An Empirical Comparison of Test Suite Reduction Techniques for User-session-based Testing of Web Applications

Sara Sprenkle, Sreedevi Sampath, Emily Gibson, and Lori Pollock
University of Delaware

Amie Souter
Drexel University

Goal: Compare the cost effectiveness of reduction techniques for Web application test suites

Motivation

- Test Web applications for correctness
- Collect user sessions as test cases
  - Representative of application's evolving operational profile
- Problem: expensive to collect and replay large number of user sessions
- Solution: *Test Suite reduction* - reduce cost of storage, replay without losing original suite's fault detection ability
- Goal: Empirical comparison of costs effectiveness of reduction techniques

Research Questions

- How much reduction in size of the test suite is obtained by the various reduction techniques?
- How effective is the program coverage of the reduced suites as compared to the original suite?
- How effective are the reduced suites when compared to the original suite with respect to fault detection?
- What are the relative time and space costs of the reduction techniques?

Subject Web Applications

- Bookstore
  - Open-source, e-commerce application
- Course Project Manager (CPM)
  - Course management tool for instructors and students

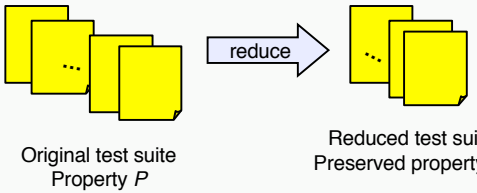
Application	Classes	Methods	NCLOC	Faults	User Sessions	URLs req	Avg. US size
Bookstore	11	383	7791	40	123	3640	29 URLs
CPM	72	174	9542	60	73	1423	19 URLs

Summary of Results

- Reducing using concept analysis is a promising, cost-effective technique
  - reduces suite to size similar to requirements-based approaches
  - reduced suites have similar coverage, fault detection without overhead of requirements mapping

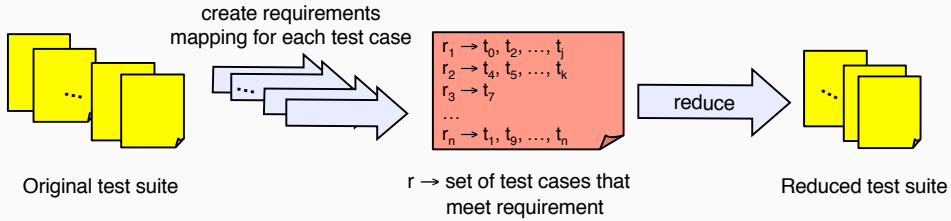
Reduction Techniques

General Reduction Algorithm

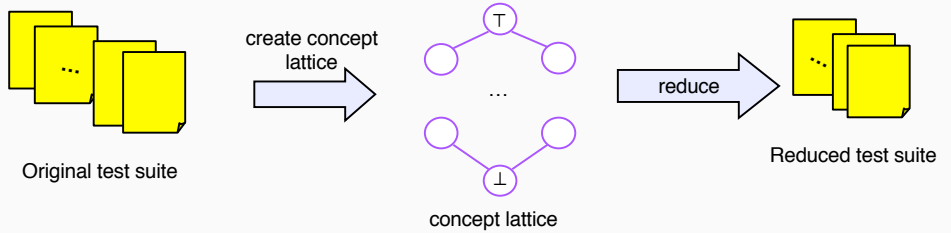


Examples of properties to preserve:
- program coverage
  - statement, method, condition, MC/DC
- fault detection
- URL coverage

Requirements-based
Variations
- Random
- Greedy
- HGS [Harrold 93]

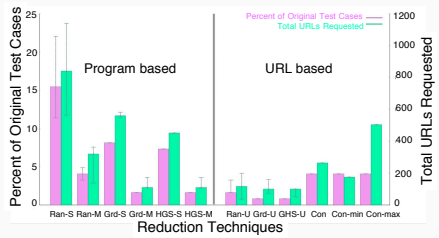


Concept Analysis Approach
[Sampath 04]
Variations: selection heuristic



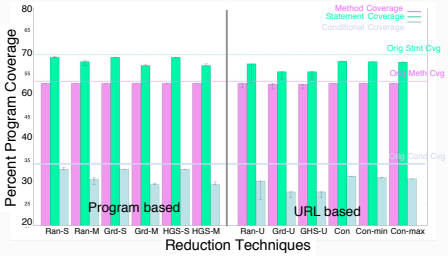
Empirical Results for Bookstore Application's Reduced Test Suites

Reduced Test Suite Size



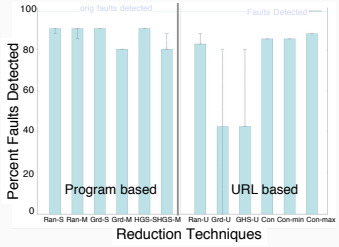
- URL-based techniques tend to have smaller test suites

Reduced Test Suite Coverage



- Similar program coverage regardless of reduced test suite's size

Reduced Test Suite Fault Detection



- Concept analysis approaches find the same percentage of faults with fewer test cases