Automatically finding bugs in a commercial cyber-physical system development tool chain with SLforge

Abstract:

Cyber-physical system (CPS) development tool chains are widely used in the design, simulation, and verification of CPS data-flow models. Commercial CPS tool chains such as MathWorks's Simulink generate artifacts such as code binaries that are widely deployed in embedded systems. Hardening such tool chains by testing is crucial since formally verifying them is currently infeasible. Existing differential testing frameworks such as CyFuzz can not generate models rich in language features, partly because these tool chains do not leverage the available informal Simulink specifications. Furthermore, no study of existing Simulink models is available, which could guide CyFuzz to generate realistic models. To address these shortcomings, we created the first large collection of public Simulink models and used the collected models' properties to guide random model generation. To further guide model generation we systematically collected semi-formal Simulink specifications. In our experiments on several hundred models, the resulting SLforge generator was more effective and efficient than the state-of-the-art tool CyFuzz. SLforge also found 8 new confirmed bugs in Simulink.

Biography:

Christoph Csallner is an Associate Professor in the Computer Science and Engineering Department at the University of Texas at Arlington (UTA). Before joining UTA, Dr. Csallner worked for Google and Microsoft Research. He graduated with a Diplom-Informatiker degree from Universität Stuttgart, Germany, and with a Ph.D. in Computer Science from Georgia Tech.

Dr. Csallner has broad research interests in software engineering and related areas. Currently he is working on problems in program analysis, automated bug finding, and mobile software engineering. Dr. Csallner’s work has received several best paper awards and distinguished paper awards, including at the IEEE International Symposium on Software Reliability Engineering (ISSRE) in 2010, at the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA) in 2006 and 2012, at the Program Protection and Reverse Engineering Workshop (PPREW) in 2014, and at the IEEE/ACM International Conference on Automated Software Engineering (ASE) in 2007 and 2015. Dr. Csallner's work has been supported by MathWorks and the National Science Foundation.