conducted within the ENABLE-S3 project that has received funding from the ECSEL joint undertaking under grant agreement no 692455. ENABLE-S3 is funded by BMVIT under the program "ICT of the Future" via FFG project number 853308. Furthermore, the work is supported by FFG project number 845582 (TRUCONF) and FWF project W1255-N23 (LogiCS Doktoratskolleg).