2011

Unifying Testing and Analysis through Behavioral Coverage

Matthew B. Dwyer
University of Nebraska - Lincoln, dwyer@cse.unl.edu

2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE)
Digital Object Identifier: 10.1109/ASE.2011.6100066

Unifying Testing and Analysis through Behavioral Coverage

Matthew B. Dwyer
University of Nebraska, USA

Abstract

The past decades have produced a wide-variety of automated techniques for assessing the correctness of software systems. In practice, when applied to large modern software systems all existing automated program analysis and verification techniques come up short. They might produce false error reports, exhaust available human or computational resources, or be incapable of reasoning about some set of important properties. Whatever their shortcoming, the goal of proving a system correct remains elusive.

Many people believe that, after an initial period of development, software systems are "mostly" correct — systems have much more correct behavior than incorrect behavior. Following this line of thinking, we explore what it means to re-orient program analysis and verification techniques away from focusing on proving properties. Rather, we explore how to develop and leverage techniques that characterize the subset of system behaviors that can be shown to be consistent with property specifications.

We describe the challenges in producing a rich suite of evidence-producing automated verification and validation techniques and suggest one approach to overcoming those challenges. We then describe the promise that combining such techniques offers — the weaknesses of one technique can be masked by the strengths of another, the results of one technique can be used to target the application of another, and evidence from multiple techniques can be combined to produce an explicit characterization of what is known about system correctness.

Biography

Matthew B. Dwyer is the Henson Professor of Software Engineering in the Department of Computer Science and Engineering at the University of Nebraska – Lincoln. He received the BS in Electrical Engineering in 1985 from the University of Rochester and worked for six years as a Senior Engineer with Intermetrics Inc. developing compilers and software for safety-critical embedded systems. His interests in verification and validation of embedded systems software led him to pursue the PhD at the University of Massachusetts at Amherst which he earned in 1995.

Dr. Dwyer is an active member of the software engineering, computer-aided verification and program analysis research communities, and has chaired program committees of top meetings in those areas (FSE, ICSE, OOPSLA). He has served as Associate Editor of ACM Transactions on Programming Languages and Systems and of IEEE Transactions on Software Engineering. His interests in software specification, dataflow analysis, symbolic execution, software testing, and runtime monitoring have resulted in more than 100 published research articles. That work has been recognized with an NSF CAREER award, distinguished paper awards, and, most recently, the ICSE “Most Influential Paper” and SIGSOFT “Impact Paper” awards in 2010.

978-1-4577-1639-3/11/$26.00 © 2011 IEEE ASE 2011, Lawrence, KS, USA