applied sciences

MDPI

Article

Software Reliability Growth Model with Dependent Failures and Uncertain Operating Environments

Da Hye Lee [1], In Hong Chang [1,*] and Hoang Pham [2,*]

[1] Department of Computer Science and Statistics, Chosun University, 146, Chosundae-gil, Dong-gu, Gwangju 61452, Republic of Korea
[2] Department of Industrial and Systems Engineering, Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08855-8018, USA
* Correspondence: ihchang@chosun.ac.kr (I.H.C.); hopham@rci.rutgers.edu (H.P.)

Abstract: Software is used in various industries, and its reliability has become an extremely important issue. For example, in the medical industry, software is used to provide medical services to underprivileged individuals. If a problem occurs with the software reliability, incorrect medical information may be provided. The software reliability is estimated using a software reliability growth model. However, most software reliability growth models assume that the failures are independent. In addition, it is assumed that the test and operating environments are the same. In this study, we propose a new software reliability growth model that assumes that software failures are dependent and uncertain operating environments. A comparison of the proposed model against existing NHPP SRMEs using actual datasets shows that the proposed model achieves the best fit.

Keywords: software reliability; non-homogeneous Poisson process; dependent failure; uncertain operating environment

check for updates

Citation: Lee, D.H.; Chang, I.H.; Pham, H. Software Reliability Growth Model with Dependent Failures and Uncertain Operating Environments. *Appl. Sci.* **2022**, *12*, 12383. https://doi.org/10.3390/app122312383

Academic Editor: Marco Mesiti

Received: 2 November 2022
Accepted: 1 December 2022
Published: 3 December 2022

1. Introduction

Since software is used in various industries, software reliability is an extremely important issue. For example, in the medical industry, software is used to provide medical services to underprivileged individuals. If a problem with software reliability occurs, incorrect medical information may be provided. Thus, the importance of software reliability cannot be overemphasized.

The software used in various industries has many different functions and its structure is complex. In addition, the reliability (stability) of both software and hardware is extremely important. Software reliability issues can cause financial and human losses. Thus, many researchers have been studying software reliability for decades. To conduct software reliability studies, software reliability growth models (SRGM) should be understood. A SRGM is a tool used to estimate the quality and reliability of software products and provide information to developers and consumers. Consumers can refer to such information to select products for purchase, and developers can judge the reliability of a product and efficiently manage the development process.

In most existing SRGMs, the mean value function is considered to follow a non-homogeneous Poisson process (NHPP). Each model has a unique mean value function that considers the failure intensity, detection rate, number of remaining failures, and various environments (e.g., development, testing, and operation). In particular, the form and parameters of the mean value function (m(t)) are significantly affected by the environments and assumptions made.

Previous studies have considered various SRGMs when discussing software reliability. The model proposed by Goel and Okumoto [1] is the most fundamental approach used in research on SRGMs and has become the focal point of various studies. Yamada et al. [2] proposed a model using an S-shaped curve. There have also been studies on models that reflect not only the shape of the function, but also various environments. Pham et al. [3,4] introduced models that consider an error introduction rate that follows an exponential function and a generalized SRGM that reflects the testing coverage. Song et al. [5] proposed a model for considering the test coverage. Pham [6] discussed a model that considers a logistic time-dependent fault-detection rate function. In addition, Pham [7] introduced a V-tub-shaped rate function. Pradhan et al. [8] presented an improved model for a generalized inflection S-shaped testing effort function. Erto et al. [9] also developed a generalized inflection S-shaped SRGM. Saxena et al. [10] proposed a SRGM that assumes imperfect debugging by a two-step process that considers fault observation and fault removal. Haque [11] presented a new Software Reliability Growth Model (SRGM) with the structure of a Logistic Growth Model in which the fault detection rate increases as the test department's skill improves as the test progresses. Nafreen et al. [12] developed a SRGM with a bathtub-shaped fault detection rate function and proposed the conditional maximization algorithms to fit the models.

There are also SRGM studies that introduce different assumptions from previous studies. For example, the controlled test environment and actual operating environment of the consumers can vary significantly. Song et al. [13,14] discussed models that consider the uncertainty of the operating environments using new parameters as random variables. Zaitseva et al. [15] discussed a method to evaluate reliability analysis considering various types of uncertainty. Most SRGMs assume that failures are independent. However, a failure is occasionally dependent. Lee et al. [16] proposed a method for efficiently judging the software reliability using the model development, assuming a dependent failure and applying a sequential probability ratio test. Kim et al. [17] developed a new model that assumes the presence of dependent defects.

Recently, research on software reliability has been conducted with various approaches. Saxena et al. [18], Kumar et al. [19], and Garg et al. [20] developed criteria for judging goodness of fit for SRGMs. The criteria are calculated based on the combination of the entropy principle and various existing criteria. Most studies discussed the individual reliability of software and hardware. Yaghoobi [21] also developed two multi-criteria decision-making methods for comparison of SRGMs. Zhu [22] introduced complex reliability that considers both hardware and software, and proposed maintenance policies applicable to these systems. Recently, as the use of open-source software and cloud services increases, research on their reliability is being actively conducted [23–25]. Finally, several software reliability studies have recently been conducted using machine and deep learning [26–29].

In this study, we propose a new SRGM with two assumptions that software failures are dependent and the operating environments of the software are uncertain. The superiority of the proposed model is explained by the criteria for goodness of fit. The goodness of fit is used to determine which model fits the failure dataset for a given software product. A model with a good fit can predict the number of failures well and provide meaningful information for evaluating product reliability and establishing release policies. In Section 2, the concept of system reliability is provided. Section 3 describes the existing NHPP SRGM and the proposed model. Section 4 introduces the criteria and data used in the experiments, then discusses the results. Finally, Section 5 provides some concluding remarks.

2. Software Reliability Growth Model

2.1. Poisson Processes

The counting process is represented by $N(t)$, i.e., the total number of events up to time t. The counting process satisfies the following conditions [30]:

(1) $N(t) \geq 0$.
(2) $N(t)$ is an integer.
(3) If $s \leq t$, $N(s) \leq N(t)$.
(4) When $s \leq t$, $N(t) - N(s)$ represents the number of events within time interval $(s, t]$.

The homogeneous Poisson process (HPP) is given by

$$\Pr\{N(t) = n\} = \frac{\lambda t^n}{n!} e^{-\lambda t},\ n = 0,\ 1,\ 2,\ \ldots \quad (1)$$

where $N(t)$ is the number of failures up to time t, and λ is a constant. Thus, the failure rate λ does not change over time t. The number of failures $N(t)$ has a Poisson distribution with the mean λt.

Most software reliability models assume that $N(t)$ follows a non-homogeneous Poisson process (NHPP). When $N(t)$ follows a/n NHPP, it is given by

$$\Pr\{N(t) = n\} = \frac{[m(t)]^n}{n!} e^{-m(t)},\ n = 0,\ 1,\ 2,\ \ldots \quad (2)$$

where m(t) is used instead of the constant λ, unlike with a HPP. This means that the failure rate $m(t)$ depends on time t. Here, m(t) can be written as follows:

$$m(t) = E[N(t)] = \int_0^t \lambda(s)ds. \quad (3)$$

2.2. Reliability Function

Using m(t), the reliability function based on a NHPP can be expressed as follows [30]: The reliability function $R(t)$ is defined as the probability that there will be no failures within the time interval $(0,\ t)$, and is given by

$$R(t) = P\{N(t) = 0\} = e^{-m(t)} \quad (4)$$

Equation (4) indicates the probability that a software error will not occur within the interval [0,t]. If t + x is given, the software reliability can be expressed as a conditional probability R(x | t), as in Equation (5).

$$R(x|t) = P\{N(t+x) - N(t) = 0\} = e^{-[m(t+x)-m(t)]} \quad (5)$$

Here, R(x | t) is the probability that a software error will not occur within the interval [t, t + x], where $t \geq 0$ and $x > 0$. The density function of x is given by

$$f(x) = \lambda(t+x)e^{-[m(t+x)-m(t)]} \quad (6)$$

where $\lambda(x) = \frac{\partial}{\partial x}[m(x)]$.

3. NHPP Software Reliability Growth Model (SRGM)

3.1. Model Formulation

(1) In general, $m(t)$ is obtained from a differential equation as follows [2,31]:

$$\frac{m(t)}{dt} = b(t)[a(t) - m(t)] \quad (7)$$

where $a(t)$ is defined as the expected total number of initial failures plus the number of newly introduced errors that remain until testing, and $b(t)$ is defined as the rate of detected failures per fault.

(2) The mean value function of the model when considering uncertainty of the operating environment can be obtained from the following differential equation [32]:

$$\frac{m(t)}{dt} = \eta b(t)[N - m(t)] \quad (8)$$

where $b(t)$ is the fault detection rate function. Let η represent the uncertainty of the operating environment. Here, η is a random variable that follows gamma distribution $g(x)$ with parameters α and β, which is given by

$$g(x) = \frac{\beta^{\alpha} x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}. \tag{9}$$

(3) The mean value function of the model when considering dependent failures can be obtained from the following differential equation [16,17]:

$$\frac{m(t)}{dt} = b(t)[a(t) - m(t)]m(t) \tag{10}$$

where $a(t)$ is defined as the expected total number of failures that remain prior to testing, and $b(t)$ is the defined detection rate of failures per fault.

3.2. Proposed Model

Most models assume independent failures (see Section 3.3); however, this paper assumes dependent failures. For example, if an error occurs in a class in a certain program code, it may occur in chains in other classes that refer to this class. In this case, the failures are dependent, and the model proposed in this study also assumes the same type of failure.

We propose a model that considers an uncertain operating environment and dependent failures. We consider the second type owing to the fact that we need to assume an uncertain operating environment. When $a(t) = N$ and $b(t) = \frac{b^2 t}{1+bt}$ are given, $m(t)$ can be obtained from Equation (8) as follows:

$$m(t) = N\left(1 - \frac{\beta}{\alpha + \int b(s)ds}\right)^{\alpha} = N\left(1 - \frac{\beta}{\alpha + \int_0^t \frac{b^2 s}{1+bs} ds}\right)^{\alpha} \tag{11}$$

$$= N\left(1 - \frac{\beta}{\alpha + bt - \log(bt+1)}\right)^{\alpha} \tag{12}$$

where b, α, β, and N are parameters of the proposed model. The parameters α and β in Equation (12) are the same as those in Equation (9). Therefore, parameters α and β of $m(t)$ are dependent on the probability distribution of η, which indicates uncertain operating environments. This relationship leads to an assumption of dependent flaws in the model.

3.3. Existing NHPP Models

Table 1 summarizes the mean value functions for the existing NHPP models that are well known for their good performance and the proposed model. Dependent failure model 1 (DPF1) and dependent failure model 2 (DPF2) are models that assume dependent failures, whereas the other models assume independent failures. VTUB assumes the uncertain operating environments. In addition, the proposed model assumes two factors, the dependent failures and the uncertain operating environments. The parameters of each model can be estimated in various ways, including a Bayesian estimation, maximum likelihood estimation (MLE), and least square estimation (LSE). However, a Bayesian estimation is difficult to achieve owing to a lack of prior information, and a MLE estimation is occasionally difficult to apply because of the complexity of the mean value function of the model. Therefore, in this study, the parameters are estimated using the LSE method, which is discussed in Section 4.3.

Table 1. Mean value functions for existing NHPP models and the proposed approach.

No.	Model	$\mathbf{m(t)}$
1	DPF1 [16]	$\frac{a}{1+\left(\frac{a}{h}\left(\frac{b+c}{c+(b\exp(bt))}\right)^{\frac{a}{b}}\right)}$
2	DPF2 [17]	$\frac{a}{1+\left(\frac{a}{h}\left(\frac{1+c}{c+\exp(bt)}\right)^{a}\right)}$
3	DS [33]	$a(1-(1+bt)\exp(-bt))$
4	GO [1]	$a(1-\exp(-bt))$
5	IS [2]	$\frac{a(1-\exp(-bt))}{1+\beta\exp(-bt)}$
6	YID [33]	$a(1-\exp(-bt))\left(1-\frac{\alpha}{b}\right)+\alpha at$
7	PNZ [31]	$\frac{a(1-\exp(-bt))\left(1-\frac{\alpha}{b}\right)+\alpha at}{1+(\beta\exp(-bt))}$
8	PZ [3]	$\frac{((c+a)(1-\exp(-bt)))-\left(\frac{ab}{b-\alpha}(\exp(-at)-\exp(-bt))\right)}{1+(\beta\exp(-bt))}$
9	TC [34]	$N\left(1-\left(\frac{\beta}{\beta+(at)^{b}}\right)^{\alpha}\right)$
10	VTUB [32]	$N\left(1-\left(\frac{\beta}{\beta+\left(a^{t^{b}}\right)-1}\right)^{\alpha}\right)$
11	NEW	$N\left(1-\frac{\beta}{\alpha+bt-\log(bt+1)}\right)^{\alpha}$

4. Numerical Example

4.1. Datasets

We used two datasets to compare the goodness of fit of the model. Dataset 1, shown in Table 2, was collected from an IBM software package and has 40,000 lines of code [35]. The software was used to enter the data, and the time unit is in days. There is a total of 46 failures collected within the 21-day period.

Table 2. Dataset 1.

Time	Cumulative Failures	Time	Cumulative Failures	Time	Cumulative Failures
1	2	8	12	15	31
2	3	9	19	16	37
3	4	10	21	17	38
4	5	11	22	18	41
5	7	12	24	19	42
6	9	13	26	20	45
7	11	14	30	21	46

Dataset 2 in Table 3 was collected from the ABC online communication system in 2000 [30]. The dataset was observed over a 12-week period, and the time unit is in weeks. A total of 55 failures were observed within the 12-week period.

Table 3. Dataset 2.

Time	Failures	Cumulative Failures	Time	Failures	Cumulative Failures
1	10	10	7	4	40
2	2	12	8	3	43
3	4	16	9	1	44
4	6	22	10	6	50
5	6	28	11	1	51
6	8	36	12	4	55

4.2. Criteria

Various criteria have been proposed to discuss the goodness of fit of a model. Table 4 shows the criteria used to compare the goodness of fit for the SRGMs, which are listed in Table 1.

In Table 4, n is defined as the number of data, and m is defined as the number of parameters in the model (mean value function). The predicted value which means number of failures based on $m(t)$ is presented as $\hat{m}(t_i)$, and the actual data are presented as y_i.

Most of the criteria mentioned measure the distance (or error) between predicted value and actual values. Therefore, it can be said that the shorter the distance, the better the mean value function of the model predicts the number of failures in the dataset.

The MSE measures the distance of predicted value by a model from the actual data with the consideration of the number parameters in the model and the number of data. The PRR measures the distance of predicted value from the actual data with the consideration the predicted value by a model. The PP measures the distance of predicted value from the actual data with the consideration the actual data. The SAE measures the sum of absolute error, which means the distance between the predicted value and the actual data.

Table 4. Criteria.

No	Criteria	
1	Mean square error (MSE) [30]	$\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{n-m}$
2	Predictive ratio risk (PRR) [30]	$\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{\hat{m}(t_i)}$
3	Predictive power (PP) [30]	$\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{y_i}$
4	Sum of absolute error (SAE) [36]	$\sum_{i=1}^{n}\lvert\hat{m}(t_i)-y_i\rvert$
5	R-square (R^2) [37]	$1-\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y}_i)^2}$
6	Akaike's information criterion (AIC) [38]	$-2\log L+2m$
7	Bayesian information criterion (BIC) [39,40]	$-2\log L+m\log n$
8	Bias [41]	$\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)}{n}$
9	Predicted relative variation (PRV) [42]	$\sqrt{\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i-\text{Bias})^2}{n-1}}$
10	Root mean square prediction error (RMSPE) [42]	$\sqrt{\text{Bias}^2+\text{PRV}^2}$
11	Mean absolute error (MAE) [43]	$\frac{\sum_{i=1}^{n}\lvert\hat{m}(t_i)-y_i\rvert}{n-m}$
12	Mean error of prediction (MEOP) [43]	$\frac{\sum_{i=1}^{n}\lvert\hat{m}(t_i)-y_i\rvert}{n-m+1}$
13	Theil statistic (TS) [43]	$100\sqrt{\frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{\sum_{i=1}^{n}y_i^2}}\%$

The R^2 (coefficient of determination) is a measure of the fit of the regression equation. It represents the proportion of the regression sum of squares (SSR) to the total sum of squares (SST) of the model. The closer the value is to 0, the better the fit of the model.

The AIC measures a model's ability to maximize a likelihood function (L). In general, the greater the number of parameters, the better the fit of the model. AIC mitigates the phenomenon by penalizing the number of parameters. In addition, $\log L$ of the AIC is given by

$$L = \prod_{i=1}^{n} \frac{(\hat{m}(t_i) - \hat{m}(t_{i-1}))^{y_i - y_{i-1}}}{(y_i - y_{i-1})!} \tag{13}$$

$$\log L = \sum_{i=1}^{n} \{ (y_i - y_{i-1}) \log(\hat{m}(t_i) - \hat{m}(t_{i-1})) - (\hat{m}(t_i) - \hat{m}(t_{i-1})) - \log((y_i - y_{i-1})!) \} \tag{14}$$

The BIC estimates the approximate value of the posterior probability of the model. In the second term of BIC, the number of parameters is penalized like AIC. Also, the term depends on the number of data.

The Bias measures the average distance between the predicted value and the actual data. The closer the value is to 0, the better the fit of the model. The Bias is also used to calculate PRV and RMSPE. The PRV calculates the standard deviation of the prediction bias, so the smaller the value, the better the fit of the model. The PRV is also called Variation or Variance. The RMSPE calculates the root mean square prediction error, which means the closeness of the predicted value with actual data using Bias and PRV. The MAE calculates the mean absolute error between the predicted value and the actual data. The MEOP calculates the sum of absolute error (SAE) with the consideration the predicted value by a model. The TS calculates the average deviation percentage over all data observation periods with regard to the actual data. The closer the value is to 0, the better the fit of the model.

To summarize the 13 criteria, the closer the value of R^2 is to 1, the closer the predicted value of the model is to the actual value. The closer the values of Bias and TS are to 0, the closer the predicted value of the model is to the actual value. For the other criteria—MSE, PRR, PP, SAE, AIC, BIC, PRV, RMSPE, MAE, and MEOP—the smaller the value, the closer the predicted value of the model is to the actual value, relative to other models on the same dataset.

4.3. Results

The estimated parameters of the models are listed in Tables 5 and 6 using a least squares estimation (LSE).

Tables 7 and 8 show the criteria obtained using the estimated parameters in Tables 5 and 6, the best value for each criteria is shown in bold.

In Table 7, the proposed model shows the smallest MSE, the smallest PRR, the smallest PP, the smallest SAE, the largest R^2, the closest-to-zero Bias, the smallest PRV, the smallest RMSPE, the smallest MAE, the smallest MEOP and the closest-to-zero TS at 1.3805, 0.0772, 0.0763, 15.9817, 0.9948, −0.0021, 1.0832, 1.0832, 0.9401, 0.8879 and 3.9233. Though the AIC and BIC are not the smallest values among all models, the proposed model displays the third= smallest AIC at 78.0915. In particular, the PRR value of DPF2 is the second-smallest value at 0.3333, but the values of other criteria are not good. The overall criteria values of VTUB are not good. When looking at the values of the overall criteria, it can be seen that the goodness of fit of the proposed model is excellent.

Table 5. Estimated parameters of models for dataset 1.

Model	$\hat{a}$	$\hat{b}$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{N}$	$\hat{c}$	$\hat{h}$
DPF1	51.350	0.001	-	-	-	0.216	2.659
DPF2	51.350	0.005	-	-	-	0.076	2.629
DS	77.253	0.0966	-	-	-	-	-
GO	14,264.000	0.0002	-	-	-	-	-
IS	58.943	0.170	-	8.386	-	-	-
YID	1.491	0.3068	1.7457	-	-	-	-
PNZ	29.875	0.192	0.045	4.900	-	-	-
PZ	59.316	0.1682	128.1029	8.2581	-	0.0005	-
TC	0.0191	1.567	839.1540	221.1735	78.7859	-	-
VTUB	1.9701	0.6892	0.2928	19.8529	87.2519	-	-
NEW	-	0.2470	2.355	1.968	126.140	-	-

Table 6. Estimated parameters of models for dataset 2.

Model	$\hat{a}$	$\hat{b}$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{N}$	$\hat{c}$	$\hat{h}$
DPF1	55.893	0.004	-	-	-	0.548	7.274
DPF2	56.058	0.008	-	-	-	0.093	7.195
DS	57.478	0.344	-	-	-	-	-
GO	94.344	0.0733	-	-	-	-	-
IS	65.781	0.206	-	1.293	-	-	-
YID	5.749	52.415	0.756	-	-	-	-
PNZ	64.922	0.208	0.001	1.286	-	-	-
PZ	7.617	0.210	0.005	1.321	-	64.992	-
TC	0.005	1.075	2001.000	84.681	80.373	-	-
VTUB	5.0693	1.793	0.0181	0.0004	57.6685	-	-
NEW	-	0.316	1.326	1.142	91.500	-	-

Table 7. Criteria values for model comparison of dataset 1.

Model	MSE	PRR	PP	SAE	R^2	AIC	BIC	Bias	PRV	RMSPE	MAE	MEOP	TS (%)
DPF1	2.0159	0.3434	0.6026	20.5585	0.9924	78.8231	83.0012	0.1065	1.3044	1.3088	1.2093	1.1421	4.7409
DPF2	2.0055	0.3333	0.5753	20.4606	0.9924	78.7999	82.9780	0.1006	1.3016	1.3054	1.2036	1.1367	4.7287
DS	1.6365	26.3229	1.2081	21.0349	0.9931	78.1184	80.2075	−0.2326	1.2239	1.2458	1.1071	1.0517	4.5159
GO	6.6010	0.8080	1.8602	42.5436	0.9721	77.3351	79.4242	0.8913	2.3317	2.4962	2.2391	2.1272	9.0696
IS	1.3952	0.6991	0.3003	17.4495	0.9944	76.6925	79.8261	−0.0325	1.1201	1.1205	0.9694	0.9184	4.0584
YID	1.7008	3.0656	0.6081	21.0606	0.9932	78.6624	81.7960	−0.0548	1.2360	1.2372	1.1700	1.1085	4.4810
PNZ	1.4844	0.9578	0.3521	17.9074	0.9944	78.9419	83.1200	−0.0504	1.1221	1.1232	1.0534	0.9949	4.0683
PZ	1.5697	0.6947	0.3003	17.5203	0.9944	80.7116	85.9342	−0.0272	1.1203	1.1206	1.0950	1.0306	4.0586
TC	1.7268	5.9590	0.7580	19.4342	0.9939	82.5660	87.7886	−0.0548	1.1740	1.1753	1.2146	1.1432	4.2568
VTUB	1.5438	0.5610	0.2699	17.5100	0.9945	80.6019	85.8245	−0.0425	1.1105	1.1113	1.0944	1.0300	4.0250
NEW	1.3805	0.0772	0.0763	15.9817	0.9948	78.0915	82.2696	−0.0021	1.0832	1.0832	0.9401	0.8879	3.9233

In Table 8, the proposed model shows the smallest MSE and the smallest AIC at 2.7776 and 57.2427, respectively. Though other criteria are not the best values among all models, the proposed model shows the second-largest R^2, the second-smallest BIC, the second-closest-to-zero Bias, the second-smallest PRV, the second-smallest RMSPE, the second-closest-to-zero TS at 0.9920, 59.1823, −0.0070, 1.4213, 1.4213 and 3.6615, respectively. In particular, DPF1 shows the smallest PRR, the smallest PP, the smallest MAE and the smallest MEOP at 0.0276, 0.0270, 1.5924 and 1.4154, respectively. The VTUB shows the largest R^2, the smallest SAE, the smallest PRV, the smallest RMSPE and the closest-to-zero TS at 0.9925, 12.6030, 1.3688, 1.3704 and 3.5306, respectively. When looking at the values of the overall criteria, it can be seen that the goodness of fit of the proposed model is excellent.

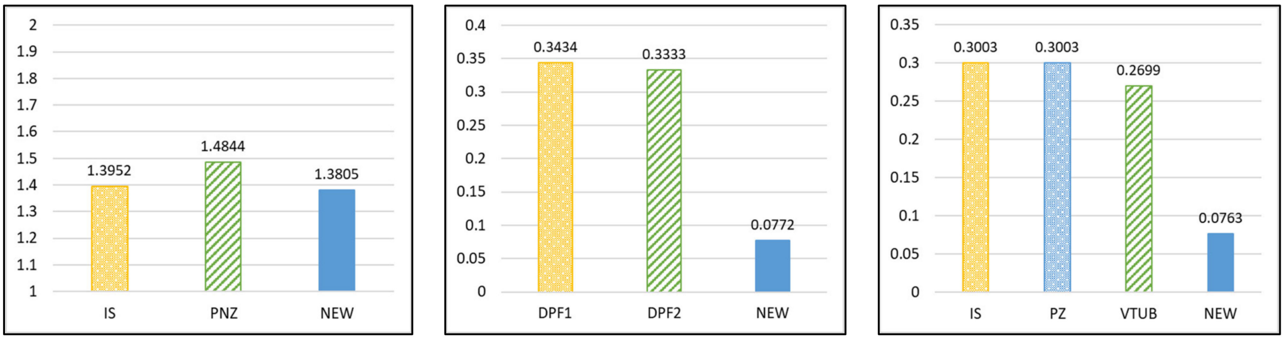
What is especially important about this result is that the fit of the proposed model is much better than DPF1, DPF2, and VTUB. In other words, DPF1 and DPF2 assume dependent failures, and VTUB assumes uncertain operating environments. It can be seen that the proposed model takes both assumptions into account and fits the dataset much better. Therefore, it is reasonable to predict the number of failures and conduct software reliability studies (reliability evaluation, release policy, etc.) based on the proposed model rather than DPF1, DPF2, or VTUB.

Table 8. Criteria values for model comparison of dataset 2.

Model	MSE	PRR	PP	SAE	R^2	AIC	BIC	Bias	PRV	RMSPE	MAE	MEOP	TS (%)
DPF1	2.8201	0.0276	0.0270	12.7389	0.9919	58.6958	60.6354	0.0103	1.4321	1.4321	1.5924	1.4154	3.6893
DPF2	2.7946	0.0283	0.0275	12.7515	0.9919	58.6274	60.5670	0.0078	1.4256	1.4256	1.5939	1.4168	3.6726
DS	8.2096	7.3679	0.6177	20.9540	0.9704	69.6251	70.5950	−0.7059	2.6305	2.7236	2.0954	1.9049	7.0377
GO	4.0245	0.2932	0.1627	19.4170	0.9855	57.7076	58.6775	−0.0522	1.9120	1.9127	1.9417	1.7652	4.9275
IS	4.0555	0.4815	0.1905	17.0520	0.9868	60.1451	61.5998	−0.1725	1.8126	1.8208	1.8947	1.7052	4.6926
YID	7.7536	0.0893	0.1027	24.4096	0.9748	58.2593	59.7140	0.0000	2.5187	2.5187	2.7122	2.4410	6.4885
PNZ	4.5632	0.4818	0.1906	17.0566	0.9868	62.1389	64.0786	−0.1722	1.8128	1.8210	2.1321	1.8952	4.693
PZ	5.2153	0.4890	0.1917	17.0459	0.9868	64.1689	66.5934	−0.1758	1.8125	1.8210	2.4351	2.1307	4.6931
TC	5.6420	0.4307	0.1888	18.3723	0.9857	64.2519	66.6765	−0.1203	1.8906	1.8945	2.6246	2.2965	4.8813
VTUB	2.9516	0.0320	0.0296	12.6030	0.9925	59.5514	61.9760	−0.0664	1.3688	1.3704	1.8004	1.5754	3.5306
NEW	2.7776	0.0606	0.0514	14.4914	0.9920	57.2427	59.1823	−0.0070	1.4213	1.4213	1.8114	1.6102	3.6615

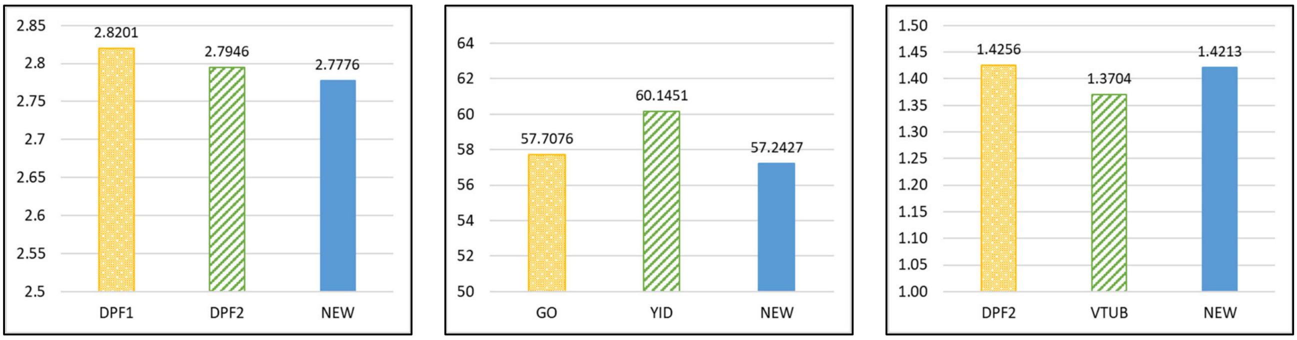
Among the 13 criteria in this paper, the most classic and well used are the MSE, PRR, and PP. Figures 1 and 2 display the three criteria values of the top three models for each dataset. For the criteria—MSE, PRR, PP, AIC, and RMSPE—the smaller the value, the closer the predicted value of the model is to the actual value, relative to other models on the same dataset.

For dataset 1 (See Figure 1), the proposed model displays the smallest MSE, the smallest PRR and the smallest PP at 1.3805, 0.0772, and 0.0763, respectively. For dataset 2 (see Figure 2), the proposed model displays the smallest MSE and the smallest AIC at 2.7776 and 57.2427, respectively. The VTUB displays the smallest RMSPE at 1.3704.



(**a**) (**b**) (**c**)

Figure 1. Three criteria values of top three models for dataset 1. (**a**) MSE; (**b**) PRR; (**c**) PP.



(**a**) (**b**) (**c**)

Figure 2. Three criteria values of top three models for dataset 2. (**a**) MSE; (**b**) AIC; (**c**) RMSPE.

Figures 3 and 4 present the 95% confidence intervals of the datasets, the formula of which is given by

$$\hat{m}(t) \pm z_{\alpha}\sqrt{\hat{m}(t)} \tag{15}$$

where z_{α} is defined as the $100(1-\alpha)/2$ percentile of the standard normal distribution [30].

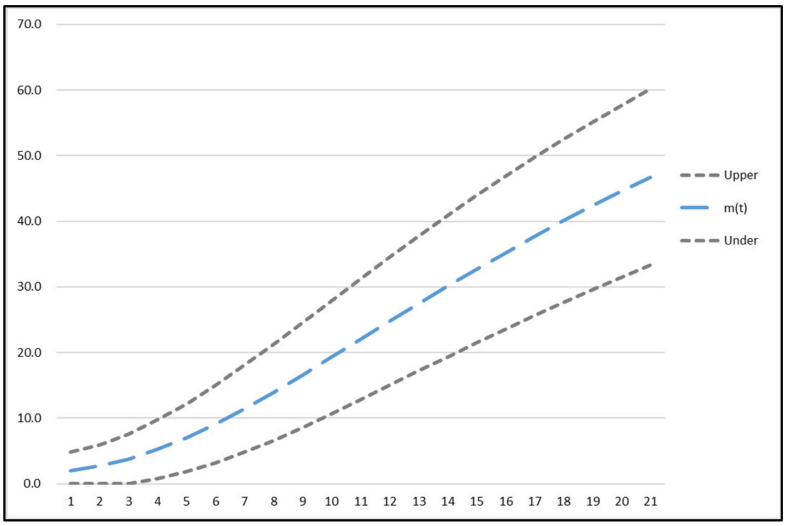


Figure 3. Confidence intervals (95%) of dataset 1.

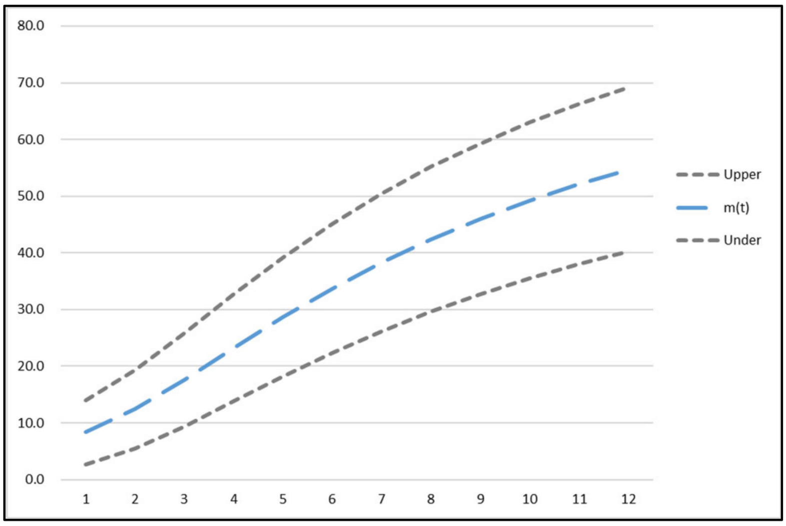


Figure 4. Confidence intervals (95%) of dataset 2.

Figures 5 and 6 show the mean value functions of the existing NHPP models and the proposed model for datasets.

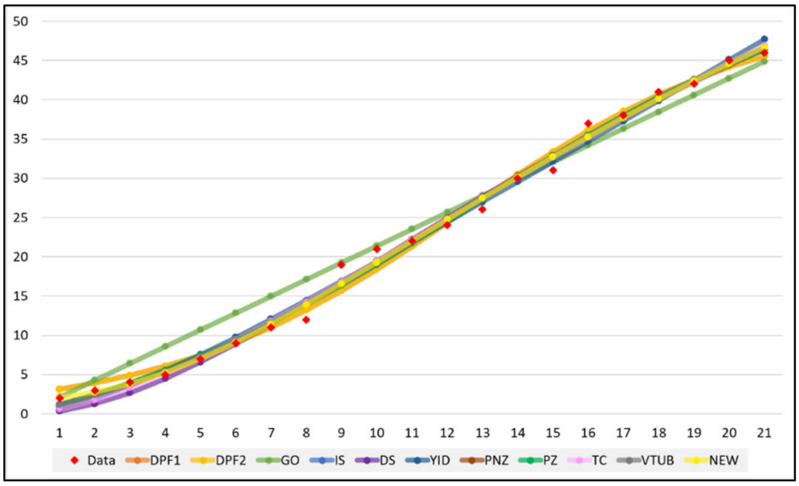


Figure 5. Mean value function of the models for dataset 1.

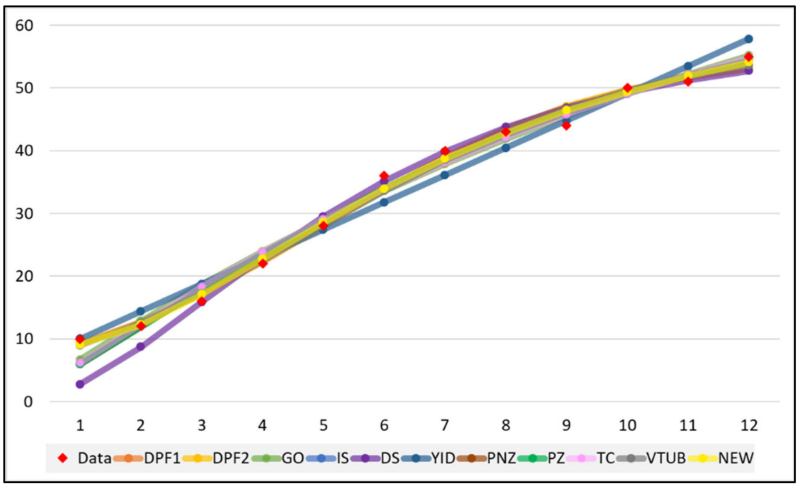


Figure 6. Mean value functions of the models for dataset 2.

5. Conclusions

Most NHPP SRGMs have been developed with stand-alone failure considerations or are environment-based (controlled). Failures can occasionally occur, and the test and operating environments used by real consumers differ. As the main idea of this study, the proposed model considers both dependent failures and uncertain operating environments. When looking at the values of the overall criteria in Tables 7 and 8, it can be seen that the goodness of fit of the proposed model is excellent. Among the 13 criteria, 11 criteria values show the best values for dataset 1. For dataset 2, 2 criteria values show the best values and 6 criteria values show the second-best values. In other words, the proposed model more closely predicts the number of failures than other models. Therefore, we conclude that it is reasonable to predict the number of failures based on the proposed model and conduct software reliability studies (reliability assessment, release policy, etc.).

6. Future Research

In the future, it is believed that various SRGMs can be developed if $a(t)$ and $b(t)$ are based on various assumptions, a dependent failure, and uncertain operating environments are taken into consideration. We will develop a model that expresses more robust dependent failures by extending the model proposed in this study. Also, we plan to convert software reliability evaluation and cost function into a single-objective function [44,45]. In addition, it can be extended to various studies [46–49]. This study can be extended into a release policy study. As the frequency of use of cloud services and open-source software is increasing, it is necessary to study the reliability of collecting datasets on them.

Author Contributions: Conceptualization, H.P.; funding acquisition, I.H.C.; software, D.H.L.; writing—original draft, D.H.L.; writing—review and editing, I.H.C. and H.P. These three authors contributed equally to the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF-2021R1F1A1048592 and 2021R1A6A3A01086716).

Institutional Review Board Statement: Not required.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository.

Acknowledgments: Many thanks to the reviewers for careful reading and valuable comments which improved the paper representation.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Goel, A.L.; Okumoto, K. Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE Trans. Reliab.* **1979**, *28*, 206–211. [CrossRef]
2. Yamada, S.; Ohba, M.; Osaki, S. S-shaped software reliability growth models and their applications. *IEEE Trans. Reliab.* **1984**, *33*, 289–292. [CrossRef]
3. Pham, H.; Zhang, X. An NHPP software reliability model and its comparison. *Int. J. Reliab. Qual. Saf. Eng.* **1997**, *4*, 269–282. [CrossRef]
4. Pham, H.; Zhang, X. NHPP software reliability and cost models with testing coverage. *Eur. J. Oper. Res.* **2003**, *145*, 443–454. [CrossRef]
5. Song, K.Y.; Chang, I.H.; Pham, H. A testing coverage model based on NHPP software reliability considering the software operating environment and the sensitivity analysis. *Mathematics* **2019**, *7*, 450. [CrossRef]
6. Pham, H. A logistic fault-dependent detection software reliability model. *J. Univers. Comput. Sci.* **2018**, *24*, 1717–1730.
7. Pham, H. Distribution function and its applications in software reliability. *Int. J. Performability Eng.* **2019**, *15*, 1306–1313. [CrossRef]
8. Pradhan, V.; Kumar, A.; Dhar, J. Enhanced growth model of software reliability with generalized inflection S-shaped testing-effort function. *J. Interdiscip. Math.* **2022**, *25*, 137–153. [CrossRef]

9. Erto, P.; Giorgio, M.; Lepore, A. The Generalized Inflection S-Shaped Software Reliability Growth Model. *IEEE Trans. Reliab.* **2020**, *69*, 228–244. [CrossRef]
10. Saxena, P.; Ram, M. Two phase software reliability growth model in the presence of imperfect debugging and error generation under fuzzy paradigm. *Math. Eng. Sci. Aerosp. (MESA)* **2022**, *13*, 777–790.
11. Haque, M.A.; Ahmad, N. An effective software reliability growth model. *Saf. Reliab.* **2021**, *40*, 209–220. [CrossRef]
12. Nafreen, M.; Fiondella, L. Software Reliability Models with Bathtub-shaped Fault Detection. In Proceedings of the 2021 Annual Reliability and Maintainability Symposium (RAMS), Orlando, FL, USA, 24–27 May 2021; pp. 1–7.
13. Song, K.Y.; Chang, I.H.; Pham, H. A three-parameter fault-detection software reliability model with the uncertainty of operating environments. *J. Syst. Sci. Syst. Eng.* **2017**, *26*, 121–132. [CrossRef]
14. Song, K.Y.; Chang, I.H.; Pham, H. NHPP software reliability model with inflection factor of the fault detection rate considering the uncertainty of software operating environments and predictive analysis malignant. *Symmetry* **2019**, *11*, 521. [CrossRef]
15. Zaitseva, E.; Levashenko, V. Construction of a Reliability Structure Function Based on Uncertain Data. *IEEE Trans. Reliab.* **2016**, *65*, 1710–1723. [CrossRef]
16. Lee, D.H.; Chang, I.H.; Pham, H. Software reliability model with dependent failures and SPRT. *Mathematics* **2020**, *8*, 1366. [CrossRef]
17. Kim, Y.S.; Song, K.Y.; Pham, H.; Chang, I.H. A software reliability model with dependent failure and optimal release time. *Symmetry* **2022**, *14*, 343. [CrossRef]
18. Saxena, P.; Kumar, V.; Ram, M. A novel CRITIC-TOPSIS approach for optimal selection of software reliability growth model (SRGM). *Qual. Reliab. Eng. Int.* **2022**, *38*, 2501–2520. [CrossRef]
19. Kumar, V.; Saxena, P.; Garg, H. Selection of optimal software reliability growth models using an integrated entropy–Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) approach. *Math. Methods Appl. Sci.* **2021**. [CrossRef]
20. Garg, R.; Raheja, S.; Garg, R.K. Decision Support System for Optimal Selection of Software Reliability Growth Models Using a Hybrid Approach. *IEEE Trans. Reliab.* **2022**, *71*, 149–161. [CrossRef]
21. Yaghoobi, T. Selection of optimal software reliability growth model using a diversity index. *Soft Comput.* **2021**, *25*, 5339–5353. [CrossRef]
22. Zhu, M. A new framework of complex system reliability with imperfect maintenance policy. *Ann. Oper. Res.* **2022**, *312*, 553–579. [CrossRef]
23. Wang, L. Architecture-Based Reliability-Sensitive Criticality Measure for Fault-Tolerance Cloud Applications. *IEEE Trans. Parallel Distrib. Syst.* **2019**, *30*, 2408–2421. [CrossRef]
24. Wang, H.; Wang, L.; Yu, Q.; Zheng, Z. Learning the Evolution Regularities for BigService-Oriented Online Reliability Prediction. *IEEE Trans. Serv. Comput.* **2019**, *12*, 398–411. [CrossRef]
25. Wang, L.; He, Q.; Gao, D.; Wan, J.; Zhang, Y. Temporal-Perturbation Aware Reliability Sensitivity Measurement for Adaptive Cloud Service Selection. *IEEE Trans. Serv. Comput.* **2022**, *15*, 2301–2313. [CrossRef]
26. Wang, J.; Zhang, C. Software reliability prediction using a deep learning model based on the RNN encoder–decoder. *Reliab. Eng. Syst. Saf.* **2018**, *170*, 73–82. [CrossRef]
27. San, K.K.; Washizaki, H.; Fukazawa, Y.; Honda, K.; Taga, M.; Matsuzaki, A. Deep Cross-Project Software Reliability Growth Model Using Project Similarity-Based Clustering. *Mathematics* **2021**, *9*, 2945. [CrossRef]
28. Li, L. Software reliability growth fault correction model based on machine learning and neural network algorithm. *Microprocess. Microsyst.* **2021**, *80*, 103538. [CrossRef]
29. Banga, M.; Bansal, A.; Singh, A. Implementation of machine learning techniques in software reliability: A framework. In Proceedings of the 2019 International Conference on Automation, Computational and Technology Management (ICACTM), London, UK, 24–26 April 2019; pp. 241–245. [CrossRef]
30. Pham, H. *System Software Reliability*; Springer: London, UK, 2006.
31. Pham, H.; Nordmann, L.; Zhang, Z. A general imperfect-software-debugging model with S-shaped fault-detection rate. *IEEE Trans. Reliab.* **1999**, *48*, 169–175. [CrossRef]
32. Pham, H. A new software reliability model with Vtub-shaped fault-detection rate and the uncertainty of operating environments. *Optimization* **2014**, *63*, 1481–1490. [CrossRef]
33. Yamada, S.; Tokuno, K.; Osaki, S. Imperfect debugging models with fault introduction rate for software reliability assessment. *Int. J. Syst. Sci.* **1992**, *23*, 2241–2252. [CrossRef]
34. Chang, I.H.; Pham, H.; Lee, S.W.; Song, K.Y. A testing-coverage software reliability model with the uncertainty of operation environments. *Int. J. Syst. Sci. Oper. Logist.* **2014**, *1*, 220–227. [CrossRef]
35. Ohba, M. Software reliability analysis models. *IBM J. Res. Dev.* **1984**, *28*, 428–443. [CrossRef]
36. Song, K.Y.; Chang, I.H.; Pham, H. A software reliability model with a Weibull fault detection rate function subject to operating environments. *Appl. Sci.* **2017**, *7*, 983. [CrossRef]
37. Li, Q.; Pham, H. A testing-coverage software reliability model considering fault removal efficiency and error generation. *PLoS ONE* **2017**, *12*, e0181524. [CrossRef]
38. Akaike, H. A new look at the statistical model identification. *IEEE Trans. Automat. Contr.* **1974**, *19*, 716–723. [CrossRef]
39. Okamura, H.; Dohi, T.; Osaki, S. Software reliability growth models with normal failure time distributions. *Reliab. Eng. Syst. Saf.* **2013**, *116*, 135–141. [CrossRef]

40. Schwarz, G. Estimating the dimension of a model. *Ann. Stat.* **1978**, *6*, 461–464. [CrossRef]
41. Zhu, M.; Pham, H. A two-phase software reliability modeling involving with software fault dependency and imperfect fault removal. *Comput. Lang. Syst. Struct.* **2018**, *53*, 27–42. [CrossRef]
42. Pillai, K.; Sukumaran Nair, V.S. A model for software development effort and cost estimation. *IEEE Trans. Softw. Eng.* **1997**, *23*, 485–497. [CrossRef]
43. Anjum, M.; Haque, M.A.; Ahmad, N. Analysis and ranking of software reliability models based on weighted criteria value. *Int. J. Inf. Technol. Comput. Sci.* **2013**, *5*, 1–14. [CrossRef]
44. Gharaei, A.; Amjadian, A.; Shavandi, A. An integrated reliable four-level supply chain with multi-stage products under shortage and stochastic constraints. *Int. J. Syst. Sci. Oper. Logist.* **2021**, 1–22. [CrossRef]
45. Askari, R.; Sebt, M.V.; Amjadian, A. A Multi-product EPQ Model for Defective Production and Inspection with Single Machine, and Operational Constraints: Stochastic Programming Approach. In *International Conference on Logistics and Supply Chain Management*; Springer: Cham, Switzerland, 2021; pp. 161–193.
46. Souza, R.L.C.; Ghasemi, A.; Saif, A.; Gharaei, A. Robust job-shop scheduling under deterministic and stochastic unavailability constraints due to preventive and corrective maintenance. *Comput. Ind. Eng.* **2022**, *168*, 108130. [CrossRef]
47. Amjadian, A.; Gharaei, A. An integrated reliable five-level closed-loop supply chain with multi-stage products under quality control and green policies: Generalised outer approximation with exact penalty. *Int. J. Syst. Sci. Oper. Logist.* **2022**, *9*, 429–449. [CrossRef]
48. Gharaei, A.; Almehdawe, E. Optimal sustainable order quantities for growing items. *J. Clean. Prod.* **2021**, *307*, 127216. [CrossRef]
49. Gharaei, A.; Diallo, C.; Venkatadri, U. Optimal economic growing quantity for reproductive farmed animals under profitable by-products and carbon emission considerations. *J. Clean. Prod.* **2022**, *374*, 133849. [CrossRef]