iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** VI **Month of publication:** June 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Behavior of Software Reliability Based On Game Theory

Sumant Bansal[1], Rachna[2]
M.tech Scholar[2] , Assistant Professor[2]
[1,2]Department of Computer Science & Engineering, GIET Sonepat, Haryana, India.

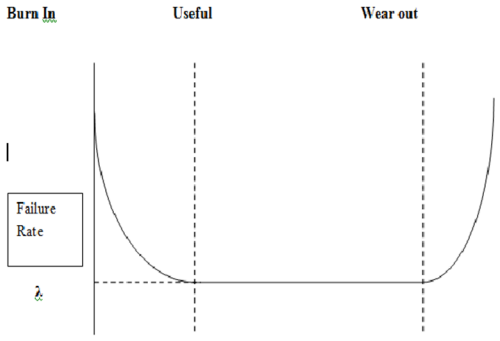
Abstract-- Software Management includes the study and analysis of the software development in an organize way so that the transparency of the whole system will be improved. Some of the common software management approaches like CMM and SPICE gives the quantitative model of software process management. But these methods are based on the accuracy and the reliability of the input data to the system. If the data is impure or some distorted or damaged data is there the quantitative results driven from these models are also not reliable. Therefore it is required to analyze the reliability of the metrics under different factors that can affect the reliability factor. The presented work is about to estimate the reliability vector of this data for different metrics of software process model under different criteria. In the existing work, a game theory based work is defined to analyze the quality or the reliability of the collected data. The presented work is also in the same direction to estimate the data or the metrics reliability. We will define the better view of the reliability factors. The existing system does not include the type of application or the software as the major criteria. Because the efforts or the processing approach followed by a developer is surely inspired from the type or the criticality of the application. To perform this, we will generate a new survey based on new questionnaire set and identify the developer perspective under the heading of software type and criticality. The improved question set will give the more clearly and application specific view of the developer. Finally the game theory will be implemented to get the conclusion regarding the analysis. The result analysis will be performed in the form of graphs.
Keywords-- software management, game theory, software reliability, questionnaire, MATLAB, Java.

I. INTRODUCTION

Software Reliability is the probability of failure free software which work for a specified period of time in a specified environment. Software Reliability is also an important factor affecting system reliability. Software reliability is different from hardware reliability because it defines the design perfection, rather than manufacturing perfection. The high complexity of software is the major contributing factor of Software Reliability problems. There are two types of reliability

A. Hardware Reliability

It is the ability of hardware to perform its functions for some period of time without any failure. It is expressed as MTBF which means mean time between failures. The hardware reliability is described on the basis of bath tub curve.

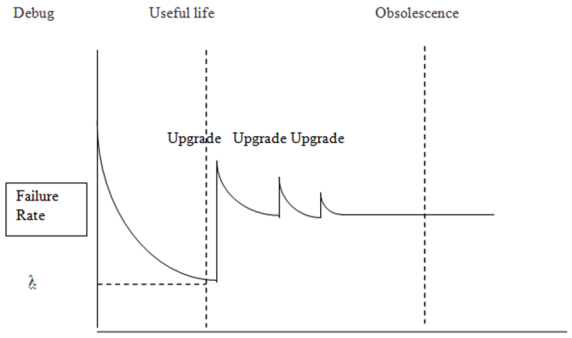


The above diagram describes the burn in phase, the useful phase that is which is very beneficial and wear out phase means that when the software is on the last stage.

©IJRASET 2015: All Rights are Reserved

B. Software Reliability

Software Reliability is the probability of failure free software which work for a specified period of time in a specified environment. Software Reliability is also an important factor affecting system reliability. Software reliability is different from hardware reliability because it defines the design perfection, rather than manufacturing perfection. Below diagram describes the software reliability.



In the above diagram the upgrades defines the features upgrades and not upgrades for reliability. When we apply the feature upgrade the complexity of software is l increased and the functionality of software is enhanced. The bug fixes sometimes lead to more software failures .Sometimes when we try to fix some problems or defects they can induce more defects in the system.

The main objective of the software reliability is to define a question hair with improved reliability vectors that will be passed to the industry level to obtain the view of different developers. So that the effectiveness of the input data to software process model will be quantify. Identify the critical and the non critical factors to perform the analysis. Define a game theory based on system to relate & identify the effect of the software process model metrics and the reliability vector under the developer's perspective. Simulate the system to identify the reliability of the data and present the results in graph form. There are number of software process model used to analyze the software system. But the reliability of these models is based on the reliability of the collected data. The presented work is about to perform the analysis of reliability of the input data so that reliability of the process model will be analyzed and improved. The presented work is about to analyze the reliability of the metrics collection parameters so that the improvement will be performed in the critical areas. The presented system will improve the reliability of the data as well as the system.

II. RESEARCH METHODOLOGY

To analyze a software system, some software process models are present. These process models basically analyze the system based on certain metrics. The accuracy and the reliability of these system based on the input values pass to the system. The reliability of the data depends on the source of the data collection. In this work, reliability of such data metrics is been defined under the developer perspective. A questionnaire will be defined to analyze the developer contribution and the commitment to provide effective input to the system. At the initial stage an improved survey based system will be defined under the heading of software type and the criticality. Based on the vector different weightage will be assigned to different parameters. Once the dataset is obtained, a game theory based analysis will be performed to identify the criticality vectors and their impact on the process model. The presented system will perform the effective analysis so that the reliability vector will be defined with the data collection so that secure reliability measures will be conducted.

A. Sources Of Data

All the knowledge required for the requirement analysis can be studied by the following approaches.

Study the available literature on Requirement Analysis. Most of the available resources of requirement analysis are used by some of the available projects and the research work.

A questionnaire approach can be performed against the different kind of stakeholders and performing a requirement acquisition. A case study can be performed on study the already running projects and approaches in the company.

B. Research Design

The first stage of the work is to perform a survey so that the quantitative analysis on the factors that affect the reliability can be analyzed and relative effective decision will be drawn from the system. In this complete work we have studied different approaches

©IJRASET 2015: All Rights are Reserved

of requirement gathering by using

Survey

Interview

Schedule

C. Game Theory

Game theory can be defined as a study of mathematical models about conflict and cooperation between rational decision-makers. Game theory using mathematical technique was designed to analyze situations when more than 2 parties should make a decision which will have an effect on their own interest under various possibilities of cooperation or conflict. This theory was first introduced in 1944 by a physicist and mathematician, Jon von Neumann and economist Oskar Morgenstern. A game consists of player (at least 2 players), player's strategy, outcome of decision, and payoff, numerical value of the outcome. Outcome is dictated by who selects which strategy. Therefore, each competitor should select a strategy which can maximize his or her own interest (performance) no matter what strategy other competitor(s) will employ.

D. Tool

Java was designed to meet all the real world requirements with its key features, which are explained in the following paragraph.

SIMPLE AND POWERFUL
SECURE
PORTABLE

MALAB

MATLAB is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and FORTRAN.

Introduction and Key Features
Developing Algorithms and Applications
Analyzing and Accessing Data
Visualizing Data
Performing Numeric Computation
Publishing Results and Deploying Applications
MATLAB provides a number of features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

III. RESULT AND ANALYSIS

Here, we present the survey results obtained from different personnel (professionals, practitioners) of different software organizations.

Defect Density Per Module	95.1	4.9	0
Defect Management Ratio	92.7	4.9	2.4
Defect Discovery Ratio	63.4	36.6	0
Requirement Change Ratio	78.0	22.0	0
Requirement Implementation ratio	56.1	44	0
Coding Standard	75.6	24.4	0
Process Improvement Impact	63.4	36.6	0
Manpower effect	82.9	14.6	2.4
Defect correction Time	39.0	56.1	4.9

©IJRASET 2015: All Rights are Reserved

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Review Time	56.1	43.9	0
Delivery Ratio	75.6	24.4	0
Each person Productivity	75.6	24.4	0
Reuse Ratio	65.9	31.7	2.4
Size effect	82.9	14.6	2.4
Module reuses ratio	85.7	11.9	2.4

Table 1. Survey Results

The analysis of collected data through the survey results is very important and difficult task. We have collected a lot of data during the survey, which has a direct or indirect relationship with our research area. We have analyzed the collected data from the survey and extract the most relevant data to our research questions. The results obtained from the survey questionnaire regarding use of Defect Reports for improving Software Processes are presented in above section. Furthermore, this section presents the analysis of results of survey questionnaire according to the flow of research questions. Data shown in table 2 in the form of 1-5 Rating. Rating 1 is given to Values between 1-20; Rating 2 is given to values between 21 to 40 and so on.

Defect Density Per Module	5
Defect Management Ratio	5
Defect Discovery Ratio	4
Requirement Change Ratio	4
Requirement Implementation ratio	3
Coding Standard	4
Process Improvement Impact	4
Manpower effect	5
Defect correction Time	2
Review Time	3
Delivery Ratio	4
Each person Productivity	4
Reuse Ratio	4
Size effect	5
Module reuses ratio	5

Table 2. Factor Rating

The result analysis will be performed in the form of graphs. This experimental result shows the software reliability under the different factor of software reliability.

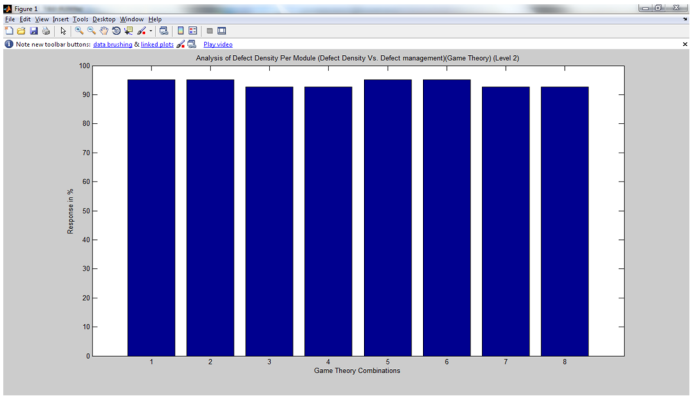


Figure 1. Analysis of defect density per module

©IJRASET 2015: All Rights are Reserved

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

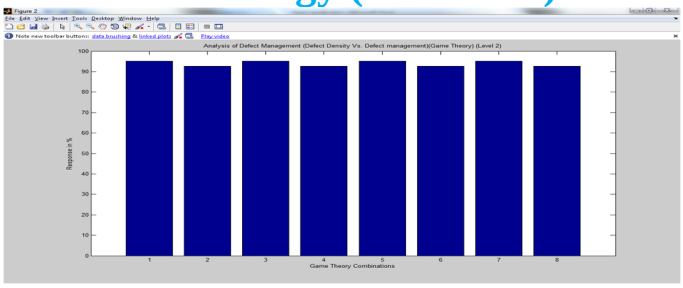


Figure 2. Analysis defect management ratio

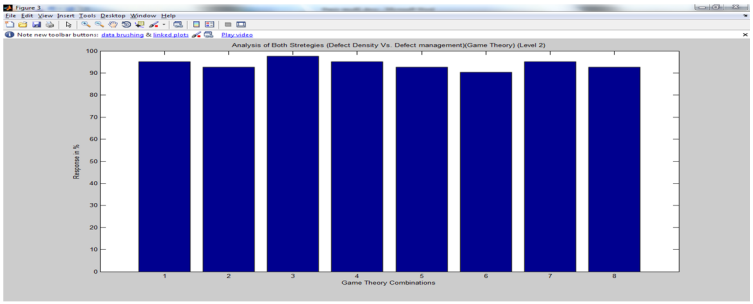


Figure 3. Analysis of both strategies

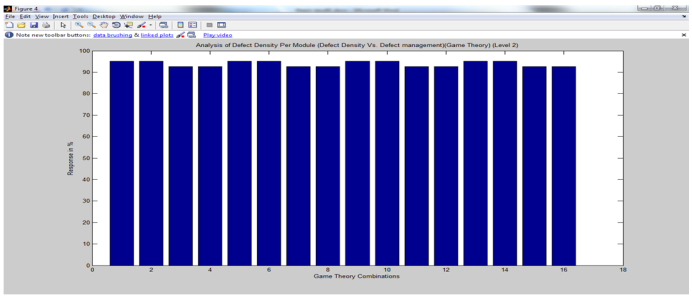


Figure 4. Analysis of defect density per module (defect density vs. defect management)

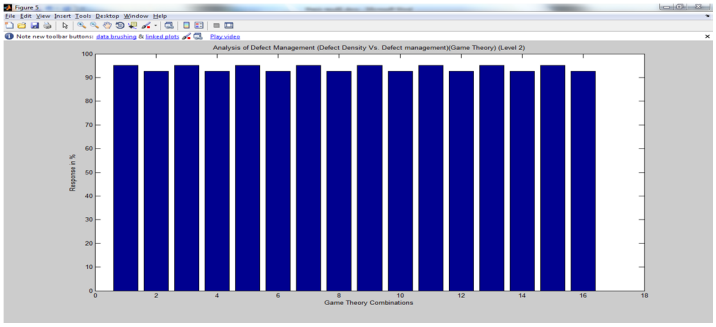


Figure 5. Analysis of defect management (density vs. defect management)

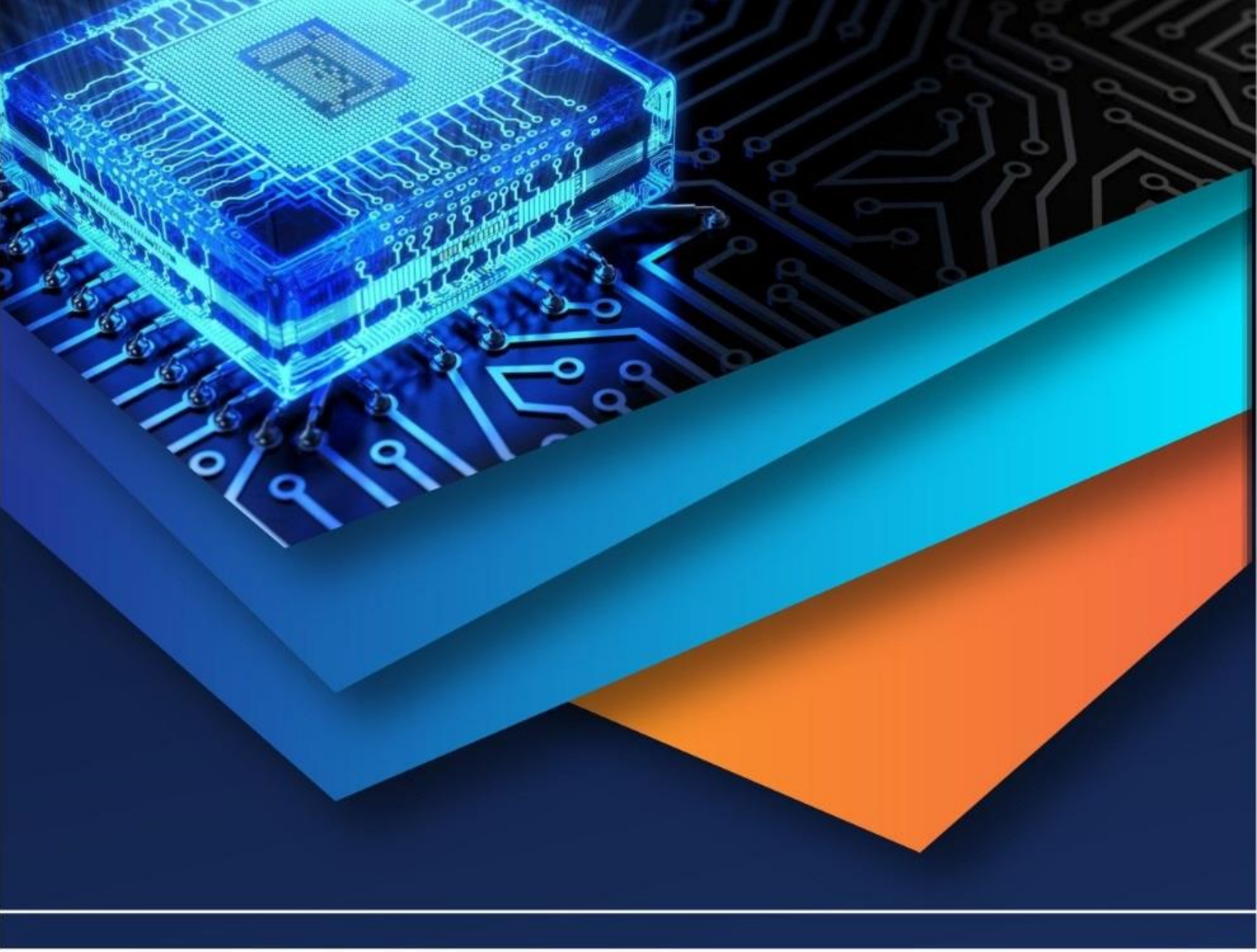
Here figure 1, 2, 3, 4, 5 is showing the between game theory combination and the response time. Here x axis represent game theory combination and y axis represents the ratio. These graphs show that the reliability factor effect reliability of the software in the higher value shows higher effect.

©IJRASET 2015: All Rights are Reserved

REFERENCES

[1] Joshua Reich," VMTorrent: Scalable P2P Virtual Machine Streaming", CoNEXT'12, December 10–13, 2012, Nice, France. ACM 978-1-4503-1775-7/12/12 (pp 289-300)
[2] Toshiya Fujii," Towards Quantitative Software Reliability Assessment in Incremental Development Processes", ICSE'21–28, 2011, Waikiki, Honolulu, HI, USA ACM 978-1-4503-0445-0/11/05 (pp 41-50)
[3] Franz Brosch," Reliability Prediction for Fault-Tolerant Software Architectures", QoSA+ISARCS'11, June 20–24, 2011, Boulder, Colorado, USA. ACM 978-1-4503-0724-6/11/06 (pp 75-84)
[4] I Mandal," Software Reliability Assessment using Artificial Neural Network", International Conference and Workshop on Emerging Trends in Technology (ICWET 2010) – TCET, Mumbai, India ICWET'10, February 26–27, 2010, Mumbai, Maharashtra, India. ACM 978-1-60558-812-4 (pp 698-699)
[5] Sang-Pok Ko," Secure Reliability of Measurement Data through Application of Mechanism Design Theory".
[6] Aleksandar Dimov," How do we collect data for software reliability estimation?", International Conference on Computer Systems and Technologies - CompSysTech'10 CompSysTech'10, June 17–18, 2010, Sofia, Bulgaria. ACM 978-1-4503-0243-2/10/06 (pp 155-160)
[7] Aleksandar Dimov," How do we collect data for software reliability estimation?", International Conference on Computer Systems and Technologies - CompSysTech'10 CompSysTech'10, June 17–18, 2010, Sofia, Bulgaria. ACM 978-1-4503-0243-2/10/06 (pp 155-160)
[8] Pankaj Jalote," Measuring Reliability of Software Products", (pp 1-8)
[9] Zibin Zheng," Collaborative Reliability Prediction of Service-Oriented Systems", ICSE '10, May 2-8 2010, Cape Town, South Africa ACM 978-1-60558-719-6/10/05 (pp 35-44)
[10] Nasser H. Abosaq,"Improvement of Software Reliability Modeling Predictions by the Detection and Removal of Test Outliers", ACMSE '09 March 19-21, 2009, Clemson, SC, USA.
[11] Zhanshan (Sam) Ma," Introduction of First Passage Time (FPT) Analysis for Software Reliability and Network Security", The Fifth CSIIRW '2009, April 13-15, Oak Ridge National Lab, Oak Ridge, Tennessee, USA. ACM 978-1-60558-518-5
[12] hao-Jung Hsu," Reliability Analysis Using Weighted Combinational Models for Web-based Software", WWW 2009, April 20–24, 2009, Madrid, Spain. ACM 978-1-60558-487-4/09/04. (pp 1131-1132)
[13] Audris Mockus," Empirical Estimates of Software Availability of Deployed Systems",ISESE'06, September 21–22, 2006, Rio de Janeiro, Brazil. ACM 1595932186/06/2009 (pp 222-231).

©IJRASET 2015: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)