DART: Directed Automated Random Testing

Koushik Sen

UC Berkeley

Abstract. Testing with manually generated test cases is the primary technique used in industry to improve reliability of software–in fact, such testing is reported to account for over half of the typical cost of software development. I will describe directed automated random testing (also known as concolic testing), an efficient approach which combines random and symbolic testing. Concolic testing enables automatic and systematic testing of programs, avoids redundant test cases and does not generate false warnings. Experiments on real-world software show that concolic testing can be used to effectively catch generic errors such as assertion violations, memory leaks, uncaught exceptions, and segmentation faults. From our initial experience with concolic testing we have learned that a primary challenge in scaling concolic testing to larger programs is the combinatorial explosion of the path space. It is likely that sophisticated strategies for searching this path space are needed to generate inputs that effectively test large programs (by, e.g., achieving significant branch coverage). I will present several such heuristic search strategies, including a novel strategy guided by the control flow graph of the program under test.

K. Namjoshi, A. Zeller, and A. Ziv (Eds.): HVC 2009, LNCS 6405, p. 4, 2011.