IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 36-39
www.iosrjournals.org

Implementation of Game theory in Software Testing

Dr. Bharati Wukkadada[1] Akshay Pendbhaje[2]
[1](Assistant Professor, K.J. Somaiya Institute of Management Studies and Research, Vidyavihar [E], India.)
[2](Student (MCA), K.J. Somaiya Institute of Management Studies and Research, Vidyavihar [E], India.)

Abstract: *We know errors in software development can occur at any stage of software development lifecycle. Later the error is identified more the correction cost is, however if error is made in earlier phase and detected early the cost of correction is less and if not solved early error is propagated in the later stages of lifecycle. These is based on an assumption that all the participants in the software development process are willing to make the software correctly at all the possible levels and not only their individual level. It is not possible for any individual to completely ignore personal goals. This results into conflict between team members of same software development team. In this paper we will analyze software development process as noncooperative game and try to understand hidden problems within it with help of game theory.*

Keywords: *Software Testing, Game theory, non co operative game, software development lifecycle.*

I. Introduction

A. Game theory

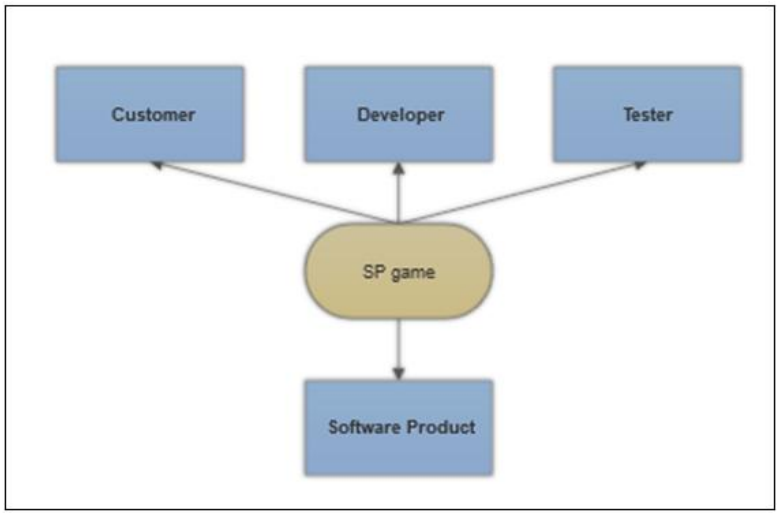
Game theory is about mathematical models where rational decision makers sometimes cooperate with each other and sometimes conflict with each other. It is widely used in many fields including economics, political science, logic, computer science. It is used in psychology too. Initially when introduced it was only about zero sum games. Zero sum games are games where one person wins and other loses. When there is gain for first participant there is loss for second. However today that is not the scenario, game theory is applied to many behaviors. Also the term is used for logical decision making in humans, computers, etc. [2]

B. Software Testing

Software testing is a process which is carried out to give knowledge about the quality of the software product to the stakeholder. It provides information about chances of failure of the software. To what extent it might be wrong [3].

Even though testing cannot detect all the possible errors in the software as exhaustive testing is not possible it can check for error in major functionalities. Earlier in time software testing was done to show how correct the software is according to the requirements. Now methodology is changed and now software testing process is carried out to show presence of defects

II. Model



Dig 1. Model of SP game

As we can see in the model there are three members involved which are customer, developer and the tester in software development project game (SP game). This SP game we will see in detail in this paper.

Huge number of errors occurs in software development project because of poor understanding at the time of different phases throughout the lifecycle. These errors can occur at the time of requirement gathering, while creating or choosing architecture, selecting wrong development model this causes a huge amount in repair and maintenance job [5][8][12]. We can say these are the major places where errors occur in software development lifecycle assuming that all the people involved in the software development project are willing to make software successful and all of the people involved agree on a single goal and method of how to achieve that goal. However, each personnel have own goals and personal targets. These goals lead to conflict in team members in the software development team which affects the success factor of the project. In this paper we will analyse software development process as non-cooperative game and try to understand hidden problems within it with help of game theory.

III. Goal Of The Software Development Project

There are some measure goals of software development project which are to create a successful software product and conduct a successful software maintenance process [4][5]. It is assumed that every participant of the project attempt to achieve these two goals. However, there can be differences in opinion and the goal may differ from one and other, this can also be because of some external situations.

There are different analytical models in software economics which relates programmer's productivity with the cost or the budget of the project, to help in making decision of pricing strategy, or to evaluate economic risks a project may have and how it might impact the project in different ways [4][5][6][7]. Here too the goal of every member of the software development project is assumed to be having same goal as that of general gaol of software development project [1]. This is not completely true because every individual has three characteristics which are rationality, mutual independence with other members, individualism [1]. Rationality means every individual works on their own self-interest. Mutual independence is important to avoid future problems in team. Individualism means member of organisation can choose between entering into a team or not. This might be because individual need to do different actions for individual goals. It is possible that a single person starts a company takes software project and work on that project, we do not consider such cases here.

When these individual goals of the participants of project clashes with the general goals of the project it might lead to failure. Such causes of failure are difficult to identify. In this paper we will try to understand and analyse such problems occurring due to such clashes.

IV. Approach

We assume software development lifecycle is collection of activities which has constraint of members in the project work to achieve their own goals and not what project demands. The process of software project is the game here. We will call it SP game. All the members of the game are players on the basis of game theory model [10].

Each player in this SP game is a rational decision maker, which means that all the players in the game make decision to achieve their own goal. The player has different choices of actions. The chosen actions of a player are denoted by A. The consequences of the action are denoted by C. since this game involves interactive decision making it is a strategic game. These choices are made simultaneously by different players. Every player in the game decides his own strategy which comprises of collection of different actions. The actions are chosen from A.

a. PLAYERS

We consider three players in the SP game. First player of the game is customer, second player is developer and third player is tester. We will denote them with letters C, D, T respectively. Customers are the people who will get the outcome of this SP game, which is the developed software. It can be an individual or a company whoever requested the software. Developers are the group of people who develop the project. Developer has the work from requirement gathering to building the architecture, designing the system to coding it. Developer also implements the system. All the players denoted by D are directly related to the software development process. Testers are players who are also involved in the software development from the beginning since the requirements are gathered. Testers are responsible for testing the software right from the requirement gathering phase until deployment phase. Testers are all the members who deal with testing in all the phases and do quality assurance. They do validation and verification in software development project. After the requirements are gathered they make sure right requirements are gathered and when development is in different phases like design architecture and coding, they validate if the design is according to requirements or not, if the modules are working properly or not, if the integration of modules with each other and with the system is

correctly done. After the system is completely ready how it is performing in terms of speed, reliability, and load. The job of tester goes hand in hand with developer.

b. STRATEGIES

Strategy of C is to reduce the price of the software to zero, to get the product as fast as possible and to have maximum features in the same price of the software. However not concerned about the cost of the software.

Strategy of D is to create successful outcome of the SP game. Along with this D has to be updated with the trend and technologies so that his value in the market remains high. D do not want to lose his existing job at the same time maintain good relation with C so that C will recommend D to more people in the future. D wants the cost of the project to be high so that he gets more incentives and at the same time doesn't want to add more features. D Wants minimum features to be developed in maximum incentives. D wants fastest development regardless of quality.

Strategy of T is to get right and quality product from the SP game. T is concerned with the quality of the outcome of the SP game. The product should be matching to the requirements and of good quality in terms of functional and nonfunctional requirements. The product can be developed in longer time but should be of best quality.

c. PAY OFF

As we already saw the strategy of three players of SP game we can see how strategy preferred by each player in the game. This is also called as payoff.

For payoff of C a measurement can be how the software is working according to the requirement. How much helpful the software product is for the business and how much it is helping to make or increase profits.

For payoff of D we can say to seek a promotion or secure a job can be the preference. However this totally depends on the outcome of the SP game.

For payoff of T we can say to make the outcome of the SP game better. To find out all the possible errors and mistakes in all the work of player D is one of the major goals of player T.

In the following table we can see the payoff matrix. The columns denote the players of the SP game i.e. C, D and T. there are different variables which have effects on the outcome of the SP game.

Here we consider four of these variables. Cost, Speed, Quality and Features. Cost represents the money which was spent on the SP game. It is not equal to price, price is at what amount the outcome was sold to C. Speed is how fast the product was developed. We can say time taken for the particular SP game. Quality is how much is C satisfied with the outcome of the SP game.

Player/Variable	C	D	T
Cost	-1	1	1
Speed	1	1	-1
Quality	1	0	1
Features	1	-1	0
Total	2	1	1

Table 1. Payoff relation of players and variables in SP game

In the above table the values are -1, 0, and 1. When a player is interested to increase a value of some variable it is denoted by 1. When a player is not bothered about the value about the variable it is denoted by 0 and when the player wants to reduce the value of the variable it is denoted by -1.

V. Analyzing The Game

According to game theory a game is non-cooperative if it satisfies two conditions, if there is no possibility of sharing the payoff. And all the players are making decisions for themselves. As we can see to share payoff D have to either sacrifice on cost i.e. less incentives for his work. This is not acceptable as D has put in lot of efforts in SP game and also sacrificed own extra time whenever crashing of project was required. Or else D will have to sacrifice speed of development which means D cannot work faster to finish one project to acquire new one.

VII. Finding The Equlibrium

In game theory we know that when we try to maximize the payoff of one player or two it will affect payoff of other players in the game. Here in our case if we increase payoff of player D it will affect payoff of player T and C. same way if we try to increase payoff of player T it will affect payoff of D and if we maximize the payoff of player C it will affect payoff of player D and player T.

Now we will try to identify the winning equilibrium. A winning equilibrium can be achieved by balancing the payoffs. We will see how to achieve the winning equilibrium with the help of game theory. One way to reach this is new relation between players of the game which might be based on new agreements and understanding between players. In the SP game this new relation can be in between player D and player T based on how much correctness is required in the outcome of the SP game. It can be among different or all players too.

In order to achieve the balance in equilibrium there should be balance in payoff. The payoff can be made in such a way that whenever a successful SP game outcome comes, all the players win in some way. This is easy to say but difficult to achieve scenario. All the credit of Successful outcome of SP game is not given to D but the blame is always given to D.

VIII. Conclusion

In this paper we would like to achieve the maximum balance between players which will in turn result into higher possibilities of getting a successful outcome from the SP game, players should cooperate with each other in following way. Player C should not try to lower the cost beyond the value of amount of efforts put into the SP game by player D and player T together. Player T should stop testing certain scenarios after certain successful test cases, which will lead to reduction in total development time of the SP game. Player D should try to keep the quality of work up to the mark as stated and accepted in requirement documentation. This will help player D to maintain own image in market. Player C should not expect more features than decided in the software requirement documentation and should be more realistic. When all the players are working for the success of the SP game along with own goals it will have more chances of successful outcome.

References

[1]. *Myerson, Roger B. (1991).* Game Theory: Analysis of Conflict, *Harvard University Press, p.*
[2]. website- https://en.wikipedia.org/wiki/Game_theory (6/11/2017)
[3]. *Kaner, Cem (November 17, 2006). "Exploratory Testing". Florida Institute of Technology, Quality Assurance Institute Worldwide Annual Software Testing Conference, Orlando, FL*. Retrieved November 22, 2014.
[4]. B. Boehm, *Software Engineering Economics*, Prentice Hall, Upper Saddle River, NJ, 1984.
[5]. B. Boehm, "Software Risk Management: Principles and Practices", IEEE Software, Jan. 1991, p.32-41.
[6]. E. DeGarmo, W. Sullivan, and J. Bontadelli, *Engineering Economy*, Prentice-Hall, Upper Saddle River, NJ, 1993.
[7]. E. Grant, W. Ireson, and R. Leavenworth, *Principles of Engineering Economy*, Wiley, New York, 1990.
[8]. F. Brooks, *The Mythical Man-Month*, Addison-Wesley, 2nd edition, August 1995.
[9]. L. Levy, *Taming the Tiger - Software Engineering and Software Economics*, Springer-Verlag, New York, 1987
[10]. M. Osborne and A. Rubinstein, *A Course in Game Theory*, MIT Press, August 1994.
[11]. Private conversations with IBM, Schlumberger, and KLA-Tencor employees.
[12]. S. Flowers, Software Failure: Management Failure: Amazing Stories and Cautionary Tales, John Wiley & Sons, December 1996.