Journal of Advanced Computing and Communication Technologies *(ISSN: 2347 - 2804)*
Volume No.4 Issue No. 3, June 2016

A Survey of Software Development using optimization techniques

By

Harjeet Singh[1], Gurmeet Kaur[2]
Lecturer Thapar Polytechnical collage, Patiala,India
harjeetsinghbiviaan@gmail.com[1], ergurmeetkaur92@gmail.com[2]

ABSTRACT

Software is becoming an increasingly critical-competitive factor because of the high cost of developing software and its extra-ordinarily high busi-ness impact. Although there are challenges in software development including accelerated and bug free software delivery which are correlated with business requirements. Now a days, new process models are introduced for different requirements. In this paper, we can discuss software development based optimization techniques.

General Terms

Your general terms must be any term which can be used for general classification of the submitted material such as Pattern Recognition, Security, Algorithms et. al.

Keywords

ASD, Software Reuse, CBD, ADF, Optimization, PSO,ACO.

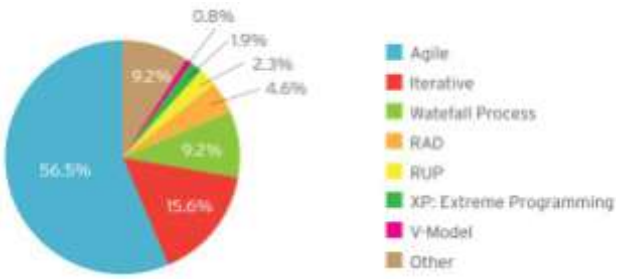
1. INTRODUCTION

The software services industry has undergone various changes during the past decade. Software development is the process for testing and fixing the bug to create applications and frameworks which can involved in a software release life cycle. Software development may include research, prototyping, modification, reuse, re-engineering, maintenance, or any other activities that results in a software product. There are lots of methodolo-gies used for developing the software. Generally used methodologies for software development are Reuse Based Development, Component Based De-velopment, Agile Software Development. All these models are different and are not able to satisfies the requirement of software industry. This paper deals with the optimization of Software Development Schedule (SDS). An optimization strategy such as minimizing or maximizing consists to optimize

[1]Harjeet Singh with Department of Computer Science, Thapar Polytechnic College, Patiala, Punjab, INDIA harjeetsinghbiviaan@gmail.com

[2]Gurmeet Kaur with Department of Computer Science, Patiala, Punjab, INDIA ergurmeetkaur92@gmail.com

The given objective function under the set of multiple constraints. In this study, the objective function is to be minimized or maximized the duration of the concurrent projects, while the constraints are resources with limited precedence and capacities relations among the tasks. Therefore, meta-heuristic techniques such as PSO and ACO to solve the Software Development Project Scheduling



Problem. PSO is a population based stochastic optimization technique developed by Eberhart and Kennedy [7][8]. The algorithm is inspired by the social behavior of bird flock-ing. The authors later extended PSO to solve the discrete optimization problems. On the other hand, Ant colony optimization (ACO) is a prob-abilistic algorithm which can evolved from the social behavior of ants. ACO was introduced and developed by Dorigo [4].

2. SOFTWARE DEVELOPMENT

A. Reuse Based Development

Many software organizations realize that de-veloping the software using components which are reusable can linearly reduce the development effort, cost, time consume and accelerate the de-livery. But the non-existence of a standard

process model in this field contributed towards in the large-scale failures in their approach. In literature, it is found that very few attempts had been made

to identify or measure the software reuse process level, technical and organizational factors neces-sary to imply reuse success [6]. Even though some studies had tried to formulate the reference model for reuse, because of the novelty of the approach they adopted, but could not achieve success in their attempt [10][5].

1. Types of Reusability:

Application System Reuse The entire appli-cation system may be reused either by in-corporating it without changes other systems (COTS reuse) or by developing application families.

Component Reuse Components of an appli-cation from the sub-systems to single objects may be reused.

Object and Function Reuse Software com-ponents which can implement a single well-defined object or function may be reused.

2. Reusable Assets of Software Development:

Architecture
Source Code
Data
Designs
Documentation
Templates
Human Interfaces
Plans
Requirements
Test Cases

3. COMPONENT BASED DEVELOPMENT

Component based development idea was or-ganized in a conference name ”Mass Produced Software Components” in 1968. Since, then en-gineering and scientific libraries are developed to re-use the previously developed functions. This concept is now widely used in Software devel-opment as component based development (CBD). Component-based software engineering (CBSE) is used to develop/assemble software from existing components [2].

4. AGILE SOFTWARE DEVELOPMENT

Agile software is different from other method-ologies and also for different projects. ASD is a group of software development methods which are based on iterative and incremental develop-ment, where requirements and solutions evaluate through the collaboration between self-organizing and customers, cross-functional teams. It enhanced adaptive planning, evolutionary based development and delivery, a time-boxed iterative approach, and encourages rapid change in software which is very flexible. It is a conceptual framework that promotes foreseen interactions throughout the development cycle. It is very difficult to change software prod-uct internally because of invisible thus software development projects are highly unpredictable and not able to meet the expectations of customers and organization [1].

1) Concepts of ASD:
Reduce Waste Successful Pace
Moderate Collaboration Fast Delivery

Continuous Feedback Empowerment

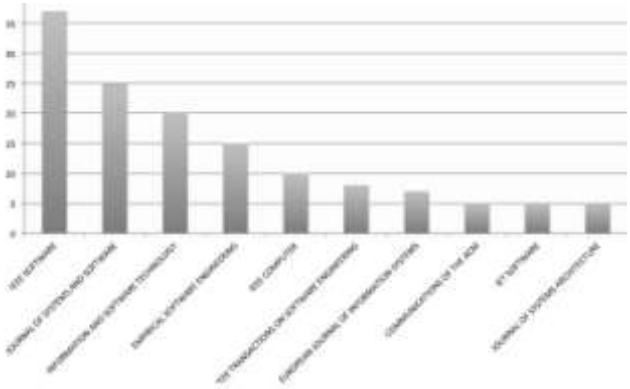


Fig. 2.Number of papers in scientific journals of ASD [3]

5. ANT COLONY OPTIMIZATION

An ant is a single individual which has very lim-ited effectiveness and of a well-organized colony, it becomes one powerful agent which can works for the development of whole colony. Ant colony optimization (ACO) is a probabilistic algorithm that has evolved from the social behavior of ants. ACO was introduced and developed by Dorigo [4]. ACO is based on the fact, that ants have the capabilities of searching for the shortest path from their entire nest to the source of food that is regarded as the best path. The searching of shortest path is being done via chemical substance called pheromone which provides the framework to communicate between ants.

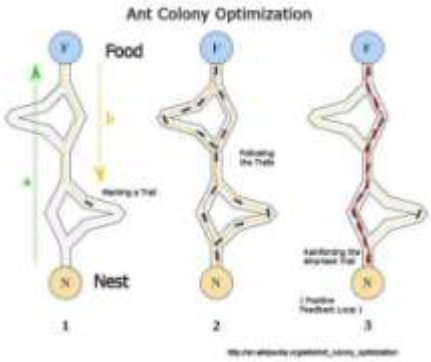


Fig. 3. Ant Colony Optimization (ACO)

5.1 ACO-based model

The below Figure represents the proposed ACO-SDPSP approach in flowchart mentioning the basic work-flow of solution to the software development problem. This approach was proposed by Suri and Jajoria [11]

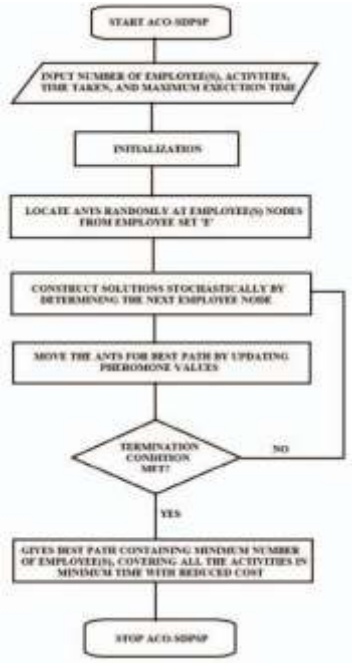


Fig. 4. Ant Colony Optimization (ACO)

6. PARTICLE SWARM OPTIMIZATION

PSO is a population based stochastic optimiza-tion technique developed by Eberhart and Kennedy[7][8]. The algorithm is inspired by the social be-havior of bird flocking. The authors later extended PSO to solve the discrete optimization problems. PSO is a meta-heuristic technique which makes few or no assumptions about the problem decision which can optimized and can search on very large spaces of candidate solutions. However, meta-heuristics such as PSO may or may not guarantee an optimal solution is ever found.

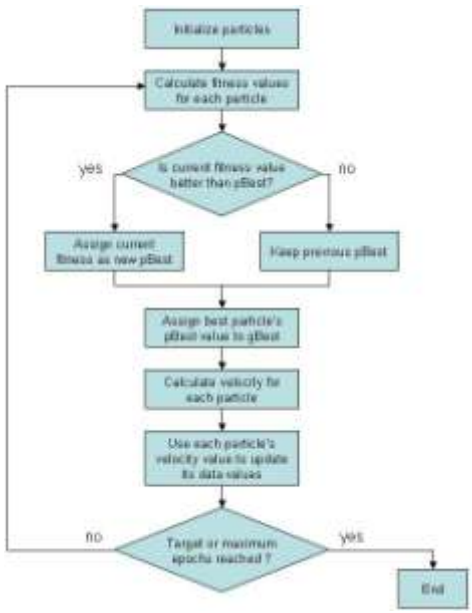


Fig. 5. Flowchart of Particle Swarm Optimization (PSO)

6.1 PSO-based model

This paper aims to overcome the optimize problem of software development by suggesting the appropriate weights for project attributes. To explicitly determine the research goal Bardsiri [9] describes the three research questions which are defined as follows:

1. How the PSO algorithm can be combined with software process to design an efficient attribute-weighting system?
2. How do the different structures of PSO based model affect the accuracy of the weighting system?
3. Can a PSO algorithm increase the accuracy of estimates achieved by other estimation methods?

7. CONCLUSION

This paper suggests the various optimization models for the software construction. Each opti-mization model has some pros and cons. We have discussed various aspects of the software re-use implementation optimization model, but it also has many limitations like it is not technically justified and the second is that the task regions are selected based on subjective measurements given by project leaders. Similarly many software systems devel-oped with Agile Software Development are often poorly designed, but still perform a good job for the organizations critical application. Due to poor design and documentation it becomes very difficult to reuse the functionality of legacy system in the future projects. The importance of the software de-velopment cannot be undetermined because some of their functions are too valuable to be discarded and too expensive to reproduce. Therefore software development with some optimization techniques may or may not guarantee optimal solutions but make sure of better results as compared to other tools and techniques because one problem can run with different simulation run.

REFERENCES

[1] Y.-L. Chen. Analysis of the agile deployment. rapport nr.: Report/DEpartment of Applied Information Technology 2009: 073, 2010.

[2] I. Crnkovic. Building Reliable Component-Based Software Systems. Artech House, Inc., Norwood, MA, USA, 2002.

[3] T. Dingsøyr, S. Nerur, V. Balijepally, and N. B. Moe. A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, 85(6):1213– 1221, 2012.

[4] M. Dorigo and T. Stutzle¨. Ant Colony Optimization. Bradford Company, Scituate, MA, USA, 2004.

[5] W. B. Frakes and C. J. Fox. Quality improvement using a software reuse failure modes model. IEEE Transactions on Software Engineering, 22(4):274, 1996.

[6] K. Jasmine and D. R. Vasantha. A new process model for reuse based software development approach. In

Proceedings of the World Congress on Engineering, volume 1, pages 2–4, 2008.

[7] J. Kennedy and R. Eberhart. Particle swarm optimization. u: Proc. ieee int. Neural Networks, Perth, Australia, str, 1948, 1942.

[8] J. Kennedy, J. F. Kennedy, R. C. Eberhart, and Y. Shi. Swarm intelligence. Morgan Kaufmann, 2001.

[9] V. Khatibi Bardsiri, D. N. A. Jawawi, S. Z. M. Hashim, and E. Khatibi. A pso-based model to increase the accuracy of software development effort estimation. Software Quality Journal, 21(3):501–526, 2013.

[10] M. Morisio, M. Ezran, and C. Tully. Success and failure factors in software reuse. Software Engineering, IEEE Trans-actions on, 28(4):340–357, 2002.

[11] B. Suri and P. Jajoria. Using ant colony optimization in software development project scheduling. In Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, pages 2101–2106, Aug 2013.