

# Co-Evolution of Model-Based Tests for Industrial Automotive Software

Eric J. Rapos  
School of Computing  
Queen's University  
Kingston, Ontario, Canada  
Email: eric@cs.queensu.ca

## I. INTRODUCTION

Model-based testing (MBT) is a rapidly expanding field involving the use of software modeling to aid in the creation, automatic generation, execution, and maintenance of software tests. However, MBT is an area with room for improvement; resulting test suites may be of unmanageable size, and maintaining test models over time can be a complex task.

The concept of co-evolution refers to two (or more) objects evolving alongside each other, such that there is a relationship between the two that must be maintained. In the field of co-evolution of model-based tests, this refers to the test models evolving alongside the source models, such that the test models remain correct for testing the source models. Previous work centered largely on the iterative development aspect of MBT [3], [4], however further attention is needed on the prolonged maintenance of model-based tests after initial release.

**Thesis Statement:** Model-based test efficiency can be improved by co-evolving test models alongside system models. This can be done through studying software model evolution patterns and their effects on test models in order to apply updates directly to the tests.

## II. RELATED WORK

The work by Zech et al. [5] for the MoVE framework, deals specifically with regression testing and the selection of the test set, however we aim to focus on the co-evolution aspect, and how exactly tests change and evolve alongside the source models. Similarly work by Farooq et al. [2] also focuses on the selection of regression tests for future testing; our work will incorporate this step but also expand to include the adaptation of existing tests. With regards to the testing work done in Simulink [1], this work looks specifically at the testing of systems, while our work aims to explore how these types of tests evolve.

## III. METHODOLOGY

The work will be conducted using Matlab Simulink as our modeling language. This is primarily due to the increasing number of automotive companies (General Motors included) that are using this technology in their development. Additionally Simulink provides several added benefits for automotive development which make it an interesting technology to study; the versatility of the Simulink environment provides real-time simulation capabilities which allow for more reliable and accurate testing early in the development process.

There are three phases to the proposed research:

- **Evolution Study:** Examining a selected set of Simulink models to determine how they evolve over time, and the effects on test cases.

- **Algorithm Design:** Using the observed effects, design a set of algorithms that will be able to apply these updates to an input set of test cases based on two consecutive versions of a model.
- **Prototype Tool Development:** Using the developed algorithms, develop a tool that is capable of model-based test co-evolution, along with several other testing processes, in order to obtain a single testing tool.

## IV. LIMITATIONS AND RISKS

A possible risk in this proposed work is the availability of a substantial set of models and tests. While we intend on working with models obtained from General Motors, these may not be made fully available to us when needed during the research, thus several sets of open source models have been selected as replacements.

## V. CONTRIBUTIONS

Our work will make the following contributions to the fields of model-based testing, and automotive software development:

- provide a methodology for co-evolution of model-based tests
- produce a catalog of evolution patterns for model development in Simulink
- automatically identify areas of test models affected by evolution of models
- increase the efficiency of MBT test evolution process
- provide a test evolution workbench for industrial automotive model development

## REFERENCES

- [1] E. Bringmann and A. Kramer. Model-based testing of automotive systems. In *Proceedings of the First International Conference on Software Testing, Verification, and Validation*, ICST '08, pages 485 – 493, April 2008.
- [2] Q. Farooq, M. Iqbal, Z.I. Malik, and Matthias Riebisch. A model-based regression testing approach for evolving software systems with flexible tool support. In *17th IEEE International Conference and Workshops on Engineering of Computer Based Systems (ECBS)*, pages 41–49, 2010.
- [3] E.J. Rapos and J. Dingel. Incremental test case generation for UML-RT models using symbolic execution. In *Proceedings of the Fifth International Conference on Software Testing, Verification and Validation*, ICST '12, pages 962–963, Montreal, Canada, April 2012.
- [4] Eric James Rapos. Understanding the effects of model evolution through incremental test case generation for UML-RT models. Masters thesis, Queen's University, Kingston, ON, September 2012.
- [5] Philipp Zech, Michael Felderer, Philipp Kalb, and Ruth Breu. A generic platform for model-based regression testing. *Leveraging Applications of Formal Methods, Verification and Validation. Technologies for Mastering Change*, 7609:112–126, 2012.