Effective Methods for Debugging Concurrent Software

Multicore is here to stay. Software developers are moving to concurrent programming. However, this move is slow and challenging due to the exponential complexity in reasoning about concurrency. In particular, “Heisenbugs” such as data races, which are non-deterministic concurrency errors, pervasively infect concurrent software, making concurrent program debugging notoriously difficult.

In this talk, I will discuss some of the major research challenges in concurrent program debugging together with several effective methods, developed throughout my Ph.D. and Postdoc research. I will focus on three techniques: LEAP - an order-based replay tool for multithreaded Java programs, CLAP - a search-based system that uses SMT solvers and thread control-flow profiles to reproduce concurrency errors in C/C++ programs, and RVPredict - a trace-based data race detector that precisely predicts all possible races that can be found based on a single trace.

Dr. Jeff Huang is currently a Postdoctoral Researcher at the Department of Computer Science, University of Illinois at Urbana-Champaign. He received his Ph.D. from The Hong Kong University of Science and Technology in 2012. His research focuses on developing techniques and tools for improving concurrent software quality and reliability based on fundamental program analyses and programming language theory. He received 2013 SIGSOFT Outstanding Dissertation Award and 2013 PLDI Distinguished Paper Award. He was also the winner of 2012 PLDI Student Research Competition and a winner of 2010 Professor Samuel Chanson Best Teaching Assistant Award.