International Journal of Business Analytics and Intelligence
6 (1) 2018, 31-34
http://publishingindia.com/ijbai/

Optimal Release Time of Software under Fuzzy Environment with Testing Effort

Asit Dwivedi*, Deepak Kumar**

Abstract

Software Release Time Problem (SRTD) focuses on the timely release of software keeping the constraints of Quality and Cost. Several SRTD models have been proposed in literature since 1970s. Software Reliability Growth Models (SRGM) give the mean number of failures during the process of debugging. In perfect debugging, fault is removed completely; whereas in imperfect debugging, faults are not removed perfectly during debugging process. A probability is always attached with the imperfect debugging. This paper focuses on the optimal release time of an imperfect debugging model with constants as cost and reliability in fuzzy environment with testing effort. Results are illustrated numerically, the problem of timely release of software in imperfect debugging is addressed, and optimal release time with given constraints is given.

Keywords: Fuzzy Membership Function, Fuzzy Optimization, Imperfect Debugging, Release Time, Software Release Time Problem (SRTD), Software Reliability Growth Model (SRGM)

Introduction

The main aim of software engineering is to develop software that are reliable in nature. As software plays a crucial role in today's world not only in business and human application as well as in daily lifestyle of an individual. Various works have contributed towards developing the software that is error free in nature; however, developing a reliable software is a major challenge faced by the software-developing industry, as developing such a software consumes lots of resources such as cost and power, with various processes to check its reliability. While developing reliable software, the project manager must be aware of the accurate information regarding how to grow software reliability to effectively maintain the project and the budget sanctioned to it (Khatri et al., 2012).

Various methods were described to produce the software reliabilty growth models using the techinques of soft computing like fuzzy logic and nueral networks (Bector et al., 2005). Various articles also studied the prediction of faults that get accumulated during the software-testing process using various parametric models as well as non-parametric models (Huang et al., 1997). In this paper authors have tried to provide solution for estimation of the defect fix effort using neural networks; for project management, fuzzy logic and neural network were utilized.

This paper presents a software reliability growth model which is used to solve the problem under fuzzy environment, which is given in literature. The efficiency of the given model is already predicted.

Software Reliability Growth Model

Assumptions

The model given in this paper follows the following assumptions:

1. Elimination of faults in the software and surveillance of malfunction follow-up NHPP.
2. During the running of software, the error present in software leads to software failure.
3. Instantaneous efforts takes place after the interpretation of failure to segregate the cause of malfunction; to separate it, various efforts are initiated.
4. Failure rate of software is straightforwardly related to the residual faults in the software.

* Research Scholar, Mewar University, Chittorgarh, Rajasthan, India. Email: asitdwivedi@rediffmail.com
** Visiting Faculty, Mewar University, Chittorgarh, Rajasthan, India.

Notations

$m_r(t)$: expected faults removed in t time

a : opening no. of hidden errors of software

b : rate of error removal

w(t) : effort of testing at a given time t

W(t) : collective effort of testing in a given duration, i.e., $\frac{d}{dt}W(t) = w(t)$

Ts: duration of warranty.

S-shaped SRGM with Testing Effort

The following model is given by Kapur et al. (1991). The equation given below illustrates fault-removal phenomenon with testing effort:

$$\frac{dm_r(t)}{dt}\times\frac{1}{w(t)} = b(t)(a - m_r(t)) \quad \frac{dm_r(t)}{dt}\times\frac{1}{w(t)} = b(t)(a - m_r(t)) \tag{1}$$

where, $b(t) = \left(\frac{b^2 W(t)}{1 + bW(t)}\right)$

On solving equation (1) given above with primary conditions $m_r(t=0)=0$ and $W(t=0)= 0$, the following mean value function can be found:

$$m_r(t) = a\left[1 - \left((1 + bW(t))e^{-bW(t)}\right)\right] \tag{2}$$

Fuzzy Problem Formulation

Total minimum estimated cost of testing with intended constrictions on reliability is given by the optimal software release time.

Function of the Cost

The estimate cost reliant on cost of the testing per unit time (Co_1), cost sustain in the debugging before releasing the software (Co_2), and the cost sustain in the debugging after releasing the software (Co_3) (Kapur et al., 1999). The function of cost can be given by,

$$\text{Co}(\text{W}(T)) = \text{Co}_1 W(T) + \text{Co}_2 \text{m}(T) + \text{Co}_3(\text{m}(T+T_S)\text{-m}(\text{T})) \tag{3}$$

Function of Reliability

The function of reliability for software is defined by,

$$R((T+Ts)|T) = e^{-(mf(T+Ts)-mf(T))} \tag{4}$$

$$\begin{aligned} &\text{Minimize} && \text{Co}(\text{W}(\text{T})) \\ &\text{Subject to} && \text{R}(\text{Ts}|\text{T}) \gtrsim \text{R}_0 \\ & && \text{W}(\text{T}) \lesssim \text{W}_0 \\ & && \text{T} \geq 0 \end{aligned} \tag{5}$$

The following symbol $\gtrsim$ or $\lesssim$ represents the fuzzy greater or equal to and fuzzy less or equal to optimization techniques of fuzzy logic such as fuzzy mathematical programming is used instead of the crisp optimization techniques to solve the problems regarding the optimization of fuzzy.

Release Time Problem Soultion: Optimization Using Fuzzy

Under this part of paper, using Zimmermann's (2001) approach we discuss various steps pertaining to solve the problem related to software release time under the influence of fuzzy environment as given in section of fuzzy problem formulation. Using the fuzzy environment, the problem (5) is reaffirmed as:

$$\begin{aligned} &\text{Find} && \text{T} \\ &\textit{Such that} && \text{Co}(\text{W}(\text{T})) \leq \text{Co}_0 \\ & && \text{R}(\text{Ts}|\text{T}) \geq \text{R}_0 \\ & && \text{W}(\text{T}) \leq \text{W}_0 \\ & && \text{T} \geq 0 \end{aligned} \tag{6}$$

For every fuzzy inequality, we define membership functions $\nu_i(T)$, i=1, 2, 3 given in the problem as:

$$\nu_1(T) = \begin{cases} 1 & ;Co(T) \leq Co_0 \\ \frac{Co^* - Co(T)}{Co^* - Co_0} & ;Co_0 < Co(T) \leq Co^* \\ 0 & ;Co(T) > Co^* \end{cases}$$

$$\nu_2(T) = \begin{cases} 1 & ;R(Ts\,|\,T) \geq R_0 \\ \frac{R(Ts\,|\,T) - R^*}{R_0 - R^*} & ;R^* \leq R(Ts\,|\,T) < R_0 \\ 0 & ;R(Ts\,|\,T) < R^* \end{cases}$$

$$\nu_3(T) = \begin{cases} 1 & ;W(T) \leq W_0 \\ \frac{W^* - W(T)}{W^* - W_0} & ;W_0 < W(T) \leq W^* \\ 0 & ;W(T) > W^* \end{cases} \tag{7}$$

Here, cost tolerance is given by Co*, R* defines reliability level tolerance, and W* gives testing efforts. In order solve inequalities with in fuzzy system corresponding to problem given in section (6), fuzzy decisions are made using Bellman and Zadeh's (1973) principle. The crisp optimization problem is given by,

Maximize α

Subject $v_i(T) \geq \alpha, \quad i = 1, 2, 3; \quad \alpha \geq 0, \quad T \geq 0 \qquad (8)$

Optimal solution for problem (8) can be found if feasible. Problem (8) can also be solved with help of mathematical programming prior to incorporation of parameter value.

Mathematical Example

It's already presumed that parameters a and b of the SRGM are already predicted and tested on the dataset composed by Obha (1984). Predicted values of constraints over the dataset provided by Obha (1984) are a=354.78 and b=0.0889. Further, it is assumed that values of Co_1, Co_2, Co_3, and T_s are already known. We have taken here Co_1=12, Co_2=30, Co_3=50, and T_s=5. Its presumed total budget cost for management is Co_0=11000, and required reliability for software at delivery time R_0=0.89 and testing effort required is W_0=70 and tolerance level of required cost, reliability, and testing effort are given as Co*=13000, R*=0.999, and W*= 150. The fuzzy optimization technique has been used to find the solution with help of above given values of constants and parameters. The cost, reliability, and testing effort membership functions plotted on reliability are shown in figures.

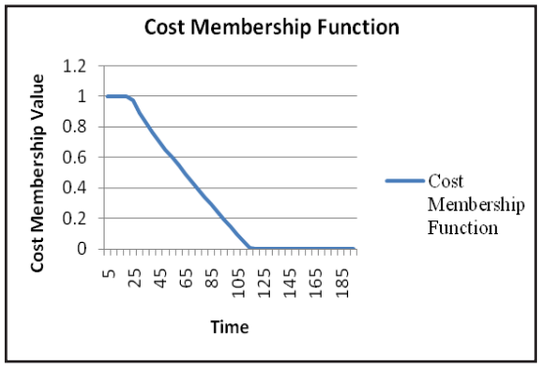

Fig. 1: Cost Membership Function

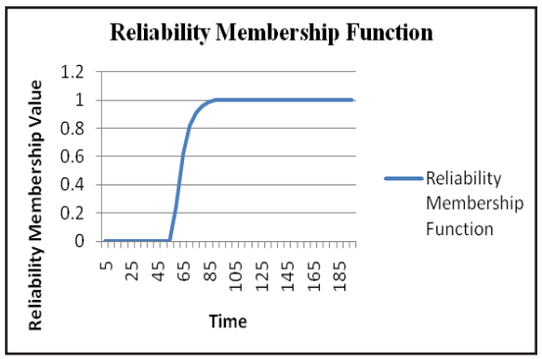

Fig. 2: Reliability Membership Function

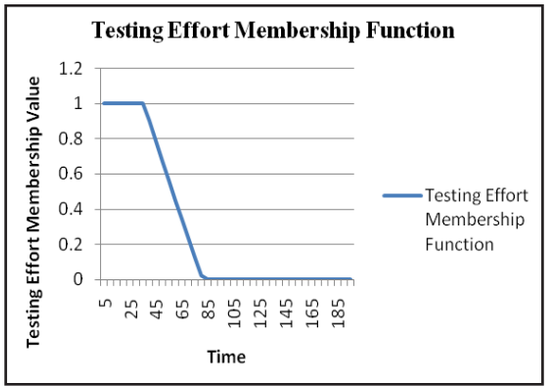

Fig. 3: Testing Membership Function

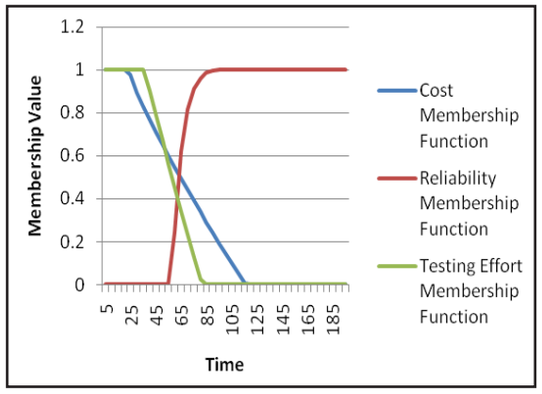

Fig. 4

From Fig. 4, on solving the problem (8) for the optimum software release time, we get T=50 and α=0.6731.

Conclusion

In the given paper fuzzy optimization has been discussed to solve problem pertaining to software release time with

the help of constraints to lower the software cost in account to testing effort and software reliability. A mathematical example along with the result is discussed for solving the fuzzy logic optimization problem.

Authors of this paper have considered cost as the objective function. Due to increasing competition in the growing software industry, software developers try to achieve and consider various things simultaneously, resulting in the optimization of software release time. In future, bi-criterion software release time problem with the help of fuzzy optimization techniques can be discussed.

References

Bector, C. R., & Chandra, S. (2005). *Fuzzy mathematical programming and fuzzy matrix games*. Berlin, Heidelberg: Springer-Verlag.

Bellman, R. E., & Zadeh, L. A. (1973). Decision making in a fuzzy environment. *Management Science*, *17*(4), 141-164.

Huang, C. Y., Kuo, S. Y., & Chen, J. Y. (1997). Analysis of a software reliability growth model with logistic testing effort function. *Proceeding of 8th International Symposium on software reliability engineering*, 378-388.

Kapur, P. K., & Garg, R. B. (1991). Optimal release policies for software systems with testing effort. *International Journal of Systems Science, 22*(9), 1563-1571.

Kapur, P. K., Garg, R. B., & Kumar, S. (1999). *Contributions to hardware and software reliability.* Singapore: World Scientific Publishing Co. Ltd.

Khatri, S. K., Kumar, D., Dwivedi, A., & Mrinal, N. (2012). Software Reliability Growth Model with Testing Efforts using Learning Function. *IEEE explore digital library, Proceedings of International Conference on Software Engineering (Conseg)*, 1-5.

Obha, M. (1984). Inflection S-shaped Software Reliability Growth Model. In Osaki S., Hatoyama Y. (eds) *Stochastic Models in Reliability Theory.* Lecture notes in Economics and Mathematical Systems, *235,* 144-162. Berlin, Heidelberg: Springer.

Zimmermann, H. J. (2001). *Fuzzy set theory and its applications*. New York: Springer.