

Thesis Proposal

Institute for Software Research
Software Engineering



Applying Variational Execution to Tackle Exponentially Large Search Spaces

Chu-Pan Wong

Tuesday, December 17th, 2019
8:30 AM - 11:00 AM
Gates & Hillman Center 6501

Variations are ubiquitous in software. Some variations are intentionally introduced, e.g., to provide extra functionalities or tweak certain program behaviors, while some variations are speculatively generated to achieve certain search goals, such as randomly mutating a buggy program to repair a bug. Although program variations provide great flexibility, their interactions are difficult to manage, as the number of possible interactions grows exponentially with the number of variations. Despite the challenges, there is increasing evidence showing that it is important to study interactions among variations in various domains, such as testing highly configurable systems, secure information flow tracking, higher-order mutation testing, and automatic program repair. In this thesis, we tackle exponentially large search spaces of interactions among variations.

Among existing approaches that study interactions among intentional variations, a recent dynamic analysis technique called variational execution has been shown to be promising. Variational execution can efficiently analyze many variations and keep track of their interactions accurately, by aggressively sharing redundancies of program executions. While existing use of variational execution has focused on intentional variations, we argue that variational execution is also useful for studying interactions among speculative variations, which remains an open challenge despite many years of research.

To study interactions among speculative variations, we set out to improve the scalability and extensibility of variational execution by using transparent bytecode transformation. With improved variational execution, not only can we extend existing work on intentional variations, but also open new avenues for analyzing speculative variations in higher-order mutation testing and automatic program repair.

Automatic program repair and higher-order mutation testing often use search-based techniques to find optimal or good enough solutions in huge search spaces of speculative variations. As search spaces continue to grow, finding solutions that require interactions of multiple variations can become challenging. To tackle the huge search spaces, we propose to encode the search problems as Boolean satisfiability problems, and use variational execution and SAT solving techniques to iterate all solutions efficiently. After identifying a complete set of solutions, we further study their characteristics to understand their nature and learn useful insights to inspire new ideas, such as more effective and lightweight metaheuristic search strategies.

**Committee: Dr. Christian Kästner (Chair), Dr. Claire Le Goues,
Dr. Heather Miller, Dr. Abhik Roychoudhury (National University of Singapore)**