Optimization of Software Test Case Selection Using Genetic Algorithms

Rodrigo Silva-Lugo
Outline

1. Introduction
2. Problem Formulation
3. Solution Approach
4. Evaluation
Introduction
Background

- Software development process:
  - Design
  - Implementation
  - Testing
- Testing phase can consume 30%-60% of total development effort.
- In some cases the number of TCs may be large (in the 1,000s)
Motivation

• Application of TCs during development and testing becomes expensive (time, money, equipment personnel).
• A *subset* of all TCs may be applied and still satisfy testing requirements.
• How to select such TC subset while satisfying testing requirements and resources constraints?
Goals

• To solve the TC selection optimization problem.
• Use a multi-objective genetic algorithm for optimization.
• Compare obtained results to existing solutions.
Contributions

- Most existing approaches focus on coverage and time as objectives.
- Adding maximization of likely failures testing as optimization objective.
- Adding minimization of computational load as optimization objective.
Problem Formulation
Problem Formulation

Given a Software under test (SUT) $S$, a set of TCs $T$, find a subset $T^*$ of TCs such that applying $T^*$ to $S$ will:
1. maximize coverage
2. minimize computational load
3. maximize testing of likely failures
4. satisfy time constraints
A testing environment $E$ which is composed of the SUT $S$, testing requirements $R$, a set $T$ of TCs, and defect history $D$ for $S$:

$$E = (S, R, T, D)$$
Objectives

• Maximize TC Coverage
• Minimize Computational load
• Maximize Testing for likely failures
Objectives

- Maximize TC coverage
- Selected TCs
- Requirement function
- Set of requirements

\[ K = \frac{\left| \bigcup_{\tau \in T^*} f_\rho(\tau) \right|}{|R|} \]

\[ T^* \subset T \]

\[ f_\rho : T \rightarrow \mathcal{P}(R) \]

\[ R = \{ \rho_i : 1 \leq i \leq q \} \]
Objectives

• Maximize testing for likely failures

• Failure function

• Failure probability of requirements covered by selected TCs

• Probability of failure for requirement

\[ \xi = \frac{\psi}{\sum_{\rho \in R} \pi_{\rho}} \]

\[ f_{\phi}(\tau) = \sum_{\rho \in f_{\rho}(\tau)} \pi_{\rho} \]

\[ \psi = \sum_{\tau \in T^*} f_{\phi}(\tau) \]

\[ \pi_{\rho} \in [0, 1] \]
Objectives

• Minimize computational load

• Load finding critical TCs

• Load finding optimal set of TCs with GA

• Load applying TCs to SUT

\[ \lambda = \lambda_1 + \lambda_2 + \lambda_3 \]

\[ \lambda_1 \]

\[ \lambda_2 \]

\[ \lambda_3 \]
Output

- A subset of selected TCs. \( T^* \subset T \)
- Satisfies all 3 objectives
  \[
  \begin{align*}
  \text{max } K \\
  \text{max } \xi \\
  \text{min } \lambda
  \end{align*}
  \]
Solution Approach
Methodology

- Obtain datasets
- Implement Genetic Algorithm (GA) to perform optimization of TC selection
- Evaluate how the results of the GA satisfy the TC selection optimization problem.
Software-Artifact Infrastructure Repository (SIR)

- SIR object = SUT
- For different versions of the SUT we have:
  - Test Cases
  - Coverage Information
  - Defect History
## Software Objects

<table>
<thead>
<tr>
<th>SUT</th>
<th>LoC</th>
<th>Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>jtopas</td>
<td>5,400</td>
<td>4</td>
</tr>
<tr>
<td>xml-security</td>
<td>16,800</td>
<td>4</td>
</tr>
<tr>
<td>jmeter</td>
<td>43,400</td>
<td>7</td>
</tr>
<tr>
<td>ant</td>
<td>80,500</td>
<td>11</td>
</tr>
<tr>
<td>jboss</td>
<td>116,638</td>
<td>10</td>
</tr>
<tr>
<td>derby</td>
<td>503,833</td>
<td>5</td>
</tr>
</tbody>
</table>
NSGA-II Algorithm

- Multi-objective evolutionary algorithm.
- Non-dominated sorting-based.
- Fast: $O(MN^2)$ time complexity vs $O(MN^3)$
  - $M$: number of objectives
  - $N$: population size
- Elitist approach: history of best found solutions
NSGA-II Algorithm

The boxed points represent feasible choices, and smaller values are preferred to larger ones. Point C is not on the Pareto Frontier because it is dominated by both point A and point B. Points A and B are not strictly dominated by any other, and hence do lie on the frontier.
NSGA-II Individual Representation

• Binary chromosomes:
  • TC is 1: selected, 0: not selected
  • Chromosome size: number of total TCs
  • e.g. for JTopas $2^{18}$ possible solutions
NSGA-II Fitness Function

- 3 optimization objectives
  - Coverage: normalized [0,1]
  - Fault history: normalized [0, 1]
  - Computational load
Overview
Evaluation
Parameters

- Redundancy: When some of the requirements are tested by more than one TC

\[ \alpha = \frac{\sum_{\tau \in T^*} |f_\rho(\tau)| - |\bigcup_{\tau \in T^*} f_\rho(\tau)|}{|R|} \]

- Selected TCs: \( T^* \subset T \)

- Failure function: \( f_\rho : T \rightarrow \mathcal{P}(R) \)

- Requirements: \( R = \{\rho_i : 1 \leq i \leq q\} \)
Parameters

• Lack of coverage: Some of the requirements may not be tested by any of the TCs in the selected subset. The ratio of the number of requirements not tested to the total number of requirements gives a measure of lack of coverage.

\[
\gamma = \frac{|R| - |\bigcup_{\tau \in T^*} f_\rho(\tau)|}{|R|}
\]
Parameters

- Lack of coverage: \( \gamma = \frac{|R| - \left| \bigcup_{\tau \in T^*} f_\rho(\tau) \right|}{|R|} \)

- Requirements: \( R = \{ \rho_i : 1 \leq i \leq q \} \)

- Selected TCs: \( T^* \subset T \)

- Requirement function: \( f_\rho : T \rightarrow \mathcal{P}(R) \)
Parameters

• Computational load: The measure of total resources employed to obtain the selected TCs, and how that value satisfies the constraints of the optimization problem.

\[ \lambda = \lambda_1 + \lambda_2 + \lambda_3 \]
Parameters

- Computational load
  \[ \lambda = \lambda_1 + \lambda_2 + \lambda_3 \]
- Load finding critical TCs
  \[ \lambda_1 \]
- Load finding optimal set of TCs with GA
  \[ \lambda_2 \]
- Load applying TCs to SUT
  \[ \lambda_3 \]
Parameters

- Failure probability: The cumulative probability of the requirements tested by the selected TCs to fail. This measures the likelihood of the selected TCs to detect failures based on defect history.

\[ \xi = \frac{\psi}{\sum_{\rho \in R} \pi_{\rho}} \]
Parameters

• Failure probability:
  \[ \xi = \frac{\psi}{\sum_{\rho \in R} \pi_{\rho}} \]

• Failure function:
  \[ f_{\phi}(\tau) = \sum_{\rho \in f_{\rho}(\tau)} \pi_{\rho} \]

• Failure probability of requirements covered by selected TCs:
  \[ \psi = \sum_{\tau \in T^*} f_{\phi}(\tau) \]

• Probability of failure for requirement:
  \[ \pi_{\rho} \in [0, 1] \]
Comparison

• Compare results obtained with the proposed approach with existing solutions in the literature and baseline case:
  • Optimization quality
  • Scalability
  • Application domain (e.g. regression only)
Review

• Solve the TC selection optimization problem.
• Use a multi-objective genetic algorithm for optimization.
• Compare obtained results to existing solutions.
umpeople.memphis.edu/rsilva

Rodrigo.Silva@memphis.edu