Advances in Computer Science and Information Technology (ACSIT)
Print ISSN : 2393-9907; Online ISSN : 2393-9915; Volume 2, Number 7; April – June, 2015 pp 24 – 26
Krishi Sanskriti Publications
http://www.krishisanskriti.org/ACSIT.html

Effort Estimation of the Scrum based Software Projects using Particle Swarm Optimization

Navil Akhtar[1], ShabinaGhafir[2] and Saurabh Tripathi[3]

[1,2,3]Dept. Of Computer Science, Jamia Hamdard University
E-mail: [1]nabeel11286@gmail.com, [2]sghafir@jamiahamdard.ac.in,
[3]saurabh.it.1987@gmail.com

Abstract—*Effort Estimation is a critical activity in any software Engineering project. Accuracy of the effort and schedule estimation heavily determines the success of failure of the projects. . Software industry has emphasized more on Web Application Projects due to their inherent benefits. Web Application can be used globally by the different user around the globe and they just need an authorized access to the application and the internet connection. Lot of estimation techniques are suggested by many researchers keeping in mind the different software development methodologies. Agile Software development methodologies are gaining popularity in the software industry which can easily accommodate the requirement changes in the form of User Stories and client doesn't need to wait for long time to use the software. Researchers have used soft computing techniques to accurately estimate the effort for software development. The paper examined the effort estimation of different releases for the web based application developed using Scrum methodology and how it can be integrated with the PSO algorithm to improve the results. Team efficiency index is a crucial part of this paper which can determine the actual effort of each release. Generally, the adequate target value for Mean magnitude of relative error (MMRE) is 25%. It shows that the magnitude of relative error (MRE) for each project for the established estimation model should be less than 25% on the average. A software development effort estimation method with a smaller MMRE value than the one with bigger MMRE value gives better estimates than a model with a bigger MMRE value [1].*

Keywords: *effort estimation, Scrum, PSO (Particle Swarm Optimization), User Stories, Sprints*

1. INTRODUCTION

Estimation is the approximate calculation of the results which is usable even if input data may be uncertain. Software effort estimation is the process of predicting the effort required to develop a software system. As cost is directly proportional to the effort, estimating the cost of a software product is one of the most difficult and error–prone tasks in software engineering which thus results into budget overruns/underruns. Different algorithmic approaches were given to find the effort estimation of the software product which follows the traditional software development life cycle methodologies like waterfall etc. Due to increasing rate of changes in the requirements and need for quick releases of the software leads to adopt a new development methodologies i.e. Agile. In Agile, there aredifferent techniques like Scrum, Extreme Programming, CANBAN, etc. and each technique have its own features and rules to follow. In scrum, there are user stories as product backlogs which are developed according to their defined priority. In this paper, we will introduce how the effort estimation is carried out previously using User stories and Team efficiency index which can determine the estimated effort. Each determinant in calculating the effort can be used as dimension to the Particle Swarm optimization algorithm. Before going forward some terms need to be understood which are given below:

Scrum

Project management framework and latest flavor of Agile- a software development methodology. Scrum is a simple yet extremely powerful set of principles and practices that help teams deliver products in short cycles, enabling fast response, continual enhancement, and speedy adaptation to change. Scrum can be used by anyone who has complex project whether developing a mobile app, new e-commerce website or maintaining the old projects etc. The main objective of Scrum is to deliver the highest priority product backlogs (User Stories) first in every sprint (Releases). Team plays a very important role due to their expertise, experience and stable size [6].

User Stories

It captures the 'who', 'what‘, 'why' and 'when' of a requirement in a simple and concise way.

Who	-	Developer with customer representative
What	-	Description of requirement
Why	-	Functionality
When	-	Throughout the project

Format

As a <type of user>, I want <some goal> so that <some reason>[2].

Example

As a <Administrator>, I want a <Login Screen > so that <Every user can be authenticated before using application >.

Sprints

Sprint is a release or an iteration of 30 days normally. However in many projects teams prefer shorter sprints, such as one-week or two-week sprints. Each sprint contains the tasks which are very high in priority and needs to be delivered first. Every sprint begins with the sprint planning meeting then Product backlog meetings and Daily standup meetings.

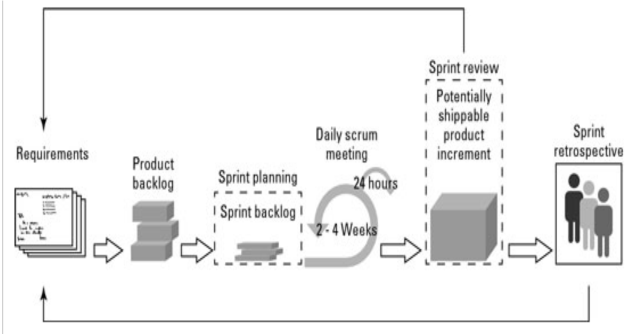


Fig. 1

2. PREVIOUS WORK

In traditional model a team member's workload bandwidth is analyzed by the manager who estimates how much effort particular tasks will take and then assigns work based on that team member's total available bandwidth. Agile methodology considers a different approach to determine a team member's bandwidth. Some of the known needs to be understood first are given below [2]:

- For every task, team is responsible not an individual.
- Scrum team estimates the effort in terms of size and complexity not the managers with the help planning poker.
- Standard scale values are used to define the size and complexity to estimate the effort for which every team member is comfortable.

Team plays a very important role as there are several key differences in Scrum team and traditional team where every decision is taken by the managers.

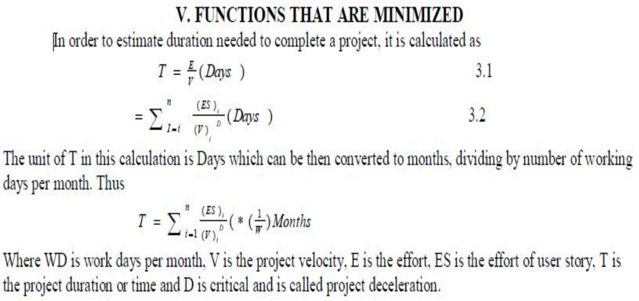
1) Agile teams are "whole teams" [2] i.e. team includes each and every member of different skills to develop, test, deliver and maintain the software product.
2) Team consists of highly skilled, experienced and specialized members.
3) Agile teams are stable in size.

There are multiple factors involved in accurately estimating the effort. Accurate estimation requires a multidimensional view to produce accurate and effective estimates. Two important factors which determine the effort are user story size and complexity which defines the overall LOC or functionality measure and difficulty and dependency that impacts the software respectively. Some of the guideline for deciding the user story size and complexity were given to scale them in terms of values between 1 to 5 moving towards to highly complex and big in size. Guidelines can be changes according to the comfort of the team. To estimate the effort from the user story size and complexity:

$$Effort = (\sum_{k=1}^{n} Size_k * Complexity_k)$$

Where n ε number of tasks or product backlog.

Particle swarm optimization (PSO) is a method that computes the optimized solution to a problem by iteratively trying to enhance a candidate solution with regard to a given measure of quality. PSO is similar to GA but there are no mutation and crossover operator are used. PSO learned from the situations and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. We call it "particle". Fitness values of all particles are evaluated by the fitness function to be optimized, and have velocities which control the flying of the particles. The particles fly through the problem space by following the current best particles [3]. To implement the effort estimation, the objective function is needed to use it in the PSO algorithm implementation [1]



V. FUNCTIONS THAT ARE MINIMIZED

In order to estimate duration needed to complete a project, it is calculated as

$$T = \frac{E}{V}(Days) \qquad 3.1$$

$$= \sum_{I-i}^{n} \frac{(ES)_i}{(V)_i^{D}}(Days) \qquad 3.2$$

The unit of T in this calculation is Days which can be then converted to months, dividing by number of working days per month. Thus

$$T = \sum_{i-1}^{n} \frac{(ES)_i}{(V)_i^{D}}(*(\frac{1}{W})Months$$

Where WD is work days per month, V is the project velocity, E is the effort, ES is the effort of user story, T is the project duration or time and D is critical and is called project deceleration.

3. PROPOSED MODEL

In Scrum, Team plays an important role so introducing the new factor i.e. Team Efficiency Index. Team Efficiency index is a measure which defines the efficiency of the team on basis of the previous work done by the team. It can be calculated by finding the efficiency of each resource in the team.

Resource Efficiency

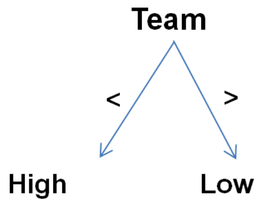
$$RE = (\sum_{k=1}^{n} AE_k / EE_k) / n$$

Where n ε number of tasks or product backlog, AE is Actual Effort and EE is estimated effort for all tasks done by the resource.

Team Efficiency

$$TE = (\sum_{k=1}^{m} RE_k) / m$$

Where m ε Team Size



To calculate the effort Team Efficiency index, equation would be:

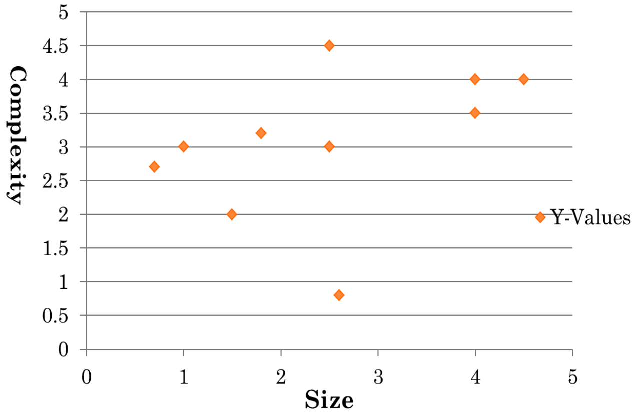
$$Effort_{new} = (\sum_{k=1}^{n} Size_k * Complexity_k) * TE$$

To implement the Particle swarm optimization, Size and complexity will be the input dimensions for the algorithm function. Effort will be considered as same as the distance.

To find the Velocity

$$Velocity = Effort_{new} / SS$$

Where SS is Sprint Size normally 1 months i.e. 22 days in a month.



4. CONCLUSION

With the increasing need of winning the software projects bids, level of complexity and delivering the product within the expected time, cost and quality has drawn much attention to the need for flawless effort estimation techniques. With the help of soft computing techniques, effort estimation can be taken to a new level. Team Efficiency is not the only factor influencing the effort estimation so with more investigation and research, we can find other factors too.

REFERENCES

[1] Manga I & Blamah NV, "A particle Swarm Optimization-based Framework for Agile Software Effort Estimation",(2014) Available: www.theijes.com/papers/v3-i6/Version-5/D0365030036.pdf

[2] Ziauddin, Shahid Kamal Tipu, Shahrukh Zia,"http://worldsciencepublisher.org/journals/index.php/ACSA/article/view/703/566",(2012) Available: arxiv.org/pdf/1310.5221

[3] Sumeet Kaur Sehra, Yadwinder Singh Brar, and Navdeep Kaur, "SOFT COMPUTING TECHNIQUES FOR SOFTWARE PROJECT EFFORT ESTIMATION",(2011) Available: arxiv.org/pdf/1310.5221

[4] Ratnesh Litoriya, Abhay Kothari, "An Efficient Approach for Agile Web Based Project Estimation: AgileMOW",(2013) Available: www.scirp.org/journal/PaperDownload.aspx?paperID=33252

[5] Rosmina, Suharjito, "FHSWebEE: An Effort Estimation Model for WebApplication",(2012) Available: http://www.researchgate.net/publication/250003478_FHSWebEE_An_Effort_Estimation_Model_for_Web_Application

[6] Scrum Alliance, "Learn About Scrum",(2012) Available: https://www.scrumalliance.org/why-scrum

[7] Xiaohui Hu, "Particle Swarm Optimization",(2012) Available: http://www.swarmintelligence.org/

[8] Open Works, "Effort Estimation", Available: http://open-works.org/blogs/effort-estimation-software-development

[9] Wikipedia, "Software development effort estimation",Available: http://en.wikipedia.org/wiki/Software_development_effort_estimation

[10] Wikipedia, "Particle swarm optimization",Available: http://en.wikipedia.org/wiki/Particle_swarm_optimization