1. SEADS: Scalable and Cost-effective Dynamic Dependence Analysis of Distributed Systems via Reinforcement Learning

Xiaoqin Fu (1); Haipeng Cai (1); Wen Li (1); Li Li (1)

Source: *ACM Transactions on Software Engineering and Methodology*, v 30, n 1, p 10 (45 pp.), Jan. 2021; **ISSN:** 1049-331X; **DOI:** 10.1145/3379345; **Publisher:** ACM, USA

Author affiliation: (1) Washington State University, School of Electrical Engineering and Computer Science, Washington, DC, United States

Abstract: Distributed software systems are increasingly developed and deployed today. Many of these systems are supposed to run continuously. Given their critical roles in our society and daily lives, assuring the quality of distributed systems is crucial. Analyzing runtime program dependencies has long been a fundamental technique underlying numerous tool support for software quality assurance. Yet conventional approaches to dynamic dependence analysis face severe scalability barriers when they are applied to real-world distributed systems, due to the unbounded executions to be analyzed in addition to common efficiency challenges suffered by dynamic analysis in general.In this article, we present SEADS, a distributed, online, and cost-effective dynamic dependence analysis framework that aims at scaling the analysis to real-world distributed systems. The analysis itself is distributed to exploit the distributed computing resources (e.g., a cluster) of the system under analysis; it works online to overcome the problem with unbounded execution traces while running continuously with the system being analyzed to provide timely querying of analysis results (i.e., runtime dependence set of any given query). Most importantly, given a user-specified time budget, the analysis automatically adjusts itself to better cost-effectiveness tradeoffs (than otherwise) while respecting the budget by changing various analysis parameters according to the time being spent by the dependence analysis. At the core of the automatic adjustment is our application of a reinforcement learning method for the decision making-deciding which configuration to adjust to according to the current configuration and its associated analysis cost with respect to the user budget. We have implemented SEADS for Java and applied it to eight real-world distributed systems with continuous executions. Our empirical results revealed the efficiency and scalability advantages of our framework over a conventional dynamic analysis, at least for dynamic dependence computation at method level. While we demonstrate it in the context of dynamic dependence analysis in this article, the methodology for achieving and maintaining scalability and greater cost-effectiveness against continuously running systems is more broadly applicable to other dynamic analyses. (0 refs)

Inspec controlled terms: decision making - distributed processing - Java - learning (artificial intelligence) - program diagnostics - query processing - software fault tolerance - software maintenance - software quality - system monitoring

Uncontrolled terms: software quality assurance - dynamic dependence analysis - real-world distributed systems - unbounded execution - SEADS - distributed computing resources - runtime dependence set - cost-effectiveness tradeoffs - analysis parameters - associated analysis cost - conventional dynamic analysis - dynamic dependence computation - distributed software systems - scalable and cost-effective dynamic dependence analysis - runtime program dependencies analysis - scalability barriers - unbounded execution traces - timely querying - decision making - Java

Classification Code: C6150G Diagnostic, testing, debugging and evaluating systems - C6263 - C1140Z Other topics in statistics - C6110B Software engineering techniques - C6110J Object-oriented programming

IPC Code: G06F9/44 - G06F11/36 - G06N20/00

Treatment: Practical (PRA) - Theoretical or Mathematical (THR)

Database: Inspec

Data Provider: Engineering Village