Survey on Automated Test Data Generation

Manpreet Kaur Brar
PG Student
CSE, RIMT-IET (PTU)
Mandi Gobindgarh, India

Sushil Garg
Professor
CSE, RIMT-MAEC (PTU)
Mandi Gobindgarh, India

ABSTRACT
Testing is the process to check the program with the aim to detect errors. Software testing is an expensive and challenging process, typically consuming at least 50% of the total development cost and 35% of total development time. The critical step of software testing is automated test data generation. Automated test data generation scheme generate optimum set of test data automatically on the basis of test data adequacy criteria. The goal of this contribution is to summarize the major techniques in recognizing the automated test data generation. The main objective of this paper is to look into the current research in the automated test data generation, various metaheuristics techniques (ACO, PSO, GA, SA etc.) and hybridization of these metaheuristics techniques (ACSGA , GAPSO etc.) all these topics have been discussed so far to mitigate the errors in the software.

Keywords
Software testing, Automated Test Data Generation, ACO, PSO, GA, Hybrid Algorithm.

1. INTRODUCTION
The software testing tries to detect and prevent errors in order to provide good quality. Software testing is very expensive and laborious activity due to the act that it is accounts to 50% of the total development cost and 35% of the total development time. Automated test data generation is an important aspect of software testing. Automated test data generation is used to generate test data automatically within the testing criteria.

Work done by previous researchers are discussed briefly, still researches are going on. Strategies explored by Eugenian et al. combined the approach of Tabu Search with Korel's chaining for generating test data automatically in order to achieve high branch coverage in software testing. For searching inside the neighbourhood of solution and remembering the best one tabu search has been used [1]. Abdelhamid did hybrid algorithm like Immune Genetic Algorithm (IGA) and compared with various evolutionary algorithms, an advanced computing technique for solving the problem of test data generation [2]. Praveen and baby utilized the use of Genetic and ant colony Optimization algorithm to

Generate optimum set of test data and compare both of them [3]. Sheng et al. proposed a hybrid algorithm based on the combination of Genetic Algorithm and Particle Swarm Optimization. This approach uses GA and PSO for cross checking the result of each other and strengthens the best part of each other [4]. Gentiana et al. had a wide range of applications covering many Evolutionary Algorithms like Particle Swarm Optimization, Simulated Annealing Algorithm and Genetic algorithm and presents a comparison between these three algorithms for generating test data which cover the target for every benchmark [5]. Minjie proposed the combine effect of Ant Colony System Algorithm and Genetic Algorithm to generate a path oriented test data, which carries on the advantage of these two mixed algorithm [6]. Rui Ding et al. proposed more efficient testing tool product which had been developed by using a combination of two Meta heuristic approaches, Genetic algorithm and particle swarm optimization. PSO has a strong local search ability which can be used to improve the deficiencies of GA [7].

The highlights of this paper are: section 2 describes the software test data Generation. Section 3 describes the different types of computing technology which are used to generate test data automatically. Sections 4 discuss the comparison between different techniques. While the paper is concluded in section 5.

2. SOFTWARE TEST DATA GENERATION
Software test data generation is a very essential and important part of software testing. Test data can be produced either manually or automatically. Software researchers put more emphasis on automated test data generation because it produces large quantities of code within less exertion. Many test data generation techniques have been automated like random test data, goal oriented test data and path oriented test data and intelligent test data generator. Random test data generation is used to take test input randomly until an efficient and useful input is found. Random test data generation is very simple and quick approach but it has one limitation i.e. poor code coverage. Goal oriented test data generation is much better than random test data generation. In the sense of generating input test data to execute the selected statement irrespective of path taken i.e. the path selection stage is eliminated. Path oriented test data generation is stronger criteria as compare to random and goal oriented test data generation. This tries to find out the percentage of code coverage to more extent and hence increase the chances of bug detection. The path oriented test data can be dividing into three parts: a program analyzer, a path selector, and a test data generation. The source code is run by a program analyzer and produced useful data to be used by a test data generation and path selector. The whole effectiveness of the system is depending on which paths are selected. Once path information is found, test data generator is used to find the input values that will traverse a specific path.

3. COMPUTING TECHNIQUES USED
In this section various methods for automated test data generation are discussed. Various existing techniques are briefly described and their limitations are pointed out. Further, new techniques based on hybrid algorithm which have evolved over the years, are also listed.

3.1 Genetic Algorithm (GA)

Genetic algorithm was introduced by professor Holland in Michigan university of United States. Genetic algorithm is one of the most popular and intensively pursued techniques which are motivated by natural evolution of spices. The use of genetic algorithm is to create the superior solution mimics for optimization and search problems. Genetic Algorithm is used to produce test cases automatically. Test cases are described by chromosomes, which consist information on the component to be produced, methods to be invoked and values to be passed as input. GA used three operators they are: selection, crossover and mutation. Selection operation is used for choosing the best individuals from current population on the basis of fitness for performing cross-over between them. There are various methods used for selection operation like tournament, roulette wheel, rank, steady state selection and many more. Crossover operator select gene from parent chromosomes and create new offspring which gives the better fitness. Mutation changes the new offspring by randomly selecting and changing its state according to the mutation probability.

3.1.1 Automated Test Data Generated by Genetic Algorithm:

Initialization*:* firstly user specified the number of elements in the population and starting and ending points of the range of population. A random population (set of solutions) is generated according to the user specifications than all the variables are initialized according to the system.

Fitness*:* The Fitness of each of the solutions is calculated on the basis of type and number of nodes traversed by this solution. If the node is simple or decision, the solution is increased by 0 or 10 and if it traverses the true part of the decision node then the fitness is further increased.

Selection*:* On the basis of the fitness of individual current solutions are selected for mutation and crossover. Two data (starting after Best Data) are selected for crossover.

Best Data: If a data in the current population traverses completely any new ACO path then this confirms that this ACO path is a feasible path and this GA data is optimal data. So this data is placed in Best Data. After sorting the population according to the fitness these best data, if any they are moved to starting positions of the current population.

Crossover: First the length of the larger parent is calculated and crossover between two selected parents is done after the bit position that is equal to the half of the number of bits of the larger parent. Fitness of the offspring is calculated, and if it is larger than the parent, parent is replaced by the offspring.

Mutation: In GA mutation is called Genetic operator which is used to maintain Genetic diversity. After performing the crossover operation if the fitness remains same, one of the bits of the parent is flipped to get a new offspring. Again fitness's of parent and offspring are compared with each other. The one with higher fitness value will survive.

Next generation: The above process is repeated for all the solutions in the initial population and the solutions with higher fitness will be selected for next population.

Termination: The complete process will be repeated either until maximum number of iterations is done which implies that test data covering all branches cannot be found.

3.2 Particle Swarm Optimization (PSO)

Kennedy and Eberhart launched Particle Swarm Optimization in 1995, which is motivated from the simulation of social behaviours of bird predation. PSO is a robust stochastic optimization technique based on the movement and intelligence of swarms. PSO has strong local search ability. It uses a number of particles that constitute a swarm moving around in the search space looking for the optimum solution. PSO is a metaheuristics algorithm, which are used to obtain good enough solutions for hard combinatorial optimization problem in a reasonable amount of computation time. In every iteration, the best previous position (which is giving the best fitness value) of each particle is called particle best , the best position among all the particle best position achieved so far is called global best and the changing rate of position for every particle is called particle velocity. The movement of each particle for new position calculated through the updated velocity. The velocity for d-dimension at iteration k of i^{th} particle is updated by:

$$v_{id}(k+1) = wv_{id}(k) + c_1 r_1 (pb_{id}(k) - x_{id}(k) + c_2 r_2 (gb_{id}(k) - x_{id}(k)) \quad (1)$$

Where i = 1, 2, ..., n and n is the total size of population, inertia weight is denoted by w, two acceleration constants are c_1 and c_2 and two random values are r_1 and r_2 in range [0,1].

The i-particle position is updated by:

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k) \quad (2)$$

3.2.1 Automated Test Data Generated by PSO

Initialization*:* Specify the particle population by randomly assigning locations (X vector for each particle) and velocities (V-vector with random or zero velocities).

Fitness: Measure the fitness of the each particle and record the best fitness value. P_{best} for individual particle till now and update P-vector related to each P_{best}. Also detect the individual's highest fitness G_{best} and record the corresponding position P. Change the velocities based on P_{best} and G_{best} position using equation (1).

Updating*:* Update the particles position by equation (2).

Termination: Terminate until requirement are met

Go to Step 2

In equation (1) above, new velocity at k + 1 is generated with the help of global fitness value where all the particles have achieved with some iteration. In this equation, the position given by the global best fitness in d dimension. Usually, global best fitness concept is expected to present possibilities of global search exploration in the search space.

3.3 Ant Colony Optimization (ACO)

M.Dorigo and a colleague introduced Ant Colony Optimization (ACO) in the early 1990s. ACO is metaheuristic technique for the solution of hard combinatorial optimization problems in a reasonable amount of computation time. ACO bring information related to foraging behaviour of real ants seeking a path between their colony and food source. When ants searching the path for their food, they initially explore the area surrounding their nest in randomly. As soon as an ant finds a source of food, it recognizes the quantity and the quality of the pheromone and carries some of the food back to the nest. While ants walking from food source to the nest, they laying down some food on the ground called pheromone. According to the smell of this laying substance ants can select their path. The more pheromone on path increase the probability of that path, will guide other ants to travel down the path and return again to their nest. This pheromone trail mechanism is an indirect communication between the ants enables them to find shortest paths collaboratively and effectively between their food sources and nest. This feature of real ant colonies is used to solve the problem of combinatorial optimization. Pheromone evaporation broadly used for avoiding the convergence to a locally optimal solution.

3.3.1 *Deciding all Possible Paths through ACO for Generating Test Data:*

ACO takes the CFG (Control flow graph) of code to be tested as input and produces all possible paths of CFG as output along with the pheromone values of each individual path. Hence, ACO ensures the entire code coverage.

Firstly, pheromone level of each node of the CFG is set to be 0. Ants are freely allowed to move in every direction in the control flow graph and their paths are recorded. While ants are traversing through the CFG, the pheromone level of each node will be increased every time an ant traverses that node and will be decreased after iteration in time domain. In this way all the possible paths will be generated along with their pheromone level.

3.4 Ant Colony System Genetic Algorithm (ACSGA)

It is a combination of Ant colony system and Genetic Algorithm. Ant colony system algorithm is advanced technique to overcome the limitation of ACO. GA will bring a problem of premature and it also has low search efficiency. In order to overcome all these limitation researchers shows some algorithm which gives better effect on search. Hybrid genetic ant colony algorithm has a good effect of controlling the local convergence and raising the search efficiency. It has the early maturity in local convergence and low effect in searching. Hybrid genetic ant colony algorithm is better than Genetic Algorithm. This hybrid algorithm is covering up the weakness of both the algorithm. In Ant Colony System Genetic Algorithm the test data is generated by using the mixed concept of both the algorithm.

3.5 Genetic Algorithm Particle Swarm Optimization (GAPSO)

It is the combination of Genetic and particle swarm optimization is called hybrid GAPSO algorithm. PSO has the strong local search capability and fast convergence characteristics, which can effectively improve the deficiencies of the GA. because all particles of the particle swarm optimization move towards its own best particles and the best particles in one-way, this one-way flow of information leads to the possibility of being trapped in local optimum, which could just be made up by the genetic algorithm. This algorithm shows the higher performance as compare to previous algorithm. The test data is generated in Genetic Algorithm Particle Swarm Optimization by using the combine power of two algorithms. This hybrid algorithm improves the power and effectiveness for solving the testing problem. Both GA and PSO are population based search method. These two algorithms share their information between their population members by using the probabilistic rules they can enhance their search process. GA-PSO had less computation time and fast convergence.

4. COMPARISON

Table1 shows the comparison between the different computing techniques that are discussed in this paper. These different techniques are used to generate test data automatically. The characteristics used to distinguish are: concept, search efficiency, iterations and result. Concept is the main step in these technologies. It is used to show how these algorithms are used in Ant Colony Optimization concept based on behaviour of ants but in particle swarm optimization based on behaviour of bird predation. Ant Colony System Genetic Algorithm and Genetic Algorithm Particle Swarm Optimization performing a concept of two mixed algorithms. Second characteristic is search efficiency i:e search for a best solution in effective manner next is the iterations for the completion of the task and last is the result analysis given by the various algorithm these includes both conventional as well as hybrid mode

Table-1: Comparison of major computing techniques of Automated Test Data Generation

Techniques	Characteristics			
	Concept	**Search efficiency**	**Iterations**	**Result**
Genetic Algorithm	Based on natural evolution	Low	High	Good
Ant Colony Optimization	Based on the foraging behavior of ants	High	Low	Feasible and effective
Particle swarm optimization	Based on behavior of bird predation	High	Low	Good
Ant Colony System Genetic Algorithm	Mixed of foraging behavior of ants and natural evolution	High	Low	Better
Genetic Algorithm Particle Swarm Optimization	Based on the mixed Natural evolution and behavior of bird predation	High	Low	Better

5. CONCLUSION

This paper's objective is to presents the major computing techniques of automated test data generation. This paper surveys some of the technique in order from the previous years. The techniques considered in this paper are Genetic Algorithm, Ant Colony Optimization, particle Swarm Optimization, hybrid Ant Colony System Genetic and Genetic Algorithm Particle Swarm Optimization. The experiment result shows that hybrid algorithm gives excellent result.

6. REFERENCES

[1] Eugenian Diaz, Javier Tuya, Raquel Blanco "Automated Software Testing Based on Tabu Search" Proceedings of the 18th IEEE International Conference on Automated Software Engineering (ASE'03), 2003.

[2] Abdelhamid Bouchachia "An Immune Genetic Algorithm for Software Test Data Generation" Seventh International Conference on Hybrid Intelligent Systems (IEEE), 2007.

[3] Praveen Ranjan Srivastava, Km Baby "Automated software testing using metahuerestic technique based on an an Ant Colony Optimization" Electronic System Design (ISED), 2010 International symposium.

[4] Sheng Zhang, Ying Zhang, Hong Zhou, Qingquan He "Automatic Path Test Data Generation Based on GA-PSO" IEEE, 2010.

[5] Gentiana Ioana Latiu, Octavian Augustin Cret, Lucia Vacariu "Automatic Test Data Generation for Software Path Testing using Evolutionary Algorithms" Third IEEE International Conference on Emerging Intelligent Data and Web Technologies,2012.

[6] Minjie Yi "The Research of path-oriented test data generation based on a mixed ant colony system algorithm and genetic algorithm" IEEE 2012.

[7] Rui Ding, Xianbin Feng, Shuping Li, Hongbin Dong "Automatic Generation of Software Test Data Based on Hybrid Particle Swarm Genetic Algorithm" IEEE Symposium on Electrical & Electronics Engineering (EEESYM),2012.

[8] Anoj Kumar, Shailesh Tiwari, K. K. Mishra and A.K. Misra "Generation of Efficient Test Data using Path Selection Strategy with Elitist GA in regression testing" Computer Science & Engineering Department MNNIT, Allahabad INDIA (IEEE),2010.

[9] Nirmal Kumar Gupta and Mukesh Kumar Rohil "Improving GA based Automated Test Data Generation Technique for Object Oriented Software" 3rd IEEE International Advance Computing Conference (IACC), 2013.

[10] Ahmed S. Ghiduk "A New Software Data-Flow Testing Approach via Ant Colony Algorithms" Universal Journal of Computer Science and Engineering Technology, Oct. 2010.