Connecting Software Reliability Growth Models to Software Defect Tracking

Reoprt Writing

Nikhil Soni, Shubham Kumar

I. Introduction

SOFTWARE reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment. Over the past, many software reliability growth models (SRGMs) have been proposed and it was assumed that in most of SRGMs, faults detected are immediately corrected. Reliability growth models existed was supposed to do only the defect finding in the softwares. Sometimes detected faults can be corrected if we remove the leading defects. So the authors have focused on the software engineer's concern that is to remove the defects immediately. So this paper has tried to connect the defect tracking with software growth models .

Defect resolution is the important feature of this paper so authors developed differential equation-based models, distributional and Markovian models. Data has been used is taken from a project of NASA to prepare the models. Athours also added that data collection and analysis part should be handle carefully that will increase the chances of growth models to be succeeded. A joint modeling of defect discovery and resolution which uses semi-Markovian or distributional process to include an arbitrary life cycle of defects for monitoring purpose. The distributional model used in this paper is used to analyze the model parameters when the unresolved defects found, and this is very useful in prediction during testing and resolving part.

II. Advantage

Authors applied the Goel-Okumoto, Weibull, Yamada Delayed S-shaped, inflection S-shaped, Jelinski-Moranda [24], and Geometric model [25] to input dataset of NASA [20] and concluded that the inflection S-shaped model is best for defect discovery process.

To focus on need of software reliability, authors used retrospective analysis which shows characteristic of historical defects discovery.

Markovian model for defect resolving will provide better results if defect tracking guidelines are followed more consistently.

Author not only stuck to one approach but also many approches like Resolution approach, Distributional approach and Markovian approach in which Markovian find out to be best with sum of squares error SSE=5.486×103.

The paper shows us how beneficial to collect high quality data for defect tracking and reslution.

III. Advancement

The software reliability growth model quantifies the decreasing time for discovering the defects and increasing the corresponding time in reliability. Earlier works which included the attempt to characterize software defect discovery and plan, consist of the work done by Schneidewind, who analyzed the defects with a discrete exponential mean value function and a resolution with a time lag. And then Xie and Zhao, who proposed a research paper on Schneidewind software reliability model revisited, extended Schneidewind's model by assuming that the defect resolution rate is directly proportional to the number of defects discovered but not yet resolved. M. Ohba, in 1984, proposed the S-shaped inflexion model that says that some defects need to be resolve before other defects. M. Ohba and fellow researchers S. Yamada and S. Osaki introduced the delayed S-shaped model with a time delay. In the year 1995, P. Kapur and S. Younes proposed a software reliability growth model that depends on defects. In later studies done by S. Gokhale, P. Marinos, M. Lyn, and K. Trivedi, who proposed multi-priority queuing models for the software defect resolution process, considered the priority levels, the effects of queuing the system structure, and priority disciplines on the differential mean times to resolve the defects of various severities. Shibata, K. Rinsaka, T. Dohi, and H. Okamura in the year 2007 proposed, Quantifying software maintainability based on a fault-detection/correction model for defect discovery and resolution process. C.-Y. Huang, C.-T. Lin, S.-Y. Kuo, M. Lyu, and C. Sue, showed how several software reliability growth models could be derived by applying time dependency delay to a function, while the other researchers, J.-H. Lo and C.-Y. Huang, proposed that an integrated defect discovery and resolution process can be expressed in terms of time-varying resolution intensity. There is a difference between the number of defects discovered and it being resolved. N. Ullah, M. Morisio, and A. Vetro, also performed a comparative study on software reliability growth model on discovery and resolution data from various open-source projects. In the year 2015, Liu, M. Xie, J. Yang, and M. Zhao, proposed a method to estimate the parameters for removing the defects, which includes the approximate time distribution from the point of time the defect was discovered. M. Cinque, D. Cotroneo, A. Pecchia, R. Pietrantuono, and S. Russo, come up with a debugging-workflow-aware software reliability growth model to strengthen debugging data managed by companies while checking for bugs while tracking the systems for the improvements. H. Okamura and T. Dohi, proposed a generalized bivariate modeling framework of fault detection and correction processes along with a model for hyper-Erlang distributions and expectation for maximizing to estimate its parameters.

IV. Related Work

The authors have also gone through alternative methods to model the discovery and resolution of software defects. Some of the ways are discussed below.

A. Integrated defect discovery and resolution processes

In this process, the rate of discovering and resolving the defects are expressed as differential equations processing forms.

$$\frac{dm(t)}{dt} = \lambda(t)(a - m(t)) \tag{1}$$

and

$$\frac{dm_r(t)}{dt} = \lambda_r(t)(m(t) - m_r(t)) \tag{2}$$

Lo and Huang derived general form of the solutions for this equation to produce expression for the mean value function for discovering defect and resolution. The researcher applied the Goel-Okumoto, Weibull, Yamada Delayed S-shaped, inflection S-shaped, Jelinski-Moranda, and Geometric model to a data set. They found that the inflection S-shaped model best suitable for defect discovering process. Therefore, two explicit forms is modeled by the inflection S-shaped model, possessing mean value function

$$c = \frac{1-r}{r}, \qquad r \in (0, 1]. \tag{4}$$

and r is the inflection rate. As r approaches 1.0, the inflection S-shaped model reduces to $m(t) = a(1 - e^{-bt})$, which is the form of the Goel-Okumoto model.

The first form of the defect resolution model assumes defect resolution intensity $\lambda_r(t) = b$, is the same as parameter b present in Equation (3), producing

$$m_r^b(t) = a\left(1 - e^{-bt} + (1+c)\log\left(\frac{1+c}{c+e^{bt}}\right)e^{-bt}\right) \tag{5}$$

The second form of the defect resolution model introduces an additional parameter through $\lambda_r(t) = d$, which implies that the defect resolution intensity is distinct from parameter b in Equation (3), producing

$$\begin{aligned} m_r^d(t) \quad = \quad & \frac{a}{c(b+d)}\Bigg(d\,{}_2F_1\left(1, \frac{b+d}{b}; 2+\frac{d}{b}; -\frac{1}{ce^{-bt}}\right)e^{bt} \\ & -(b+d)\,{}_2F_1\left(1, \frac{d}{b}; \frac{b+d}{b}; -\frac{1}{ce^{-bt}}\right) \\ & +(b+d)\,{}_2F_1\left(1, \frac{d}{b}; \frac{b+d}{b}; -\frac{1}{c}\right)e^{-dt} \\ & -d\,{}_2F_1\left(1, \frac{b+d}{b}; 2+\frac{d}{b}; -\frac{1}{c}\right)e^{-dt}\Bigg) \end{aligned} \tag{6}$$

B. Distributional Approach

To make the resolution time predictions possible, it is really necessary to depend on the discovery and resolution times available up until time t. The problem of identifying the distribution for the best fit, the defect resolution times, when only a subset of the defects discovered has been resolved, This may be estimated from the following likelihood function.

$$Lik = \prod_{i \in \mathcal{R}}^{k} f(t_{(i)}; \theta) \prod_{i \in \mathcal{D}}^{n-k} 1 - F(T - t_i; \theta) \tag{7}$$

C. Markovian Approach

A Markov chain is a stochastic process where a sequence of states X_t, which takes the values from the finite set of 'm' possible states. And in the context of software defect tracking, the states represent the possible stages for the life cycle from discovery to resolution. So, there is a one-to-one correspondence between the states of the defect life cycle and the one-step transition probability matrix for the Markov Chain.

Markov hypothesis states that the, present state at time 't' is conditionally independent for those up to and including time (t2) and only depends on the state at time (t1) such that

$$\begin{aligned} & P(X_t = i_0 | X_0 = i_t, ..., X_{t-1} = i_1) \\ & = P(X_t = i_0 | X_{t-1} = i_1) = q_{i_1, i_0}(t) \end{aligned} \tag{9}$$

where $i_t, ...i_0 \in \{1, ..., m\}$. Assuming the state transition probability $q_{i_1,i_0}(t)$ is time-invariant reduces to q_{i_1,i_0}, producing a homogeneous Markov chain.

Considering all combinations of i_1 and i_0, we construct the transition matrix

$$\begin{array}{cc} & \begin{array}{cccc} 1 & .. & .. & m \end{array} \\ Q = & \begin{pmatrix} q_{1,1} & q_{1,2} & .. & q_{1,m} \\ q_{2,1} & .. & .. & .. \\ .. & .. & .. & .. \\ q_{m,1} & .. & .. & q_{m,m} \end{pmatrix} \begin{array}{c} 1 \\ . \\ . \\ m \end{array} \end{array}$$

where rows denote the previous state (X_{t-1}), columns the successor state (X_t), and each row sums to 1.0. In practice, the entries $q_{i,j}$ may be determined from the information contained in the defect tracking database by computing the number of times state i transitioned to state j and dividing by the total number of transitions out of state i. This is possible because each row of the defect tracking database provides distinguishable data on the lifecycle (sequence of state transitions) within the defect tracking model.

D. Higher-order Markov Chains

In higher-order Markov chains, maximum likelihood estimation of transition probabilities, and a hypothesis test to identify the order of the Markov chain is done.

V. Methodology

Here the author used software reliability growth model to connect through the defect tracking databases. Then, differential-equation based, distributional, and Markovian approaches were used to develop. ANASA defect tracking database was cleaned and a state model was basically constructed. Then a statistical hypothesis was applied to check the order, whether it is a first or second order Markovchain, which best characterized the transitions taken during the defect lifecycle. Models were applied to the full data, referred to as retrospective analysis, and the steps taken to apply the distributional and Markovian approaches explained. Alternative models for the goodness of fit was also assessed with respect to the defect resolution intensity. And then, at last the predictive accuracy and computational efficiency were assessed in an online manner.

VI. Conclusion Future Work

The paper presented has well connected the defect tracking defect resolution. Approaches like Differential equation based, Distributional and Markovian were well tested and applied. Distributional approach was best fit into defect resolution data which shows the lowest run time and provided online tracking during the detection and resolution of defects. Experimental results shows that proposed method of linking software reliability model to defect tracking and resolving has worked well under distributional approach which will really help to improve the efficiency of software development.

National Aeronautics and Space Administration (NASA) has supported this work under Grant Number (80NSSC18K0154) and NSF award (1749635). Identification of factors which reduce the cost of modeling for resolving defects can be a good research area for future. Resource allocation of the resources involved in defect resolution can be improved if researched properly in future.

VII. Individual Contribution

The section Introduction, Advanatges, Conclusion Future Work is done by Nikhil Soni

The section Advancement, Related Work, and Methodology is done by Shubham Kumar