ABSTRACT

Test Effort Estimation is an important activity in software development. The test effort can be calculated on the basis of effort cost and time required for testing. Several studies have been done for developing test effort estimation models but to some extent only, most of these models result in erroneous results. So there is a strong need to optimize the efforts estimated. Meta heuristic techniques can be used for this purpose, to optimize a problem by iteratively trying to improve a solution, using some computational methods. In this paper, we have implemented meta-heuristic based search algorithm namely PSO. The particle swarm optimization algorithm is used for improving testing effort estimation. The particle swarm optimization algorithm (PSO) is applied on use case point (UCP) and results led us to the conclusion that test effort estimation can be optimized by applying PSO. The PSO optimization can also be applied for estimating efforts of software development. This implementation increases the accuracy of testing effort estimation.

Keywords

Software testing, particle swarm optimization (PSO) and use case point (UCP).

1. INTRODUCTION

Testing is the most important quality assurance measure of software. Software testing is an important and complex issue of software development process. Software testing is defined in the different activities of the test process. Testing is a process rather than activity. It involved the series of the activities. The defects find out without executing the code is static testing and the code is executed for manifest or pre conversion factor and then convert testing project size into person hours of effort using Delphi Technique or Analogy Based Estimation technique.

We can also derive software testing project size and effort using Delphi Technique or Analogy Based Estimation technique.

2. RELATED WORK

James Kennedy et al.,[3]introduces a method for optimization of continuous nonlinear functions. The method was discovered through simulation of a simplified social model; thus the social metaphor is discussed, though the algorithm stands without metaphorical support. This paper describes the particle swarm optimization concept in terms of its precursors, briefly reviewing the stages of its development from social simulation to optimizer. Discussed next are a few paradigms
that implement the concept. S. Aloka et al. has presents a Particle Swarm Optimization (PSO) algorithm. It was applied on two techniques: use case points (UCP) and test point analysis (TPA) and the results led us to the conclusion that test effort estimation can be optimized by applying PSO [1]. The results were compared with those obtained from existing methods, and were found to be closer to the actual effort. Suresh Nageswaran et al. [2] developed a new approach to the estimation of software testing efforts based on Use Case Points [UCP] as a fundamental project estimation measure. The V model must be in use and use of case generation must start becoming available right at the requirements gathering phase. The author conjectures that this could become a more robust method of estimation over a period of time. The availability of data from past projects will definitely contribute to the accuracy of these estimates. The estimation technique is not claimed to be rigorous, but the approach offers significant practical advantages over ad hoc techniques currently in use.

3. PROPOSED WORK
Software testing forms an integral part of software development process and is said to consume maximum effort of the whole process, which is nearly 50% of the total effort. Test Effort Estimation is an important activity in software development because estimating the effort beforehand enables project managers to allocate resources i.e. budget, time and staff efficiently and avoid future inconvenience. Many effort estimation techniques are available resulting into satisfactory estimates. However, with increasingly tight schedules and market competition, more accurate estimates are needed. Use Case Point analysis one such technique which has proven to be successful over time in estimating testing effort based on the number of use cases. However, there are a number of parameters which are utilized in the analysis but their values have to be set by the tester on their previous experience. This increases the dependency of the estimation accuracy on human and can vary widely as per the tester’s capabilities. This encourages the use of optimization techniques to be utilized to tune the parameters of the UCP analysis. In this project, a heuristic technique called Particle Swarm Optimization (PSO) has been utilized to do the same and its performance has been tested over a well-known data and shown to perform exceedingly well in terms of both accuracy and convergence rate.

4. METHODOLOGY

Fig 1: Flowchart of Methodology

5. WORK DONE
All experiments are performed in the MATLAB framework. This framework is used for the PSO algorithm implementation. In this experiment, main aim is to tune up the parameters of PSO algorithm to improve the test effort estimation. We have taken 10 particles, with 16 parameter values corresponding to actors (3 for actor weights), use cases (4 for use cases), and technical factors (9 for technical factors) in Use Case Point Analysis in the search space randomly. The range of values for the parameters was bound in the intervals shown in Table 4.1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor</td>
<td>1-3</td>
</tr>
</tbody>
</table>
We have experimented on the 10 particles, with 16 parameters of the UCP analysis. Initially, effort of all 10 particles is calculated by taking the effort equation as the objective function. This will be their initial local bests, and from these local bests, the best one is taken to be the initial global best. The 16 parameters are actor, use cases and technical factor. In each iteration, these parameters for each particle are updated using the position and velocity equations for PSO. And then, objective function is again computed to find the local best and global best. Thus in each iterations, local and global best get updated, and the swarm moves towards optimality. In this problem 100 iterations are considered because results converge to an optimal solution within this limit. More iteration may be required, depending on the problem. The actual effort is 390 man days.

6. CONCLUSION AND FUTURE WORK
In this paper, we have implemented meta-heuristic based search algorithm namely PSO. The particle swarm optimization algorithm is used for improving testing effort estimation. The particle swarm optimization algorithm (PSO) is applied on use case point (UCP) and results led us to the conclusion that test effort estimation can be optimized by applying PSO. The PSO optimization can also be applied for estimating effort of software development. This implementation increases the accuracy of testing effort estimation. In future, we will implement the other algorithms of meta-heuristic technique such as ant colony optimization, firefly algorithm etc. and generate a hybrid algorithm for increase the accuracy of test effort estimation.

7. REFERENCES


