

# Optimization in Software Testing using Genetic Algorithm

Jinkal Javia, Arpita Gupta, Sapan Gandhi

**Abstract**— The software should be reliable and free from errors. Software testing is an important part of the software development life cycle. Software testing is the critical element of the software quality assurance and gives the ultimate review of the specification, analysis, design and development. As the human is prone to errors, the software development life cycle should be accompanied by the quality assurance mechanisms. Hence, the software testing comes into the picture. Well defined test case is the key to success. Time, cost and other parameters does not permit us to perform the exhaustive testing. This arise the need for the automated testing process. The intention is to find maximum the number of errors and with the minimum effort and time. This paper presents the optimization of testing in software engineering using the Genetic algorithm (GA). Genetic Algorithm is used for the solving of the non linear problem. The software path clusters are generated by GA in accordance with the criticality of the path and tested. The paper discusses some key concepts of the Genetic Algorithm viz. Selection, Crossover, Mutation. The paper shows that GA has positive influence on the performance of the software test cases.

**Index Terms**— Crossover, Genetic Algorithm, Mutation, Optimization, Selection, Software testing.

---

## 1 INTRODUCTION

Software Testing is introduced as to remove the errors encountered during the design and the implementation phase. The presence of errors in system and application software causes remarkable increase in the costs, mainly because of the damages provoked and the need for the corrections for the corresponding errors. Software Testing is the activity aimed to discover the unavoidable errors in the software products in the most optimal way possible as to detect the maximum number of errors and at the same time minimize the testing cost and time. Based on the parameters like efficiency, probability, usability, capability, etc. testing is used to generate the estimation about the quality of the software. Testing in the Software development life cycle is very crucial phase as it enhances the reliability of the software. Reliability can be defined as the probability that the system works without errors within a given period of time.

**Definition:** *Software testing consists of the dynamic verification of behavior of the program on the finite set of test cases, suitably selected from the usually infinite executions domain, against the specified expected behavior[2].*

Testing consumes lot of resources in terms of time and memory for the detection of the errors. Thus susceptible to optimization i.e. the best equilibrium between the number of test cases generated and the expected value for the discovery of the errors that has to be achieved. On the basis of the intrinsic probability of the occurrence of the error, the value of the test case execution may be assigned.

Purpose of testing is more than just debugging and finding of bugs. It includes the purpose[5]:

1. Improving and assuring the quality of software.
2. Verification and validation.
3. Estimation reliability.

Automation of the test case generation helps in reducing the cost significantly. Many methods are proposed for developing test cases automatically like using neural networks, fuzzy logic, machine learning, probabilistic reasoning, evolutionary computation and many more. Genetic algorithm has been applied in many optimization problems for the generation of the test plans for the purpose of functionality test, feasibility test, black box test and in object oriented unit test.

The paper is organized in 4 sections. Section 2 discusses the working of GA, Section 3 provides the algorithmic procedure used for the implementation purpose, Section 4 includes the observations obtained by the implementation of the modules in MATLAB version 7.0 and Section 5 comprises of the conclusion.

## 2 GENETIC ALGORITHM: A BRIEF INTRODUCTION

Genetic algorithm falls in the category of evolutionary algorithms which are used for solving many real life problems. GA is the robust optimization techniques and search method. Genetic algorithm is inspired by the evolution theory of the nature i.e. way of selecting the fittest individuals. In the case of software testing, the possible solution test cases are treated as population of chromosomes. Each chromosome is made up of string of binary digits called gene. In the case considered attributes are treated as genes. This initial population (possible test cases) can be generated in the random manner. One of the possible ways for generating can be using greedy algorithm[3].

**Genetic Algorithm:**

Initialize (population);

```
Evaluate (population);  
while (criteria not satisfied)  
{  
    Selection;  
    Crossover (population);  
    Mutation (population);  
    Evaluate (population);  
}
```

It consists of three basic operations: 1) selection, 2) crossover, 3) mutation

### 2.1 Selection

Selection is applied to select chromosomes from the existing population based on their fitness. Fitness can be defined as the capability to survive and reproduce in nature. The fitness function can be defined based on the weights assigned to each of the possibility that can occur. The higher weight indicates the criticality of the path to test the data generation. For instance, the weights can be assigned in such a manner that more weight is given to loops and complicated control transfer compared to the sequential execution. The fitness function is used to select the better chromosomes for the better next generation.

### 2.2 Crossover

The crossover is carried out over the pair of selected chromosomes. For the pair of selected chromosomes, swapping of the sequence of bits is carried out[1]. The process is repeated with different parents until the intended population is achieved.

For instance,

Parent 1: aAbBcC

Parent 2: 123456

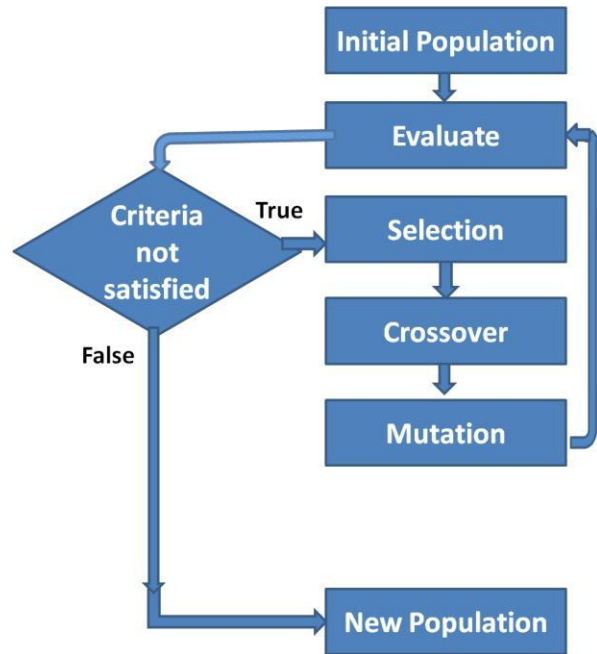
Child: aAb45 (one possibility out of many)[4].

### 2.3 Mutation

Alteration of chromosomes takes place in mutation. The main aim is to bring the diversity in the population by introducing good traits[1].

For instance, in the above discussed example  
Child: aAbZ56 (changing 4 to Z).

## 3 PROCEDURE



The procedure for genetic algorithm starts with initializing the population. It is evaluated and then the condition for selecting the chromosomes is checked on the basis of predefined criteria. If it turns to be true then the chromosomes are selected for the crossover followed by mutation process. After mutation, the new population is again evaluated and the process is repeated. On the termination of the algorithm, the new population set is ready and thus the optimal test cases are contained in the new population.

## 4 OBSERVATION

In this paper generation of test cases is carried out for the program which computes  $f(x) = x^2$  by black box testing. The input data for the testing purpose consists of valid as well as invalid data based on the defined precondition. This precondition comes from the programming constraints, specification and the computing environment. In the program containing multiple inputs, there will be set of test cases for each input. Using GA, all possible combinations for the test sets are selected and processed. The testing is carried out using random testing. The screen shots for the experiment carried out using MATLAB 7.0 are shown below.



- [4] Sanjeev Dhawan, Kulvinder S. Handa, Rakesh Kumar, Optimization of Software Testing Using Genetic Algorithm, ISSN: 1790-2769
- [5] K. Koteswara Rao, Dr. GSVP Raju, Srinivasan Nagaraj, Optimizing the Software Testing Efficiency by Using a Genetic Algorithm – A Designing Methodology, ACM SIGSOFT Software Engineering Notes, May 2013 Volume 38 Number 3.

IJSER