International Institution for Technological Research and Development
Volume 1, Issue 1, 2015

A Survey on Refinement of Reliability by Improving Reliability Estimation Tool

Mohit Surati[1], Bhavesh Tanawala[2], P.B.Swadas[3],
Department of Computer Engineering,
B.V.M Engineering College,
Vallabh Vidhyanagar, Anand (Gujarat), India.
Research Scholar[1], AssistantProfessor[2] , AssociateProfessor[3]
mysurati@gmail.com [1], bhavesh.tanawala@bvmengineering.ac.in[2], pbswadas@bvmengineering.ac.in[3]

***Abstract*— Software reliability has become a buzzword for quite some time, still there were very few standardized tool had been developed to estimate software reliability in quantitative terms. Software reliability is such a critical factor in deciding the overall quality of software, thus it should be estimated. Here we have surveyed many existing software reliability tools like SMERFS (Software Modeling & Estimation of Reliability Functions for Software), SRMP (Statistical Modeling & Reliability Program), SoftRel, SoRel and specially focused on the surveying of CASRE. From the survey we have suggested that software reliability measurement can be improved by modifying the trend test, SRGM and filters in CASRE. This paper focuses on improving the existing CASRE tool so that the software reliability measurements obtained are much more accurate & efficient. This will help us to improve the reliability scenario in critical & real-time applications.**

Keywords: Software Reliability; CASRE; SRGM.

I. INTRODUCTION

People think that software is always correct and once it runs correctly it will be correct forever. But in contrast to that thoughts, series of classic tragedies like Therac 25 radiation therapy tragedy [1986], Gulf War tragedy [1991], Arine 5 missile launching tragedy taken place because of less reliable software technology proves this to be wrong [6]. This made the researchers wonder whether software is always reliable or not? This gave rise to the concept of Software Reliability.

According to ANSI, **"**Software Reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment" [6]. Therefore we can say that reliability is a critical factor in deciding the overall quality of software.

Software Reliability has been defined by the ANSI in the early 90's and since then it has been a hot concept for researchers in the area of software engineering. It motivated researchers to develop SRGM (Software Reliability Growth Model) which can be used to estimate the improvements in software during the development life cycle (SDLC). SRGM also helps to determine when to launch a product in order to achieve certain reliability goal.

Still there were very few standard for measuring software reliability in quantitative terms. The reliability requirement of any product varies according type of application and therefore it must be specified in SRS (Software Requirements Specification).

After the evolution of research and development in measurement of software reliability CASRE stands to become a relatively better tool to measure the software reliability which is intellectual property of NASA (National Aeronautics and Space Administration) [7]. Still there is a scope of improvement in CASRE by modifying the trend test, SRGM and filters.

Therefore researchers move towards the estimation of Software Reliability which then gave rise to the various software reliability estimation tools.

II. BACKGROUND THEORY & RELATED WORK

Researchers developed various tools for estimating "Software Reliability". These tools give the detail analysis of reliability in such a way that it can be further utilized to improve the reliability criteria for real-time applications.

Any Software Reliability Estimation Tool involves 5 main steps:

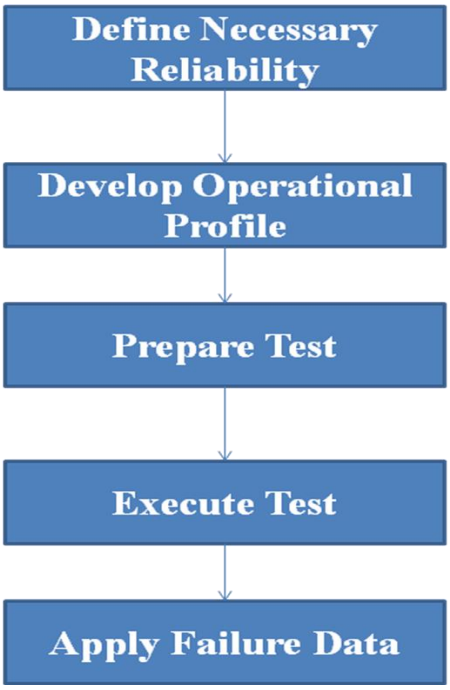

Figure 1: Steps Involved in Reliability Estimation Tool [7]

Few of the Software Reliability Estimation Tools are discussed below:

[1] **SMERFS:** SMERFS stands for **S**tatistical **M**odeling and **E**stimation of **R**eliability **F**unctions for **S**oftware.[7] It is developed in FORTAN language. The main advantage of this tool is the flexibility in data collection and multi-model debugging. The output of this tool is in SMERFS format and not ASCII format.

[2] **SRMP:** SRMP stands for **S**tatistical **M**odeling and **R**eliability **P**rogram [7]. It is a command line oriented tool developed for IBM PC/AT and UNIX based system. It requires ASCII data file as input & analyst here must be knowledgeable enough to set up input file.

[3] **SoftRel:** SoftRel is a console-based application written in C language [7]. It takes one input project file which is in formatted text and generates one output file which in CSV format. It is more of a simulation tool & the main disadvantage of this tool is that it only works for waterfall model based projects.

[4] **SoRel:** SoRel stands for **So**ftware **Re**liability Program [7]. It is a Macintosh-based tool developed by LAAS, a lab in France. SoRel is composed of two parts: trend test and software reliability growth models. It uses ASCII file as input. The disadvantage of this tool is that only one model can be executed at a time.

[5] **CASRE:** CASRE stands for **C**omputer-**A**ided **S**oftware **R**eliability **E**stimation **T**ool. It is a PC-based tool developed by Jet Propulsion Laboratories in 1993 & copyrighted by NASA.[7] The main advantage of this tool over the others is the ease-of-access to the users & the ability to combine various SRGM (**S**oftware **R**eliability **G**rowth **M**odel). Four model combinations are already available in CASRE tool & others can also be created according to the requirements.

Table 1: Specification of Various Software Reliability Estimation Tool

Tool	Specification
SMERFS	• Developed in FORTAN language • Flexibility in data collection and multi-model debugging.
SRMP	• Command line oriented tool. • Developed for IBM PC/AT and UNIX based system.
SoftRel	• Console based application. • Simulation tool. • Only works for waterfall model based projects.
SoRel	• Macintosh based tool. • Only one model can be executed at a time.
CASRE	• Developed by Jet Propulsion Laboratories. • Copyrighted by NASA. • Provides ease-of-access to users. • Ability to combine various SRGM.

III. LITERATURE SURVEY

After reviewing the papers & some web material related to the CASRE tool, we can analyze the current trends in software reliability which will eventually help us to improve the CASRE tool in order to achieve more accuracy in reliability measurements.

The authors Micheal R. Lyu and Allen Nikora in paper [1] presents CASRE tool (Computer-Aided Software Reliability Estimation Tool) which is fully automated and methodical application of software reliability modeling that can be applied for any real-time project. According to the research the author has put emphasis on the practical implementation of Software Reliability Growth Model (SRGM) rather than building a brand new one. This tool has feature of enhanced GUI (Graphical User Interface) which makes the dull process of software reliability estimation very easy for the user. It combines various linear SRGM for achieving more efficient result of reliability. Real-world application of this tool and its methods are showing encouraging results.

The author Chris F. Keremer in paper [2] has presented lack of results in CASRE tool due to various troubles in its evaluation process. Author has also proposed a series of solution by further research in this area to overcome the pitfalls in mangers' recent ability to deal with these issues. Author has also stated in this paper while the CASRE tool has gain the popularity among the users, many developers have embrace "Wait & See" attitude which contribute to absence of result for CASRE tool. However, in the problem there is an opportunity for the researchers to improve in the area of software metrics, models & gathering more real-time data to validate the results.

The author He Yan in paper [3] has proposed a new Software Reliability Growth Model (SRGM) which improves the Non-Homogenous Poisson Process (NHPP) assumptions based on fault debugging and incomplete fault debugging. Many previous researchers have proposed SRGM which do not consider the effect of imperfect debugging properly. But here the author has proposed a new SRGM in which total number of fault will change over a period of time. By adding this feature the author has made the model more realistic which accurately assess the software reliability.

The authors Joydip Dhar, Anamika, Seema Ingle and Yaminee Sheshker in paper [4] have presented a NHPP Software Reliability Growth Model (SRGM) based on logistic-exponential testing-effort in imperfect debugging environment. This paper also provides the insights about the release time of any software for reaching any particular level of reliability. Reliability is the most important aspect of software quality and every company wants its software to be error free. So in order to develop such software SRGM is very useful. Here the proposed SRGM by the author using the log-exponential Testing Effort Function (TEF) gives more realistic results for reliability and reduces the optimal release time because it considers imperfect debugging environment.

The authors Yan Luo, Torsten Bergander and A.Ben Hamza in paper [5] have presented a new weighted Laplace test statistics for Software Reliability Growth Modeling (SRGM) which raises the effectiveness and performance of the SRGM. The proposed statistics takes both activities in the system and proportion of reliability growth within the model into account. In general terms this approach is called weighted combination of growth & non-growth reliability models. The author has shown by the experimental results that the weighted Laplace test statistics gives far better performance than traditional approach for software reliability modeling.

IV. CONCLUSION

After reviewing all the research done and the web content related to CASRE and other Software reliability measurement tools, we can conclude that the existing CASRE tool have solved lots of issues caused by the earlier tools, but it still has room for improvement. More care should be taken while validating the tool, and there is a need of improved standardization in trend test, SRGM and filtering in order to achieve higher & better reliability estimates.

V. ACKNOWLEDGMENTS

It is a matter of great pleasure for me to get this opportunity of expressing my deep and sincere gratitude to those who has always helped me throughout my work. I heartily thank my guides Mr. Bhavesh A. Tanawala & Mr. P.B.Swadas for providing me their valuable guidance and constant support in my work. They guide me with their excellent knowledge in the area of "Software Engineering." I am also thankful to my friend Mr. Dhavalkumar H. Joshi for his constant cooperation, help and guidance throughout this journey. I would also like to thank all my colleagues who have helped me in some way or the other. I would like to thank my parents & family member for their love and support to build my moral during the work.

VI. REFERENCES

[1] Michel R. Lyu and Allen Nikora, "CASRE-A Computer Aided Software Reliability Estimation Tool", *IEEE Computer. Soc. Press [1992] Fifth International Workshop on Computer-Aided Software Engineering* - Montreal, Que., Canada..

[2] Chris. F Keremer, "An Agenda For Research In The Managerial Evaluation Of Computer Aided Software Reliability Estimation Tool", *IEEE Computer. Soc. Press [1989] Twenty-Second Annual Hawaii International Conference on System Sciences*, Volume II-Software Tr.

[3] HE Yan, "NHPP Software Reliability Growth Model Incorporating Fault Detection & Debugging", *IEEE 2013 IEEE 4th International Conference on Software Engineering and Service Science (ICSESS)* - Beijing, China (2013.05.23).

[4] Joydip Dhar, Anamika, Seema Ingle and Yaminee Sheshker, "Software Reliability Growth Model with Logistic-Exponential TEF in Imperfect Debugging Environment", *IEEE International Conference on Recent Advances and Innovations in Engineering, (*ICRAIE-2014), May 09-11, 2014, Jaipur, India.

[5] Yan Luo, Torsten Bergander and A.Ben Hamza., "Software Reliability Growth Modelling using a weighted Laplace Test Statistics", *IEEE 31st Annual International Computer Software and Applications Conference* - Vol. 2 - (COMPSAC 2007) - Beijing, China.

[6] Jiantao Pan, (1999 Spring). *Software Reliability*[Online]. Available:http://users.ece.cmu.edu/~koopman/des_s99/sw_reliability/

[7] B.H.Far, *Software Reliability & Quality*[Online]. Available:http://people.ucalgary.ca/~far/Lectures/SENG521/PDF/SENG521-12_handout.pdf

[8] W.H. Fan and O.D. Smith, "Statistical Modeling and Estimation of Reliability Functions for Software(SMERFS) User's Guide", TR 84-373, Revision 1, NSWC, December 1988.

[9] M.R. Lyu and A. Nikora, "Software Reliability Measurements Through Combination Models: Approaches, Results, and A Case Tool", in *Proceedings the Isth Annual International Computer Sofiware and Application Conference (COMPSAC'9I),* pp. 577-584, Tokyo, Japan, September 1991.

[10] C. Li, XG. Yue, and HW. Liu," A NHPP Software Reliability Growth Model Incorporating Test Coverage and Fault Removal Efficiency", Electronics Quality, 2005: pp. 34-36, doi: cnki: ISSN: l 003-0107.0.2005-03-012.

[11] Sk.Md.Rafi, K.Nageswara Rao and Shaheda Akthar," Software reliability growth model with logistic-exponential TEF and analysis of software release policy", IJCSE, VOL. 2, No. 2, 2010, pp. 387-399.

[12] S.Yamada, S.Osaki, "Software reliability growth modeling: models and applications", IEEE Trans. Software Engineering, vol.l, no.12,p.1431-1437, 1985.

[13] J.H. Lo and C.Y. Huang, "An integration of fault detection and correction processes in software reliability analysis", Journal of Systems and Software, vol. 79, pp. 1312-1323, 2006.

[14] NASA KSC and Tim.C.Adams, *THE LAPLACE TEST*[Online]. Available:http://kscddms.ksc.nasa.gov/Reliability/Documents/Laplace_Test.pdf

[15] Robert N.Charette (2005 September). *Why Software Fails*[Online]. Available:http://spectrum.ieee.org/computing/software/why software-fails