Test-Case Generation for Simulink/Stateflow Models of Software-Intensive Mission-Critical Systems (Abstract)

Corina S. Păsăreanu[1], Johann M. Schumann[1], Peter C. Mehlitz[1], Gabor Karsai[2], Harmon Nine[2], Sandeep Nima[2]
[1]NASA Ames Research Center, Moffett Field, CA 94035, USA
[2]ISIS, Vanderbilt University, USA

[1]{corina.s.pasareanu, johann.m.schumann, peter.c.mehlitz}@nasa.gov
[2]{gabor, hnine, sandeep}@isis.vanderbilt.edu

Development projects for modern Aerospace software as well as NASA mission software are increasingly using the Mathworks tool suite Simulink and Stateflow [MATHWORKS] for the modeling of safety-critical software systems, and for the automatic generation of flight code (with RealTime Workshop). Therefore, adequate sets of test cases are necessary during validation (e.g., unit testing) for testing functionality and to obtain the required code coverage.

Toward this end, we have developed an automated test-case generation framework for Simulink/Stateflow models that aims to provide seamless integration with model-based development frameworks. We have used the framework to generate test-cases for parts of the Guidance Navigation and Control (GN&C) code that will be flight-tested on the new Orion space capsule. The enabling technologies for test-case generation are software model checking, symbolic execution, and constraint solving, combined to work in a synergistic way in the Ames' Symbolic (Java) PathFinder tool [SPF].

In our framework, Simulink/Stateflow models are first translated into an intermediate common representation that is specially tailored for analysis. This translation is performed by the MICTES tool-suite from Vanderbilt University [MICTES].

The translated models are then fed to Symbolic PathFinder for analysis. Symbolic PathFinder generates test cases (i.e., test vectors or test sequences) to ensure the desired coverage of the models (e.g., state, transition, path, MC/DC or some other user-specified test coverage). Symbolic PathFinder also checks functional properties, during the test case generation process.

Test cases can be fed back to model simulators (e.g., Matlab's simulator) or can be used to test the code generated from the models. While the code is not necessarily auto-generated from the models, we do assume a close correspondence between models and code.

The test cases generated by Symbolic PathFinder can reveal problems such as un-covered code, undesired discrepancies between models and code, etc. We believe that such testing should complement other analysis and testing activities at the code level. The test cases can be used for the following activities: *testing the code* generated from the models, *validating the model transformation* (e.g. by running them against Matlab's simulator), and *validating the code generators*.

Our framework is able to automatically analyze complex models, that combine Stateflow and Simulink models with Embedded Matlab code and have features that *are currently not handled by commercial tools*, such as T-VEC [TVEC] and Mathwork's Design Verifier [DV].

In the paper, we will describe in detail the framework architecture and its underlying technologies. We will also discuss results from our case studies of flight-critical software systems, will comment on related (commercial) tools, and will describe directions for future work.

REFERENCES

[MATHWORKS] http://www.mathworks.com

[SPF] C. S. Pasareanu, P. C. Mehlitz, D. H. Bushnell, K. Gundy-Burlet, M. R. Lowry, S. Person, M. Paper: *Combining unit-level symbolic execution and system-level concrete execution for testing NASA software*. Proceedings of International Symposium on Software Testing and Analaysis 2008: 15-26.

[MICTES] G. Karsai, A. Ledeczi, S. Neema, J. Sztipanovits: *The Model-Integrated Computing Toolsuite: Meta-programmable Tools for Embedded Control System Design*. Proceedings of 2006 IEEE International Symposium on Computer-Aided Control Systems Design: 50-55.

[TVEC] http://www.t-vec.com/

[DV] http://www.mathworks.com/products/sldesignverifier/

Filename: 13.doc
Directory: D:\F Drive_new\www-smcit09\abstracts-contri_papers
Template: C:\Documents and Settings\klittle\Application Data\Microsoft\Templates\Normal.dot
Title: .
Subject: IEEE Transactions on Magnetics
Author: -
Keywords:
Comments:
Creation Date: 11/3/2008 12:41:00 PM
Change Number: 5
Last Saved On: 11/3/2008 12:46:00 PM
Last Saved By: Corina Pasareanu
Total Editing Time: 6 Minutes
Last Printed On: 5/5/2009 1:59:00 PM
As of Last Complete Printing
Number of Pages: 1
Number of Words: 552 (approx.)
Number of Characters: 3,490 (approx.)