JELINSKI-MORANDA SOFTWARE RELIABILITY GROWTH MODEL: A BRIEF LITERATURE AND MODIFICATION

Lutfiah Ismail Al Turk[1] and Eftekhar Gabel Alsolami[2]

[1] Statistics Department,King Abdulaziz University,Kingdom of Saudi Arabia

[2]Mathematics Department,University of Jeddah,Kingdom of Saudi Arabia

ABSTRACT

Analyzing the reliability of a software can be done at various phases during the development of engineering software. Software reliability growth models (SRGMs) assess, predict, and controlthe software reliability based on data obtained from testing phase.This paper gives a literaturereview of the first and wellknownJelinski and Moranda(J-M) (1972)SRGM.Also a modification to Jelinski and Morandamodel is given, Jelinski and Moranda and Schick and Wolverton (S-W) (1978)SRGMsare two special cases of our new suggested general SRGM. Our proposed general SRGMalong with our Survey will open doors for much more useful researches to be done in the field of reliability modeling.

KEYWORDS

Software reliability growth model (SRGM),Jelinski and Moranda(J-M) SRGM, Schick and Wolverton (S-W)SRGM, GenerlizedJelinski-Moranda (GJ-M) SRGM.

1. INTRODUCTION

Software reliability is defined as the probability of failure-free software operation in a specified environment for a specified period of time [Lyu(1996)]. A software failure is defined as the disparity between the behavior of the software and its specifications. During the testing phase and according to a specific input, a latent software fault may cause a software failure. One of the approaches to describe the single system software failure behavior and quantify software reliability is software reliability growth models (SRGMs) that based on data collected duringtesting phase.Enormousnumber of SRGMs have been proposed during the past 45 years, two general classification of software reliability models (SRMs) are: white box and black box models; the first take into account the internal structure of a software while the second donot.In (1972) one of the earliest black box SRMs, Jelinski–Moranda (J–M) model,was proposed. This continuous time-independently distributed inter failure timesSRGM which also assumes independent and identical error behavior forms the basis for many other SRGMs, large number of research studies have used and considered this model. This earliest SRGM supposes that failures occur according to the Poisson process with ahazard rate decreasing as more faults are detected and successfully removed.One of the earliest studies that modified this model is the study by

DOI : 10.5121/ijsea.2016.7204

Schick and Wolverton in (1978), his modification based on suggesting increasing failure ratebetween successive failures instead of the previously suggested constant failure rate by Jelinski and Moranda (1972). The purpose of this work is divided into two parts: firstly,providing areview of literatureon studies of thewell-knownJ-M SRGM and secondly, generalizing the hazard rate function of the Schick and Wolverton (S-W) SRGM by considering a new shape parameter into its formulation. Through our review of literature, which shows the importance of J-MSRGM and being thefoundation of many others research studiesthe importance of our new modification isdemonstrated. Great contribution in the field of reliability will be gained by re-applying those previous studies on the new suggested general formula. According to changing the values of the suggested shape parameter several SRGMs will be generatedwith different hazard rate behaviour.This ultimately increases the possibility of finding the best fit model for a particular situation with less effort and time and help with the problem of not having a standard reliability model for all data sets. The J-M and S-W SRGMs are considered as two sub-models of our suggested general formula.

2. REVIEWS on The Jelinski-Moranda (J-M)Model

In (1972) Jelinski and Moranda developed SRGM which based on several assumptions among them that the hazard rate of each fault does not change over time, but remains constant. In the following the assumptions of this model will be mentioned in more details, the model's measures of reliability will be summarized, a review of several research studies in different areas of this very common model will be provided.

2.1.Assumptions of Jelinski-Moranda Model

J-M Model assumes the following:

- At the beginning of testing the software code contains unknown but fixed N faults.
- During a testing phase, each fault in the code is independent and equally likely to cause a software failure
- Time intervals between software failures are independent and are exponentially distributed.
- Removing detected faults occurs with certainty whenever a failure happens, and no new faults are introduced during the removal process.
- The software failure rate during a failure interval is constant and proportional to the number of faults remaining in the software.

Because of its unrealistic assumptions (iii, iv and v), this model loose the flexibility of suiting various data cases but even with that this model remained to be the best known SRGM and its simplicity attracted lots of researchers over the past 45 years. The model was developed for use on a Navy software development program as well as a number of modules of the Apollo program. In this work assumptions (iii and v will be modified to produce a generalized SRGM with more flexibility at describing the time dependent behaviour of testing phase.

2.2.Characteristicsof Jelinski-Moranda Model

The constantsoftware failure rate of the J–M model at the i^{th} failure interval is given by:

$$\lambda(t_i) = \varphi_{J-M}[N - (i - 1)], \qquad (1)$$

where

φ: theconstant failure intensity contributed by each failure.
N: the number of latent software faults before the testing starts.
t_i : the time between two consecutive failure $(i-1)^{st}$and i^{th} failures.

The mean value and the failure intensity functions for this model which belongs to the binominal typecan be obtained by multiplying the inherent number of faults by the cumulative failure and probability density functions (pdf) respectively:

$$\mu(t_i) = N(1 - e^{-\varphi t_i}) \qquad (2)$$

and

$$\xi(t_i) = N\varphi e^{-\varphi t_i} \qquad (3)$$

Those characteristics plus four other characteristics of the J-M model are summarized in Table 1.

Table 1:List of various characteristics underlying the J-M model.

Measures of reliability name	**Measures of reliability formula**
The probability density function	$f(t_i) = \varphi[N - (i - 1)]e^{-\varphi[N-(i-1)]t_i}$
The software reliability function	$R(t_i) = e^{-\varphi[N-(i-1)]t_i}$
The failure rate function	$\lambda(t_i) = \varphi[N - (i - 1)]$
The mean time to failure function	$\mathrm{MTTF}(t_i) = \dfrac{1}{\varphi[N - (i - 1)]}$
The mean value function	$\mu(t_i) = N(1 - e^{-\varphi t_i})$
The failure intensity function	$\xi(t_i) = N\varphi e^{-\varphi t_i}$
The median	$m = \{\varphi[N - (i - 1)]\}^{-1}\ln 2$

2.3.Different Studies: Comparatives,Modification,Parameters Estimation of Jelinski –Moranda Model

After 1972, many later SRGMs aremodifications, improvements, or extensions of the J-M model with lots of other studies that used this famous and simple model in applications.In the following we will mention some:

Many authors haveaimed to modify the J-M modelthrough different approaches in the following some of these studied will be surveyed:

Littlewood and Verrall in (1973) proposed the most well-known Bayesian SRGM for a software which is considered as a modification to the J-M model that assumesφ is a proportionality constant represents the improvement in hazard rate after each fixing. The concepts underlying their proposed Bayesian model are introduced within the software development context, but they are easily applicable to a wide variety of situations. They assumed exponential time between failures and Φ has the pdf $\beta\mathrm{GAM}(\beta\varphi\,;\alpha)$, where $\mathrm{GAM}(x;\alpha)$is a gamma pdf, $x^{\alpha-1}e^{-x}/\Gamma(\alpha)$ .The scale parameter of the gamma is the focus of the reliability growth in the model. The parameter is allowed to vary with time, which implicitly accounts for the effectiveness of corrective actions that are implemented. Estimation of the gamma scale parameter were also discussed, and numerical procedures were provided along with a Kolmogorov-Smirnov goodness of fit approach that eliminates the need for numerical integration procedures. According to their model, the times till failure of the N faults were found to be independent random variables with N unknown, having a common Pareto distribution of the second kind.

Jewell (1985a) presented a Bayesian software reliability model of the J-M model, the prior distribution of N is suggested to be the negative binomial distribution and the unit failure rate has gamma distribution.Langberg and Singpurwalla (1985) proposed a Bayesian model that interpreting the failure process by assuminga prior distributions on the parameters N and φ of the J-M model, they showed thatGoel and Okumoto(G&O) (1979) and Littlewood and Verall (L-V) (1973) SRGMs arise as special cases of their proposed model.Jewell (1985b) extended the Bayesian model by Langberg and Singpurwalla (1985), the main assumption is that the distribution of the unknown number of faults is assumed to be Poisson whose parameter has a Beta prior distribution.

Littlewood and Sofer (1987) investigated the J-M model and implied that the poor predictive capability obtained by this model because of using the maximum likelihood estimation (MLE) method. They proposed new Bayesian-Jelinski-Moranda model (BJM) and its mathematical tractability was approved,In addition several metrics of BJM were given. Based on several data sets they compared between the J-M and BJM reliability models and shown that the BJM has better performance for allthe studied situations.

Another modification to the J-M model, by Moranda (1975), his modification allows unequal change in software failure intensity after debugging. His suggested model is called the geometric de–eutrophication model and the original assumption that the failure intensity is proportional to the current fault content is replaced by a more realistic one. On this basis the Moranda model would seem to be more realistic and useful.

In (1978) Schick and Wolvertonproposed a model which assumed that the hazard rate function is proportional to the product of the number of faults remaining and the testing time:

$$\lambda(t_i) = \varphi_{S-W}[N - (i - 1)]t_i \, . \tag{4}$$

This model gives the best prediction for the remaining errors for data that obtained from projects with large testing phase, and it is considered as a modification of the J-M model that overcomes the problem of constant hazard rate.

Joe and Reid (1985b) gave new alternative formulas of the J-M and L-V SRMs, their idea depends on replacing the inter-failure time by the failure time. Basically, Their models based on observing the first n order statistics from a random sample of N exponential and Pareto distributions respectively. They discussed the maximum likelihood estimateand an improved estimate of the initial number of faults in the software of the J-M model which was previously introduced by Joe and Reid (1985a). Their study gave a way to help in deciding about the suitability of the J-M model or the software reliability model with decreasing failure rate.

Ho et al. (1991) modified the J-M model by replacing the assumption of all faults contribute the same in the failure rate by another more realistic one. They assumed that different types of faults may have different effects on the failure rate of the software, for accomplishing that they presented a method of determining the size of the failure-quantum of a fault based on the internal structure of software. Their application that based on set of real data showed that the predictive capability of their model is better than the J-M model.

Wang et al. (2002) proposed formulation to estimate the total number of faults in a system. The L-V and J-M models are two special cases of their proposed formulation. Simulation studied were given and also two real data sets were used in an illustrative example. According to the results, their proposed two-stage estimating procedure, a conditional likelihood and a Horvitz-Thompson estimator, were found to be efficient.

Luo et al. (2011) introduced mathematical description of the J-M model. They presented a novel approach to the modification of the J-M model based on cloud model. The J-M model only gives a point or interval estimation for a reliability index based on analysis of some failure data. Point estimation values vary with different samples,even with in one sample, point or interval estimation is also different due to different statistics. In the cloud model which is a new cognitive model for uncertain transformation between linguistic concepts and quantitative values, they employed the expectation, the entropy, and the hyper–entropy to represent the concept as a whole. Especially, the normal cloud model can avoid the flaw of fuzzy sets to quantify the membership degree of an element as an accurate value between 0 and 1, Therefore, may be more adaptive for the uncertainty description of linguistic concepts. Then cloud model can be utilized to represent software reliability so as to deal with the universal uncertainty in the concept, which contains many kinds of uncertainties, and in particular randomness, fuzziness and the correlation between them.Their approach was demonstrated with a real software reliability data set.

During the imperfect debugging process, two types of imperfect removal can be occurred: (i) the fault is not removed successfully while no new faults are introduced and (ii) the fault is not removed successfully while new faults are created due to incorrect diagnoses. Mahapatra and Roy (2012) modified the J-M model by considering the imperfect debugging process in fault removal activity instead of the assumptions of perfect debugging process. They considered the second type of imperfect removal which is the most practical situation in fault removing activity. They allowed the imperfect debugging process to introduce new faults into the software due to incorrect modifications or diagnoses. The parameters of their modified J-M model were estimated by using the maximum-likelihood estimation method. An illustration example was given based on real data set, According to their experimental results they concluded that the difference between the probability of perfect debugging and the probability of raising new faults should be decreased to get better predictions results. Also they showed that their developed model has better predictive capability than the J-M model.

Many researchers have used the J-M model to demonstrate the goodness of their suggested ideas, methods, or to make a comparative studies to show the superiority of their suggested models, the following are some:

Michael (2000) compared and discussed the basic differences and commonalities between the two popular software reliability models: J-M and Musa-Okumoto models. He focused on specific attributes and qualities of the two models, such as the form of their respective failure rate functions, the data inputs required to exercise the models, and the assumptions that must be true for the models to produce reliable and useful data. Finally and according to his obtained reliability predictions, he gave a short analysis of the differences between the two models.

Ahuja et al. (2002) considered the J-M model for software reliability prediction of failure time. For analyzing and studying the accuracy of their obtained prediction results they used different techniques which are Brawn statistic, prequential likelihood function, U-plot and Y-plot and the Kolmogorov–Smirnov distance. They also generated optimized simulation trajectory based on evolutionary algorithm for predicted mean time to failure (MTTF). According to their study they found that the MTTF values is much closer for predictions which are less noisy and they concluded that their experiments will be useful for better reliability monitoring of software.

Kim et al. (2007) demonstrated that there are some possibilities that software reliability growth models to be applied for the sake of proving the high reliability of safety-critical software at the point where all the inherent software faults are identified and correctly repaired, to illustrate their idea they used the J-M model. However, they also described the limitations of these possibilities caused by either the high sensitivity of the estimated total number of inherent software failure time data which is obtained by the SRGMs or the uncertainty of the availability of sufficient software failure data sets for the safety-critical software.

Srivastava and Sharma (2014) discussed using the neural network approach instead of the parametric SRGMs to predict the reliability, their opinion is to produce a model with no assumptions about the behavior of software failure. In their study they compared between the neural network model and two parametric, J-M and Musa-Okumoto,SRGMs.Through a simulation study, they found that the neural network has the best performance among the selected parametric SRGMs.

In practice, the unknown parameters of the time-dependent J–M model is estimated from data astimes taken between successive failures. Several different estimation methods have been proposed in the literature. Many authors have used the maximum likelihood estimation (MLE) and least squares estimation (LSE) methods to estimate the model parameters. In the following some literature in this regards will be mentioned:

The maximum likelihood (MLE) method is a widely used method to estimate the unknownparameters in reliability models, it provides a consistent approach to parameter estimation problems. The J-M model suffers from difficulties associated with parameter estimation (giving decreasing reliability), this model has poor predictive capabilities in many cases. As for software reliability growth models, Littlewood and Verrall (1981) presented a simple necessary and sufficient condition for the maximum likelihood estimates of J-M model to be finite and suggested that this condition be tested prior to using the model.

Spreij (1985) studied the problem of maximum likelihood estimation in J-M model. He obtained the distribution of the stochastic variable that completely determines the maximum likelihood estimate, and then by using the same stochastic variable the s-confidence intervals for the initial error content of the program was formed. An illustrative study was conducted using an inter-failure time real data set.

Van Pul (1992) investigated how well the maximum likelihood estimation procedure and the parametric bootstrap behave in the case of the software reliability J-M model. He discussed the results of computations, estimations and statistical methods based on simulated data using the J-M model.

The least squares estimation (LSE) method are also employed to estimate the J-M model parameters, Schafer et al. (1979) proposed the traditional LSE technique to estimate the parameters of J-M model.Qureshi and Jeske (1997) introduced the concept of proxy failure times for situations where system test data only consists of the fraction of test cases that fail for a set of execution scenarios. They showed how proxy failure times can be simulated if external information about the user-frequency of the test cases is available. Also, they developed statistical inference procedures for fitting the J-M model. In particular, they presented a graphical diagnostic for testing goodness-of-fit and showed how it suggests appropriate transformations of the failure times that would improve the fit. Influential observations were also identified by the diagnostic, and moreover, it provides regression estimators of the model parameters as a quick alternative to the maximum likelihood estimators. Formulas for Iikelihood-based confidence intervals for the model parameters were provided. The simulation of proxy failure times and the statistical inference procedures for the J-M model were illustrated with an example. The use of profile likelihood functions and graphical goodness-of-fit was shown to be new suggestions to the analysis procedures associated with fitting this model.

Cai (1998) discussed two LSE methods, least squares method type I and least squares method type II. Since J-M model is an exponential class model.Liu et al. (2008) derived the logarithm nonlinear least squares estimation (LogLSE) of the J-M model, they evaluated its performance based on three real failure data sets. Liu and Xu (2011) focused on the time-independent modeling of J-M model with LSE. As an extension to the logarithm nonlinear least squares estimation LogLSE method, they developed a general function based nonlinear least squares estimation (FNLSE) method by merging the compression merits of transformation function in statistics with the weighted nonlinear least squares estimation (WNLSE), their aim was to overcome the statistical modeling problem produced by heteroscedasticity. Theyshownthat FNLSE is a WNLSE method, and proposed a power function based LSE (powLSE) to estimate the parameter of J-M model. They conducted an application using six bench–mark failuretime databases to examine the LSE, MLE, LogLSE and powLSE methods, their prediction results of MTBF prediction shown the effectiveness of their novel powLSE method.

Jukić (2011) considered the Lp-norm($1 \leq p < \infty$) estimation problem thatthe best Lp-norm estimate does not necessarily exist for the J-M model. First, he briefly reviewed the J-M software reliability model. Then, he described Lp-norm fitting problem and gave a necessary and sufficient condition which guarantees the existence of the best Lp-norm estimate. He presented two theorems on the existence of the LS estimate. The first theorem gives a necessary and sufficient condition for the existence of the LS estimate. For practical purposes, the second theorem is extremely important, as it gives a very simple and natural sufficient condition for the existence of the LS estimate. His paper concluded with some examples illustrating the problems arising with the nonlinear normal equation approach for solving the LS problem for the J-M model.

3.GENERLIZEDJELINSKI-MORANDA (GJ-M) MODEL

Our modificationbased on addingadditionalparameter$\eta$to the hazard rate function of the S-W model in Equation (4). This shape parameter makes the model more flexible andincreases the possibility of giving the best prediction for the remaining errors for data that obtained from projects with various situations. The hazard rate function of our suggested general formula is of the form:

$$\lambda(t_i) = \eta\varphi_{GJ-M}[N - (i - 1)]t_i^{\eta-1}. \tag{5}$$

Where, as in the J-M model,φ is a proportionality constant, N is the number of initial faults present in system and t_i is thei^{th} time interval between detection of $(i-1)^{st}$and i^{th}faults. The basic assumption of the GJ-M model are:

- The amount of debugging time between fault occurrences has a Weibull distribution.
- The faults rate is proportional to the number of faults remaining and the term $\eta t_i^{\eta-1}$.
- Each fault discovered is immediatelyremoved thus reducing the number of faults by one.

The reliability of this model could be obtained as follows:

$$\mathrm{R}(\mathrm{t_i}) = e^{-\int_0^{\mathrm{t_i}} \lambda(\mathrm{t_i}) d\mathrm{t_i}} = e^{-\int_0^{\mathrm{t_i}} \varphi_{\mathrm{GJ-M}} \eta [\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta-1} d\mathrm{t_i}}$$
$$= \mathrm{e}^{-\varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta}} \quad (6)$$

The failure probability density function(pdf) is used to determine the probability of at least one failure in the time period $\mathrm{t_{i-1}}$ to $\mathrm{t_i}$and can beobtained as follows:

$$\mathrm{f}(\mathrm{t_i}) = \lambda(\mathrm{t_i})\mathrm{R}(\mathrm{t_i}) = \varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)]\eta t_i^{\eta-1} \mathrm{e}^{-\varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta}} \quad (7)$$

More measures of reliability of the GJ-M are summarized in Table 2. Notice that the J-M model can be obtained when $\eta = 1$ while theS-W modelis a special case when $\eta = 2$ and$2\varphi_{\mathrm{GJ-M}} = \varphi_{S-W}$.

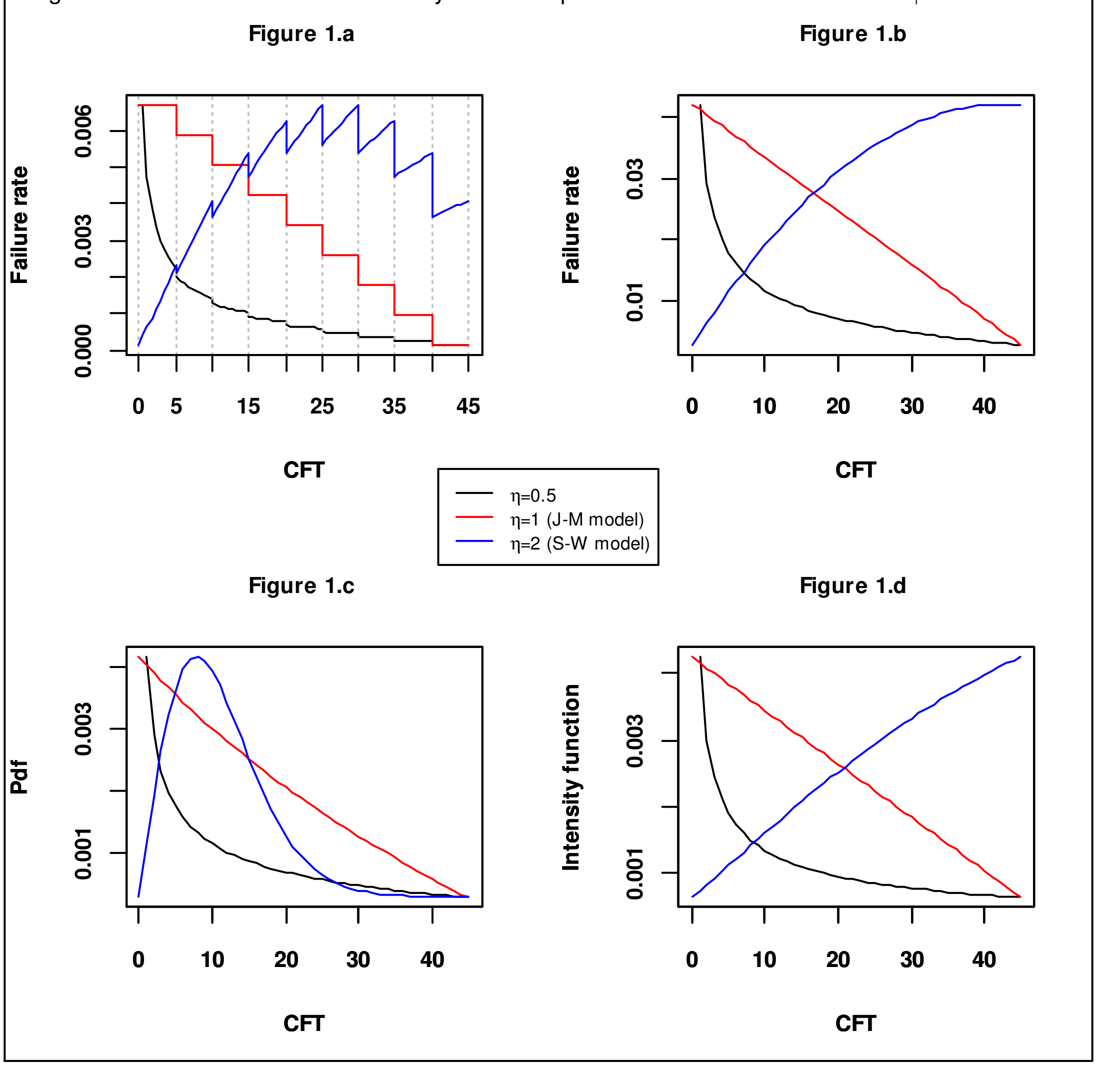
Also some of the GJ-M model's characteristics are represented graphically in Figure 1, this figure shows the curves of some measures of reliability for three special cases of the GJ-M SRGM with respect to operating time. The hazard function measures the probability of experiencing the failure in a given time period conditional on not having experienced the failure up to that period,in Fig. 1.a and Fig. 1.b two graphical representation of the hazard function of the GJ-M are shown: monotone decreasing, constant, and increasing failure rate can be provided by our suggested GJ-M SRGM as seen in those plots. The improvement at each fixing is illustrated in Fig. 1.b, as seen we have decreasing hazard function betweensuccessive failureswhen $\eta = 0.5$, constant when $\eta = 1$ and increasing when $\eta = 2$ but in the three cases it decreases by a factor $\varphi_{\mathrm{GJ-M}}$ in steps of following the removal of each fault. Therefore, as each fault is removed, the time between failures is expected to be longer.The failure density function is shown in Fig. 1.c, at $\eta = 0.5$ and 1 the pdf decrease exponentially with time, while at $\eta = 2$ it shows an initial raise before decreasing as operating time increases. The intensity function plays the same role as the hazard function and gives the rate of failure occurring. As seen in Fig. 1.d, the behavior of the intensity function for the GJ-M model is similar to the GJ-M model's hazard function: at$\eta = 1$ it decreases linearly with failure time, also at $\eta = 0.5$ it decreases exponentially on time interval, but at $\eta = 2$ it increases on time interval.

Table 2: List of various characteristics underlying the GJ-M model.

Measures of reliability name	Measures of reliability formula
The probability density function (pdf)	$\mathrm{f}(\mathrm{t_i}) = \varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)]\eta t_i^{\eta-1} \mathrm{e}^{-\varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta}}$
The software reliability function	$\mathrm{R}(\mathrm{t_i}) = \mathrm{e}^{-\varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta}}$
The hazard rate function	$\lambda(\mathrm{t_i}) = \eta\varphi_{\mathrm{GJ-M}}[\mathrm{N}-(\mathrm{i}-1)] t_i^{\eta-1}$
The mean time to failure function	$\mathrm{MTTF}(\mathrm{t_i}) = \frac{1}{\eta}\{\varphi_{\mathrm{GJ-M}}[N-(i-1)]\}^{-\frac{1}{\eta}} \Gamma\left(\frac{1}{\eta}\right)$

The mean value function	$\mu(t_i) = N\left(1 - e^{-\varphi_{GJ-M} t_i^{\eta}}\right)$
The failure intensity function	$\xi(t_i) = N\varphi_{GJ-M}\eta t_i^{\eta-1} e^{-\varphi_{GJ-M} t_i^{\eta}}$
The median	$m = \left[\left\{\varphi_{GJ-M}[N-(i-1)]\right\}^{-1} \ln 2\right]^{\frac{1}{\eta}}$

Figure 1: Some measures of reliability for three special cases of the GJ-M model ϕ=0.001 N=85



4. CONCLUSION

Our survey of the commonly cited J-Mmodel that was suggested at the early seventies gives insight at its significant impact in the field of reliability modeling, asit forms the basis for many useful studies in this field.Schick and Wolverton (1978) modified this model by a more realistic assumption, our idea is generalizing his suggested hazard rate formula through adding new shape parameter. The new hazard rate general formula has a great flexibility in accommodating all formsof time- dependent behavior and can introduce a variety of SRGMs that could be used with less effort and time in any model selection study, and will increase the chance of finding the best fit model for a variety of problems for modeling software failure data. Another important characteristic of the GJ-M is that it contains special sub-models, the common J-M and S-WSRGMs. Plus all the previous studies that haven been done based on J-M model can be re-applied on our general formula with wider expected findings.

REFERENCES

[1] AhujaS., Mishra G. and TyagiA.(2002). Jelinski-Moranda Model for Software reliability Prediction and its G.A.based Optimised Simulation Trajectory, IEEE Transactions on Evolutionary Computation 47 (1), 79-87.

[2] Cai K.Y.(1998). " On Estimating the Number of Defects Remaining in Software " . Journal of Systems and Software, 40(2), 93-114. Elsevier Science Inc. New York, NY, USA.

[3] Goel A.L. and Okumoto K. (1979). Time Dependent Error Detection Rate Model for Software Reliability and other Performance Measures, IEEE Transactions on Reliability, R-28(3), pp. 206-211.

[4] HoT., ChanW. and ChungC.(1991).A Quantum Modification ToThe Jelinski-Moranda Software Reliability Model.

[5] Jelinski Z. and Moranda P.B. (1972). Software reliability research, Statistical computer performance evaluation (Edited by W. Freiberger), Academic Press, New York, pp. 465-497.

[6] Jewell W. (1985a). Bayesian estimation of undetected errors. In Bayesian Statistics 2 (Edited by J. Bernardo, M. DeGroot, D. V. Lindley and A. F. M. Smith), 663-672. Amsterdam: North-Holland.

[7] Jewell W.S. (1985b). " Bayesian extensions to a basic model of software reliability ". IEEE Transactions on Software Engineering, 11 (12), pp. 1465-1471.

[8] Joe H. and Reid N. (1985a). " On the software reliability models of Jelinski-Moranda and Littlewood ". IEEE Transactions in Reliability, R-34, 216-218.

[9] JoeH. and Reid N.(1985b). "Estimating the number of faults in a system" system", J. Amer. Statist. Assoc., vol 80, pp 222-226.

[10] Jukić D. (2011). " Total least squares fitting Bass diffusion model ". Mathematical and Computer Modelling. Model. 53 (2011), pp. 1756–1770, doi:10.1016/j.mcm.2010.12.054.

[11] KimM.,JangS. and HaJ.(2007). "Possibilities and limitations of applying software reliability growth models to safetycritical software," Nuclear Engineering and Technology, vol.39, no.2, pp.145-148.

[12] Langberg N. and Singpurwalla N. D.(1985). "A unification of some software reliability models". SIAM Journal Scientific and Statistical Computation, 6(3), 781-790.

[13] Littlewood B. and Verall J.L., (1973). "A Bayesian Reliability Growth Model for Computer Software ". Journal of Royal Statistical Society Series C-Applied, 22(3), pp. 332-345.

[14] Littlewood, B. andSofer A., "A Bayesian modification to the Jelinski-Moranda software reliability model", Software Engineering Journal, 2, pp30-41, 1987.

[15] Lyu M.R.(ed.). (1996). Handbook of Software ReliabilityEngineering, IEEE Computer Society Press and McGraw-Hill, New York, and IEEE CS Press, Los Alamitos.

[16] Liu J.W., Hu Y.B., Wang X.L., andXu M.Z. (2008). Novel Nonlinear Least Square Parameter Estimation Method for Jelinski-Moranda Model. Journal of System Simulation. 20(18), 4791–4794.
[17] Liu J.W. and XuM.Z., (2011). “Function Based Nonlinear Least Squares and Application to Jelinski–Moranda Software Reliability Model”, stat. ME.
[18] LuoZ.,CaoP.,TangG. and WuL. (2011) "A Modification to the Jelinski-Moranda Software Reliability Growth Model Based on Cloud Model Theory," Computational Intelligence and Security (CIS), 2011 Seventh International Conference on, Hainan, pp. 195-198.
[19] MahapatraG. S. and RoyT. K.(2012). “Modified Jelinski-Moranda Software Reliability Model with Imperfect Debugging Phenomenon”. International Journal of Computer Applications (0975 – 888),Volume 48– No.18.
[20] Michael J. Mangieri, "A Comparison Of Two Software Reliability Engineering (SRE) Models: Jelinski-Moranda And Musa-Okumoto Logarithmic Poisson", At CSMN 658: Software Reliability And Reusability The University Of Maryland University College Graduate School December 7, 2000.
[21] Moranda P.B. (1975). A Comparison of Software Error-Rate Models, Texas Conference on Computing, 2A-6.1-2A-6.9.
[22] Schick G. J. and Wolverton R.W. (1978). An analysis of competing software reliability models, IEEE Transactions on Software Engineering, Vol. SE-4, No. 2.
[23] Schafer R.E., Alter J.F., Angus J.E. andEmoto, S.E. (1979). “Validation of Software Reliability Models”. Rome Air Development Center, Technical Report RADC–TR– 79–147, Rome, NY.
[24] SpreijP.J.C. (1985). “ Parameter estimation for a specific software reliability model” . IEEE Tr. Rel. 34, 323-329.
[25] Srivastava S. and SharmaM. (2014). Neural Network analysis of Software Reliability Growth Models, International Journal of Application or Innovation in Engineering and Management, Volume 3, Issue 11.
[26] Van Pul M. C. (1992). “ Simulations on the Jelinski-Moranda model of software reliability; application of some parametric bootstrap method” .entre for Mathematics and Computer Science, Amsterdam.
[27] Wang Y., YipP. and Hayakawa ,Y.,(2002). A FRAILTY MODEL FOR DETECTING NUMBER OF FAULTS IN A SYSTEM, StatisticaSinica 12(2002), 1001-1013

AUTHORS

Lutfiah Ismail Al turk is currently working as Associate Professor of Mathematical Statistics in Statistics Department at Faculty of Sciences, King AbdulAziz University, Saudi Arabia. Lutfiah Ismail Al turkobtained her B.Sc degree in Statistics and Computer Science from Faculty of Sciences, King AbdulAziz University in 1993 and M.Sc (Mathematical statistics) degree from Statistics Department,Faculty of Sciences, King AbdulAziz University in 1999. She received her Ph.D in Mathematical Statistics from university of Surrey, UK in 2007. Her current research interests include Software reliability modeling and Statistical Machine Learning.

Eftekhar Gabel AlsolamiispresentlyTeaching Assistant in MathematicsDepartment at Faculty of Sciences, University of Jeddah, Saudi Arabia. Eftekhar Gabel Alsolamiobtained her B.Sc degree in Statistics from Faculty of Sciences, King AbdulAziz University in 2011.