

# YU GAO

Institute of Software, Chinese Academy of Sciences

gaoyu15@otcaix.iscas.ac.cn

<http://www.tcse.cn/~gaoyu15/>

## RESEARCH INTERESTS

---

Software reliability, program analysis, testing and analysis of distributed systems

## EDUCATION

---

### PhD student, Software Engineering

*Sept. 2017 - present*

Institute of Software, Chinese Academy of Sciences, Beijing, China

Advisor: Jun Wei, Wensheng Dou

### Master student, Software Engineering

*Sept. 2015 - June. 2017*

Institute of Software, Chinese Academy of Sciences, Beijing, China

Advisor: Jun Wei, Wensheng Dou

### B.E., Software Engineering

*Sept. 2010 - June. 2014*

Nankai University, Tianjin, China

## PUBLICATIONS

---

- 1 **Yu Gao**, Wensheng Dou, Feng Qin, Chushu Gao, Dong Wang, Jun Wei, Ruirui Huang, Li Zhou, Yongming Wu. An Empirical Study on Crash Recovery Bugs in Large-Scale Distributed Systems. In Proceedings of the 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2018). Acceptance rate: 21%.
- 2 Jie Wang, Wensheng Dou, Chushu Gao, **Yu Gao**, Jun Wei. Context-Based Event Trace Reduction in Client-Side JavaScript Applications. In Proceedings of the 11th IEEE Conference on Software Testing, Validation and Verification (ICST 2018). Acceptance rate: 25.2%.
- 3 Jie Wang, Wensheng Dou, **Yu Gao**, Chushu Gao, Feng Qin, Jun Wei. A Comprehensive Study on Real World Concurrency Bugs in Node.js. In Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2017). Acceptance rate: 21%.

## RESEARCHES

---

### 1. Understanding and combating crash recovery bugs in distributed systems

*Accepted by ESEC/FSE 2018*

Current distributed systems have various crash recovery mechanisms to combat node crashes. However, faults in these mechanisms and their implementations can introduce intricate crash recovery bugs. We have performed a comprehensive empirical study on 104 crash recovery bugs from four open source distributed systems to understand them deeply (ESEC/FSE18). We now focus on crash recovery bug detection and testing.

### 2. Understanding and combating concurrency bugs in Node.js

*Accepted by ASE 2017*

As a popular server-side JavaScript runtime environment, Node.js adopts an event-driven model, and supports asynchronous I/O. The asynchrony and non-determinism of event processing in Node.js can introduce intricate concurrency bugs, which are unique to Node.js. We have performed an in-depth study on 57 real world concurrency bugs in Node.js (ASE17). We further promote effective techniques to detect this kind of concurrency bugs.

### **3. Trace reduction in client-side JavaScript applications**

*Accepted by ICST 2018*

Trace reduction can help remove failure-irrelevant events from a failure trace produced by a client-side JavaScript application. Delta debugging adopts the divide-and-conquer algorithm to generate a minimal event subtrace. However delta debugging is slow because it may generate lots of syntactically infeasible candidate event subtraces. We proposed EvMin, a context-based trace reduction technique which avoids generating syntactically infeasible event subtraces, and dramatically speeds up delta debugging.

### **AWARDS AND HONORS**

---

Merit Student, University of Chinese Academy of Sciences	<i>2018</i>
National Scholarship, University of Chinese Academy of Sciences	<i>2017</i>
Merit Student, University of Chinese Academy of Sciences	<i>2017</i>