

# Optimal Test Data Generation Using Hybrid Techniques IWD & ACO

<sup>1</sup>Neelabh Sao, <sup>2</sup>Neema Patel

<sup>1</sup>Professor, <sup>2</sup>M.Tech Scholar,

<sup>1</sup>Computer Science & Engineering,

<sup>1</sup>Rungta College of Engineering & Technology , Bhilai(C.G),India

**Abstract**— As the complexity is increase in the software industry, generating an effective test data has become a necessity. Construction of test data in a software project is one of the major issues in the area of time consuming. The main agenda in the field of testing is to automate the testing process with the help of testing tools. Software testing includes test data generation, test case selection, test suite optimization. In this we proposed a technique Automated software test data generation that applies in IWD & ACO algorithm for generating the test data. Test data generation is done to satisfy all the requirements like functional, non function, business etc. Test data generation in program testing is the process of identifying a set of test data which satisfy the given test condition. In this research work, we propose an approach for automated software test data generation which will evaluating the different format of data on testing performance and estimating the computational factors like reliable of software, software quality, time and space requirements etc using integrated algorithm IWD & ACO. The main objective of this work is to enhance quality of different format of data type.

**IndexTerms**—Software Testing, Test data generation, Intelligent water drop, Ant colony optimization

## I. INTRODUCTION

Software testing is an integral part & plays a vital role in the development process of the software. Testing phase is one of the important phase in software development life cycle to develop the high quality software product. Our main focus is to provide good quality software to the users [1]. Software testing is labour intensive process which also associated with high cost. Software testing is analyzing a system or a components by providing defined inputs & comparing them with desired outputs to check the discrepancies between the desired & actual outputs & correct them [2]. Testing is the process of evaluating a system or its components with the intent to find that whether it satisfies the specified requirements or not testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to actual desire or requirements [3].

Software testing can be done in two ways manually or automatically it means software can be tested either manually or automatically. The two approaches are complementary : automated testing is a process through which thorough & fast testing can be done & it can perform a huge number of tests in short time period by using the automated software testing helps in avoiding the human errors or mistake & it also provide storing a result of the tests accurately& results can be automatically fed into a database & can be used to provide useful statistics on how the software development process is progressing, whereas manual testing done manually that it is requires human input analysis & evaluation & it also uses the knowledge of the testing engineer to target the testing components that are assumed to be more error prone. In software development software testing should not be different phase but it should be applicable throughout the maintenance phases & design development. Software testing is often used in association with terms verification & validation [4].

Test data are data which have been specifically identified for use in testing computer program. This is one of the complex activity because it involves so many steps & each sub-steps has some issues. With the help of proper test data it is easy to identify & verify the reliability & quality of the software & this process is known as software testing. Based on specification or code test data can be generated. Either manually or automatically test data can be designed.

Automated software test data generation play a significant role in the field of software testing. This is one of the research area in the field of software testing. Test data generation in program testing is the process of identifying a set of test data which satisfies the given testing criterion [5]. In a testing environment one of the important components is automatic test data generator for a given program system that automatically generate the test data. A number of test data generation techniques such as random test data generator, path oriented test data generator, goal oriented test data generator & intelligent test data generator have been automated. By using the hybrid methods for the test data generations it gives the advantage in both the method static & dynamic because it ignores the infeasible path. In the field of automated software test data generation object oriented is one of the key research area. Automated test data generation is play an significant role in reducing the cost of software test development. It satisfy the functional requirements, non-functional requirements & business requirements.

- Test data generation based on symbolic execution
- Test data generation based on concrete/actual value execution
- Automated test data generation based on symbolic & actual value execution
- Automated test data generation based on object oriented approach .

## II. LITERATURE REVIEW

Sapna Varshney & Monica Mehrotra (2014) proposed paper Automated Software Test Data to give a brief outline of automated test data generation. The basic technique for generating test data is briefly described. Automatic test data generation technique based on GA that is given by the data flow dependencies in the program. This approach can be used for programs with loops, without loops & procedures. This technique accepts input of the program to be tested. This algorithm produces a sets of test cases, the sets of def-use associations covered by each test case, & list of uncovered def-use associations. This algorithm keeps track of all the test cases that cover any new def use path. This algorithm applied to a number of classical C program such as quadratic equation problem, triangle classification problem, date problem etc. A random test data generator is also implemented for comparison with proposed approach. For the experiments it uses the input variables integer type, range 0-100 & 7 bits are used to each variable, population size is 5,10, 20, fitness function, cross over function & mutation operation are applied, cross over probability is 0.8 & mutation probability is 0.15. which show the result that the proposed technique outperformed the random testing technique in most of the programs used in the experiment in terms of the number of generations required to achieve the same def use & the size of the resulting test suite [6].

Yeresime Suresh & Santanu KU Rath (2013) focussed on test data generation basis path testing by using the techniques genetic algorithm. In this explained the soft computing which is automatically generated test data which is based upon the set of basis path by using the genetic algorithm with test data which is using optimization technique local & global search & have the capability to improve the generation capacity of test data. It helps in reducing test effort & time of a tester by using the automated process of generating test data optimally. It gives the result that around 38% of test data have higher fitness value 'f(x)' & lie in range between 1.0 & 0.7 [5].

Praveen Ranjan Srivastava et.al (2012) discussed about the test data generation based on test path discovery by using the technique intelligent water drop. In the program it discover the existing path. The IWD technique is used to generate the test path & sequence generation. For the test data generation it takes only the numeric & character types of data. It avoids the problem of consuming a large number of iterations. It gives the result that IWD is more efficient then the random process. It finds the optimal solution for automatic test path exploration [7].

Bashar Aldeeb et.al (2014) this paper has done the survey on intelligent water drop algorithm. This is one of the meta-heuristic based method which is inspired optimization algorithm from the natural phenomena of a swarm of water drops with the soil into the river bed. In this paper gives the brief description about IWD algorithm in terms of theory & application. It is easy to adapt numerous optimization problems which are relevant to both discrete & continuous problems & easy to find the optimal solutions. IWD is successful in finding the different optimization problem such as TSP multidimensional Knapsack problem etc. It gives the different application of IWD algorithms are Economic Dispatch problem, travelling salesman problem, maximum clique problem, web services composition problem, feature selection problem, steiner tree problem, robot path planning problem, automatic multilevel thresholding problem, vehicle routing problem, single UCAV smooth path planning problem, k means clustering problem, A mobile Ad-hoc network (MANET) etc. This algorithm deals with the different optimization problems to identify the appropriate solutions with optimal qualities [8].

Jon Edvardsson (2011) proposed a paper A survey on automatic test data generation. In this paper explained different methods for generating test data. The most appropriate classification in terms of test data generation are random, path oriented & goal oriented. By monitoring the program flow the system can determine if the intended path was taken. Using different kinds of searching methods the flow can altered by manipulating the input in a way that the intended branch is taken. This technique is quite expensive. It requires many iteration before a suitable input [9]

Ruchika Maholtra & Mohit Garg (2011) proposed a paper an adequacy based test data generation by using the genetic algorithm & mutation analysis. In this technique it will generate software test data using adequacy based testing criteria & genetic algorithm. Mutation analysis is one of the most common ways to generate a set of adequate test cases. In this path testing technique is selected because path testing strategy can alone detect almost 65% of errors in the program. MATLAB tool is used for genetic algorithm software. In this analysis is done on the basis of number of test cases that are generated & time required in generating test cases. Which gives the result in achieving 20%-60% savings in time to consumed to generate test cases & it gives the suggestion that the adequacy based proposed technique is better than the reliability based path testing technique & time taken to generate test cases & reduction in number of generated test cases [10].

Sanjay Singhla, et.al (2011) discussed about a hybrid PSO approach to automate test data generation for data flow coverage with dominance concepts. In this paper they uses the two techniques that is based on a combination of genetic algorithm (GA) & particle swarm optimization (PSO) & thus is called GPSCA (Genetic-Particle Swarm Combined Algorithm) to automatic test data generation for searching test data & it checks the condition whether the coverage of data flow using the fitness function or not because it evaluates the fitness of test data based on its relation through dominance to the definition & use in the data flow requirement. It is effective technique of test data generation for achieving the coverage & reducing the number of test cases by using the empirical formula. Which gives the result that it performs better than GA & PSO in number of generation. & it is successful in achieving def use coverage & it analyse that the number of test cases generated in case of GPSCA is less than GA & PSO [11]

Laheeb M.Alzubaidy & Baraa S. Allhafid (2013) proposed the software testing using hybrid technique intelligent water drop & ant colony optimization. Intelligent water drop algorithm use the white box testing technique for generating the basis path testing & for the test generating the test data use the ant colony optimization algorithm. In this parsing tool is used which converts the input program to the corresponding control flow graph for the program & with the help of technique intelligent water drop algorithm it generate the independent path & test data in order to test all the independent paths in the program it covered all the path efficiently & it find the optimal solution or path. Ant colony optimization generate the best test data. Which gives the result to cover all the independent path efficiently test the data [12]

Praveen Ranjan srivastava & Km Baby (2010) focussed on automated software testing using metaheuristic technique by applying ant colony optimization for generation of optimal of optimal & minimal test sequences for behaviour specification of software & gives the comparison between two metaheuristic techniques (genetic algorithm & ant colony optimization) for transition based testing. In this it describe the standard method of state transition based testing & its coverage level. By using the ACO technique it finds the optimal test sequence for the state transition based software testing & it provides the full coverage. This technique enhance the quality of tool as it limits the repeated number of transition in the test sequence [13]

Hitesh Tahbildar & Bichitra Kalita (2011) in this paper it describes the different implementation techniques with their advantages & disadvantages for the automated software test data generation. It gives the brief description about architecture for the test data generation. Architecture I & architecture II requires path selector phase. Architecture III & IV avoid the path selector phase. Constraint & generator are required for architecture I. Test data generator is required for the architecture II. Architecture III finds the all unique feasible path & avoid the constraint solver. Object oriented programming is useful for Architecture IV. Different program analyzer are used based on test information requirements & the method of test data generation. It satisfy functional, non-functional & business requirements. With the help of automation functional requirement testing is done [14]

Kevillienuo Kire & Neha Malhotra (2014) proposed software testing using intelligent technique. Ant colony optimization is one of the software testing technique. It reduce the test suite of regression testing by using the Ant colony optimization. It select the test cases from test suite which will reduce the test efforts, cost & time of regression testing. It find the best optimal solution. This optimization algorithm gives the better & effective fault revealing. It gives details how to generate the path after each iteration & eliminate the redundant test cases. It also generate the pheromone table. It select & prioritize the test case by selecting the best path with minimum execution time[15]

Basem O.Alijlja et.al (2014) proposed a modified intelligent water drops algorithm & its application to optimization problems. For IWD algorithm two ranking based selection methods have been proposed. In this it uses the three combinatorial optimization problems i.e.RSFS, MKP & TSP. The most important parameter in LRS-IWD & ERS-IWD is to determine the optimal setting by doing the systematic analysis. In selection method the gradient of mapping function is control by the selective process. It gives the result that IWD is used to solve the complex real world optimization problems based on the three optimization problems , ERS-IWD performs more effectively then LRS-IWD as well as the current FPS method. The computational requirement of ERS-IWD is high from 1.15 to 13 times higher than those of FPS-IWD [16].

S.Rao Rayapudi (2011) proposed the intelligent water drop algorithm for solving the economic load dispatch problem. The main objective of this paper is to minimize the total cost of generation. It is easy to implement & capable of finding feasible near global optimal solution with less computational effort. Various generator constraint has been applied. It tested the 6-units & 20-units for solving the ELD problem. It handle the constraint effectively. This is one of the methods which determine the most efficient, low cost & reliable operation of a power system by dispatching available electricity generation resources to supply load on the system. It gives the numerical results. The result of this algorithm is compare with the genetic algorithm, particle swarm optimization & bee colony optimization. This algorithm is robust & fast convergence as compare with the others. The convergence characteristic of 6 unit it is seen that solution is converged to near optimal solution after 48<sup>th</sup> iteration & the convergence characteristic of 20 unit is seen that the solution is converged to near optimal solution after 90<sup>th</sup> iteration [17].

Matthew Schmid & Frank Hill (2011) discuss about the techniques for automated software robustness testing. In this it gives the two techniques of generating data for the testing of automated software robustness. In this paper analyze the effectiveness of two techniques & explore their usefulness in automated software robustness testing. The techniques are generic data & intelligent data. Generic data is not dependent on the software component being tested & intelligent data it is tailored specifically to the component under test. In the experiment it perform the robustness testing of windows NT software components. The components which are tested is command line utilities & win 32 API functions. Using generic & intelligent testing techniques both types of components are tested & it gives the result that intelligent testing uncovers robustness failures that could never be discovered using only generic testing. By using both the technique generic testing proves the valuable result in both the experiment & it is inexpensive & still this is one of the useful testing techniques [18].

**III. PROBLEM IDENTIFICATION**

A challenge for IT industry is to develop software system that meets business needs. If the bugs or error are not removed from the software that can cause major loss in IT organization. So the important parameter for developing software that makes free from bugs & defects are software testing. Software testing is performed to support quality assurance. Software testing is an important & valuable part of the software development life cycle. Quality of test data is one of the issue for automated test data. One of the major time consuming activities in a software project is the construction of test data. The problems of test data generation are

- Arrays & Pointers
- Objects
- Loops
- Modules
- Infeasible path
- Constraint satisfaction
- Oracle

**IV. PROPOSED METHODOLOGY**

The main objective of this research work is to formulate various techniques for testing the test data generation. In the proposed work two well known algorithm namely intelligent water drop & ant colony optimization are implemented over the automated software test data generation

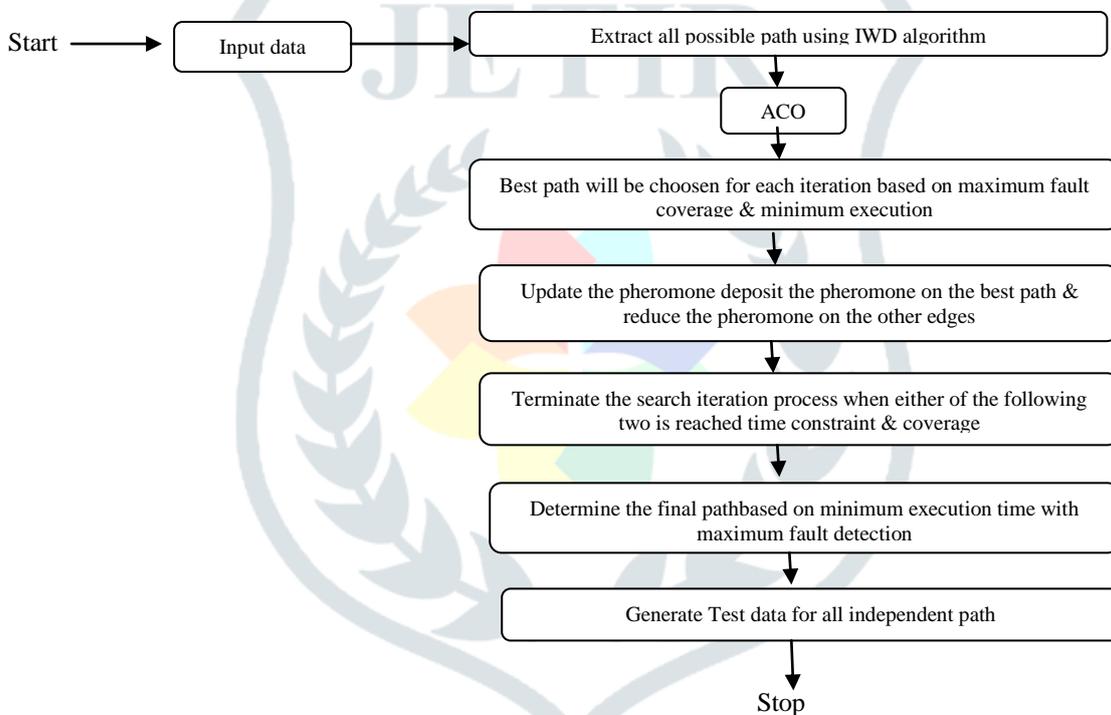


Fig 1 flow of proposed methodology

**Steps for proposed methodology**

**Step 1** Start

**Step 2** Input data (Static or dynamic parameter)

**Step 3** Extract all possible path using Intelligent water drop algorithm

$$P(i, j) = \frac{path(i)}{path(j)} + \frac{soil(i, j)}{Init - soil} - \sum \frac{soil(i, k)}{Init - soil} / \text{No. of } k$$

where

path (i) = no. of path from node I to be the end node

$\sum soil(i, k)$  = sum of the soil of every path i to k ,where i is not equal to k

**Step 4** Applying Ant Colony Optimization

**Step 5** Best path will be chosen based on maximum fault coverage & minimum execution time

$$P_{ij}^k = \frac{(T_{ij}^\alpha)(\eta_{ij}^\beta)}{\sum_{j \in i} (T_{ij}^\alpha)(\eta_{ij}^\beta)}$$

Where

$p_{(ij)}$  = pheromone deposited from node i to j

$\alpha$  = parameter to control the influence

$\eta_{ij}^\beta$  = desirability of state transition i j & it also represent the other possible state

Transition

**Step 6** Update the pheromone deposit the pheromone on the best path & reduce the pheromone on the other edges. When all the ants have completed a solution the trails are updated

$$T_{(ij)} = (1-\rho)T_{ij} + \sum T_{ij}^k$$

Where

$T_{ij}$  = amount of pheromone deposited for a state transition i,j

$\rho$  = pheromone evaporation coefficient

$\sum T_{ij}^k$  = amount of pheromone deposited by the k ant

$\sum T_{ij}^k = Q / L_k$  ( if ant k is use curve i j in its tour)

$L_k$  = cost of the k ants tours

Q = constant

**Step 7** Terminate the search iteration

**Step 8** Generate test data path for all independent path while iterating step 2

**Step 9** Data will be generated in different format while identifying the path

**Step 10** End

## V. INTELLIGENT WATER DROP ALGORITHM

**Input :** Problem data set

**Output :** An optimal solution

**Step 1** Initialization of static parameters i.e. parameter are not changed during the Search process

**Step 2** While algorithm termination the condition is not met do

**Step 3** Initialization the dynamic parameters i.e. parameter changed during the Search process

**Step 4** Spread IWDs randomly on the nodes of the graph

**Step 5** Update the list of visited node to include the source node

**Step 6** For IWD residing in current node it select the next node

**Step 7** For each IWD moving from one node to another node it update the velocity

**Step 8** Compute the soil

**Step 9** Update the soil

**Step 10** Solution found by the IWDs it find the best iteration solution

**Step 11** Update the soils on the paths that form the current iteration best solution

**Step 12** Update the total best solution by the current iteration best solution

**Step 13** Increment the iteration number

**Step 14** Stop with the total best solution

## VI. ALGORITHM FOR ANT COLONY OPTIMIZATION

Step 1 Set parameters, initialize the pheromone trails

Step 2 While (termination condition not met) do

Step 3 Construct ants solutions

Step 4 Apply local search

Step 5 Update pheromones

Step 6 End while

## VII. RESULT & DISCUSSION

Every individual or an enterprise requires that its maintenance cost always remain low and security remains higher, so that they could get gain best results. Automated test data generation is a best choice for testing data and accessing without extra effort. As technologies grow flaws in the area of automated software test data generations. So, always newer and better systems are constructed to achieve best testing techniques. This proposed integrating scheme intends to give best result in generating different types of data,

- The methodology proposed in the previous chapter is implemented and the simulation result of the implemented algorithms is shown in this chapter. The algorithms are coded in .net and the simulation of these algorithms over automated software test data generation is performed through N unit testing Tool. The simulation is carried step by step by generating different format of data. The performance is analyzed over one computational factor that is data type. Below fig shows the generation of data in different format by using the integrated techniques. Which accepts the different data type like string, character, float, integer which is major issue in the field of automated software testing. But using this technique we overcome this issues

PROGRAM 1. Min Max Problem

```

1  0  #include<stdio.h>
2  0  int main()
3  0  {
4  1  int a, b, min, max;
5  1  printf("Enter the two number");
6  1  scanf("%d%d",&a,&b);
7  2  if(a<b)
8  3  {
9  3  min=a;
10 3  max=b;
11 3  }
12 4  else
13 5  {
14 5  min =b;
15 5  max=a;
16 5  }
17 6  printf("minimum=%d, maximum=%d", min, max);
18 6  return 0;
19 6  }
    
```

Control Flow Graph Of Min Max Program

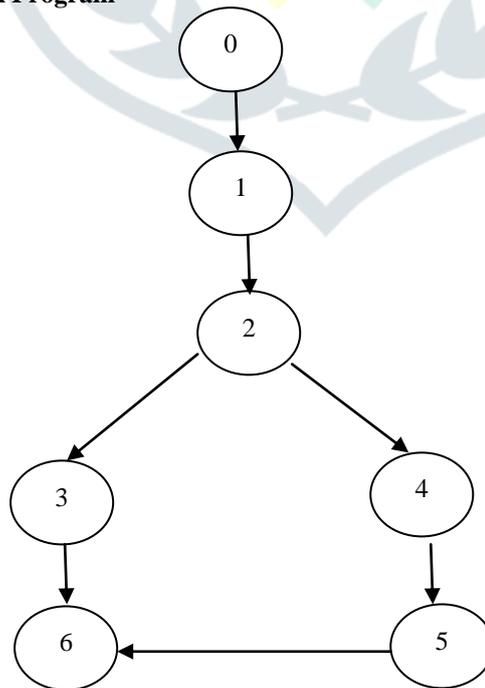


Fig 2 control flow graph for the min max program

**Computational Use :-**

$$C\_use = 'min' \quad 'max'$$

**Predicate Use :-**

$$P\_use = 'a' \quad 'b'$$

**Table 1. Experimental result of min max problem for path exploration :-**

Path	Probability	Chosen Path
(0,1)	0.2	(0,1)
(1,2)	0.1	
(2,3,4)	0.3	(3,6)
(3,6)	0.4	
(4,5)	1	(4,5)
(5,6)	0.12	
(6,0)	2	(6,0)

Path 1 : 0, 1, 2, 3, 6

Path 2 : 0, 1, 2, 4, 5, 6

Node : 0, 1, 2, 3, 4, 5, 6

Cyclomatic Complexity : 1, 1, 1, 1, 1, 2

**Table 2 Experimental result for min max program :-**

Path Traversed	Condition	variable			
		a	b	min	max
0, 1, 2, 3, 6	a<b	4	7	a=4	b=7
0, 1, 2, 4, 5, 6	a>b	9	2	b=2	a=9

**Test Data Generated**

Data = 28      80  
          47      46  
          26      87  
          125     54  
          50      131  
          229     232  
          26      161  
          12      26  
          143     100  
          197     14

## REFERENCES

- [1] Shaveta Rani, Neha sethi & Paramjeet singh, "A review on test case selection & priotization in swarm intelligence",International Journal of Emerging Technology & Advanced Engineering,2014.
- [2] Vishal D Rampure, "Automated software testing", Advanced software engineering CSC532
- [3] Navdeep koundal, Ankur Shukla & Mohisn Rashid Mir, "Test case selection using bee colony optimization", International journal of science & research,may 2014
- [4] Er.Rajendra kumar & Dr.M.K gupta,"Dynamic test case design scenario & analysis of module testing using manual vs automated technique",International journal of computer science & network,2012.
- [5] Yeresime Suresh & Santanu Ku Rath, "A genetic algorithm based approach for test data generation basis path testing," International journal of soft computing & software engineering,2013.
- [6] Sapna Varshney & Monica Mehrortra "Automatic software test data generation for data flow dependenciesusing genetic algorithm," International journal of advanced research in computer science & software engineering,2014.
- [7] Praveen Ranjan srivastava, Amit kumar patel et.al, "Test data generation based on test path discovery using intelligent water drop," International journal of applied metaheuristic computing, 2012.
- [8] Bashar Aldeeb, Norita Md Norwawi, Mohammed A. Ai-Betar,"A survey on intelligence water drop algorithm", International journal of computers & technology,2014.
- [9] Jon Edvardsson,"A survey on automatic test data generation",linkopping university,Sweden,2013.
- [10] Ruchika Malhotra & Mohit garg,"An adequacy based test data generation technique using genetic algorithms",journal of information processing system,2012.
- [11] Sanjay Singhla, Dharminder Kumar, HM Rai & Priti single,"A hybrid approach to automate test data generation for data flow coverage with dominance concepts",International journal of advanced science & technology,2011
- [12] Laheeb M.Alzubaidy & Baraa S. Alhafid,"Proposed software testing using intelligent techniques(Intelligent water drop(IWD) & Ant colony optimization(ACO))",International journal of computer science, 2013.
- [13] Praveen Ranjan Srisvatava & Km Baby,"Automated software testing using metahurestic technique based on ant colony optimization",Electronic system design(ISED),2010.
- [14] Hitesh tahbildar & Bichitra Kalita,"Automated software test data generation:Direction of research",International journal of computer science & engineering survey, 2011.
- [15] Kevilienuo Kire & Neha Malhotra,"Software testing using intelligent technique", International journal of computer applications,2014.
- [16] S.Rao Rayapudi,"An intelligent water drop algorithm for solving economic load dispatch problem",World academy of science engineering technology,2011.
- [17] Mathew Schmid & Frank hill,"Data generation techniques for automated software robustness testing ",Reliable software technologies corporation,2012.
- [18] C Doungsa-ard, K Dahal & A hossain,"AI based framework for automatic test data generation",IEEE,2012.
- [19] Sahil Batra & Dr. Rahul Rishi,"Improving quality using testing strategies".journal of global research, 2011.
- [20] Phil McMinn,"Search based software test data generation:A survey",university of Sheffield,2004.
- [21] Gordon fraser, Andrea Arcuri & Phil McMinn,"Test suite generation with memetic algorithm",GECCO,2013.
- [22] Getha Devasena M.S & Valarmathi M.L,"Optimized test suite generation using tabu search technique", International journal of computational intelligence techniques,2010.