SRGM Decision Model Considering Cost-Reliability

Wenqian Jiang, Harbin Institute of Technology (Weihai), China
Ce Zhang, Harbin Institute of Technology (Weihai), China
Di Liu, Harbin Institute of Technology (Weihai), China
Kaiwei Liu, Harbin Institute of Technology (Weihai), China
Zhichao Sun, Harbin Institute of Technology (Weihai), China
Jianyuan Wang, Harbin Institute of Technology (Weihai), China
Zhongyin Qiu, Harbin Institute of Technology, China
Weigong Lv, Harbin Institute of Technology (Weihai), China

ABSTRACT

Aiming at the current software cost model and optimal release research, which does not fully consider the actual faults in the testing phase, a cost-reliability SRGM evaluation and selection algorithm SESABCRC is proposed. From the perspective of incomplete debugging, introducing new faults, and considering testing effort, the imperfect debugging SRGM is established. The proposed SRGM can be used to describe the testing process of the software through the actual failure data set verification and is superior to other models. Based on the proposed SRGM, the corresponding cost function is given, which explicitly considers the impact of imperfect debugging on the cost. Furthermore, an optimal release strategy is proposed when given restricted reliability target requirements and when considering the uncertainty that the actual cost may exceed the expected cost. Finally, an experimental example is given to illustrate and verify the optimal publishing problem, and parameter sensitivity analysis is carried out.

KEYWORDS

Multiple-Attribute Decision Making, Optimal Distance, Software Reliability, Software Reliability Growth Model, Testing Effort

INTRODUCTION

SRGM (Software Reliability and Growth Model) is an important mathematical tool for modeling and predicting the reliability improvement process in software testing stages (Ahmad, Bokhari, Quadri & Khan, 2008; Dohi, Matsuoka & Osaki, 2002; Zhang, Meng & Wan, 2016). Accurately modeling software reliability and predicting its possible trends are essential to determine the reliability of the entire product (Yamada, 2014; Okamura, Etani & Dohi, 2011; Okamura & Dohi, 2014; Zhang, Meng, Kao, Lü, Liu, Wan, Jiang, Cui & Liu, 2014). The description of SRGM is mainly implemented by establishing a mathematical model describing the testing process and obtaining the expression

DOI: 10.4018/IJDCF.302873

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

for the number of failures m(t) of cumulative testing. The study only from m(t), which is used to describe SRGM, is implemented from the perspective of reliability. At the same time, the test cost needs to be considered, that is, the TE (testing effort) needs to be considered. It is closely related to cost (Zhang, Meng, Kao, Lü, Liu, Wan, Jiang, Cui & Liu, 2014). TE describes the consumption of test resources, which can be represented by TEF (testing effort function). The release of software must consider not only the reliability requirements but also the cost factor (Zhang, Cui, Liu, Meng & Fu, 2014; Huang & Lyu, 2005). That is, the software release must consider the comprehensive standard of "cost-reliability". Therefore, TE has become an important branch of SRGM research and has achieved a series of results.

Counting from the G-O model (Ahmad, Khan & Rafi, 2010) in the late 1970s, SRGM research has spanned two centuries, with a research history of nearly 40 years. Hundreds of related models have been proposed. These results have enriched the connotation of the research, but at the same time, they have also brought difficulties to the evaluation and selection of SRGM. At present, the performance of SRGM is mainly evaluated from the perspective of fitting and prediction, that is, the fit of *m*(*t*) to the real historical failure data and the prediction of future failures. For example, in the evaluation of the fitting between the model and historical data, *MSE*, *variation*, *MEOP*, *TS*, *RMS-PE*, *BMMRE and R-square* (Goel & Okumoto, 1979) are often used as metric choices. Among them, the closer the R-square standard is to 1, the better, but other standards are the different (the smaller these standards, the better); *RE* is used as the model's evaluation standard for future data sets, it is still difficult to find a model that performs well in the abovementioned fitting and prediction standards. In addition, it is difficult and nonquantitative to intuitively and singly judge the performance of the models from the level of these values.

Obviously, how to select and evaluate the performance of the model and make more objective and scientific decision-making has become an important aspect of SRGM research, and some progress has been made currently. Sharma K (Zhang, Cui, Mend, Liu & Wu, 2013) performs the optimal selection of SRGM based on Euclidean distance and determines the optimal one by calculating the shortest distance between the solution of the model and the ideal solution, but this method lacks TE and more evaluation performance standards, especially the consideration of predicting RE standards. The literature (Sharma, Garg, Nagpal & Garg, 2010) proposes a method for selecting SRGM in engineering practice from the perspective of decision-making software release, which selects SRGM by evaluating the accuracy of the model predicting the remaining faults in the software. The literature (Stringfellow & Andrews, 2002) evaluates SRGM from the MSE fitting standard and self-defined predictive index (relatively balanced RE) and gives a specific implementation process of SRGM selection. Lakshmanan (Rana, Staron, Berger, Hansson, Nilsson, Toerner, Meding & Hoeglund, 2014) proposed a feed-forward neural network-based method that selects SRGM by measuring the goodness of fitting of each model, but this method lacks a description of cost loss and model prediction performance.

These studies are implemented from different angles and promote the development of SRGM selection and decision-making research. The current research has the following shortcomings:

- 1. For existing SRGM models, when they compare, they often consider only a few evaluation criteria, such as *MSE and R*-square, ignoring the important position of the prediction standard RE in the model selection process, so these methods are very likely to select a model that does not conform to the engineering software.
- 2. In terms of fitting and prediction standards, different evaluation standards have different degrees of model performance measurement, so their weights should be considered.
- Some existing SRGM models were proposed very early, and the actual factors were not fully considered at the beginning of the design. Therefore, only the failure data information was used as the modeling condition, and the proportion of TE in the modeling was not considered.

As a result, the evaluation of some SRGM models for the project is very different from the actual project.

Based on our previous research work on the reliability process with SRGM (Lakshmanan & Ramasamy, 2015), this paper studies the performance evaluation and decision-making problems of SRGM considering TE from the perspective of "cost-reliability" in view of the deficiencies of current research. The contributions of this article are as follows:

- 1. Established an SRGM performance evaluation and selection framework model that considered "cost-reliability" and covered the dual performance of fitting and prediction, and gave its formal problem description.
- 2. Establish the SRGM evaluation system structure tree to provide data structure support for the evaluation of SRGM including TE.
- 3. An SRGM evaluation and selection algorithm considering cost reliability is proposed. The SRGM model is divided into two types for processing. One type takes TE into account when modeling, and the other type does not consider TE. In this way, it is necessary to classify and deal with decision-making. The first type evaluates TE first and then evaluates the SRGM model, while the second type does not have TE participation, so you can directly evaluate the SRGM model.

Considering that TE can describe the cost and that more fitting and prediction standards are based on reliability, the SRGM decision that takes into account both cost and reliability is particularly important. This article combines the fitting information of the existing decision-making evaluation data, fully considers the *RE* value of more fitting and prediction information of the model, and supports the setting of the weight between fitting and prediction and the weight between TE and SRGM. Based on the comprehensive consideration and calculation of the decision algorithm under the hierarchical structure of the SRGM evaluation architecture tree, the performance partial order relationship among SRGMs is obtained. In the verification, the decision-making algorithm based on the ordinal preference method (TPOSIS) is used to calculate the distance between the solution of the model and the ideal solution to rank the SRGM.

CONSTRUCTION OF SRGM PERFORMANCE EVALUATION AND SELECTION FRAMEWORK CONSIDERING TE

A Formal Description of the Problem

Choosing a model with excellent performance among many SRGMs or sorting the models can be abstracted as a multi-attribute decision-making problem. As shown below, the fitting and prediction results of K SRGMs on the specified dataset DS are arranged into a MADM (multiple attribute decision matrix) form:

$$MSE \dots BMMRE RE$$

$$MADM = \begin{cases} SRGM_1 \\ SRGM_2 \\ \dots \\ SRGM_k \\ fv_{2,1} & \dots & fv_{2,L-1} \\ \dots \\ fv_{K,1} & \dots & fv_{K,L-1} \\ fv_{K,L} \\ fv_{K,L} \\ mv_{K,L} \\ mv$$

In this matrix, $fv_{i,j}$ represents the fitting result value of $SRGM_i$ on L-1 fitting standards, $pv_{i,L}$ represents the predicted result value of $SRGM_i$ on 1 predictive *standard*, $1 \le i \le K$, $1 \le j \le L$ -1. In this way,

*SRGM*_i corresponds to the decision plan; fitting and predictive indicators: *MSE*, *MEOP*, *Variation*, *RMS-PE*, *TS*, *BMMRE* and *RE* correspond to the decision attributes; the result values of these fitting and predictive indicators correspond to the attribute values of the plan.

Definition 1: Let FP_{i} , FP_{2} ..., FP_{k} denote K models to be decided, which can be either SRGM or TE. The evaluation of the model in O FDSs (Failure Data Set) includes two aspects: fitting and prediction. There are L-1 fitting standards and one prediction standard (*RE*). The fitting and prediction result vector of the i-th model on the *j*-th FDS can be expressed as:

$$FP_{i,j} = \Big[fv_{i,1}, fv_{i,2}...fv_{i,L-1}, pv_{i,L} \Big], 1 \le i \le K, 1 \le j \le O$$

In this expression, $fv_{i,n}$ represents the result value of the *i*-th model on the nth fitting standard, where $1 \le n \le L-1$, and $pv_{i,1}$ represents the corresponding prediction result value.

- **Definition 2:** Let the set of FDS be represented as DSSet, $DSSet=\{FDS_i|1 \le i \le 0\}$, where FDS i can be described by the following four-tuple: <TYPE, TIME, $FAILURE_NUM$, $TE_VALUE>$, where TYPE represents the data set type, Including two types of data sets: data sets containing TE and data sets not containing TE; TIME represents the recording time of the number of failures, which can be recorded by CPU time, calendar time, etc., mostly in weeks; corresponding to TIME The number of failures is represented by $FAILURE_NUM$, which usually shows an increasing trend; TE_VALUE represents the record value of TE consumption.
- **Definition 3:** Set the fitting and prediction weights to W_{jii} and W_{pre} , respectively, satisfying the condition $W_{fit} + W_{pre} = 1$. W_{fit} includes two types of fitting standards, namely, M key standards: $CC = \{p_i | 1 \le i \le M\}$ and N nonkey standards: $NCC = \{p_i | 1 \le j \le N\}$, which meet the following conditions:

$$\sum_{i=1}^{M} w_{cci} + \sum_{j=1}^{N} w_{ncj} = 1 \sum_{i=1}^{M} w_{cci} + \sum_{j=1}^{N} w_{ncj} = 1$$
, M+N=L-1

Obviously, the weight of key evaluation indicators is higher than that of nonkey standards, that is, the following conditions are met:

 $\forall w_{_{cci}} \ in \ CC > \forall w_{_{nci}} \ in \ NCC \ \forall w_{_{cci}} \ in \ CC > \forall w_{_{nci}} \ in \ NCC$

The fitting and prediction weight vector W can be expressed as:

$$\begin{split} W = & \left[W_{_{fit}}, w_{_{cc1}}, ..., w_{_{cci}} ... w_{_{ncM}}, w_{_{nc1}}, ..., w_{_{ncj}}, w_{_{ncN}}, W_{_{pre}} \right] \\ & 1 \leq i \leq M, 1 \leq j \leq N \end{split}$$

In this way, the weights that cover all the criteria satisfy the following normalization relationship:

$$W_{fit}\left(\sum_{i=1}^{M} w_{cci} + \sum_{j=1}^{N} w_{ncj}\right) + W_{pre} = \left(\sum_{i=1}^{M} \left(W_{fit} \times w_{cci}\right) + \sum_{j=1}^{N} \left(W_{fit} \times w_{ncj}\right)\right) + W_{pre} = 1$$
(2)

Standardization of Fitting and Prediction Results

In the fitting standard, for the *R*-square indicator, the closer it is to 1, the better the model performance. Contrary to the *R*-square indicator, the smaller the other indicators, the better. Unlike the *RE* standard, which has only one prediction, there are multiple substands for the fitting standard, some of which are widely used. To this end, we set a total of M+N+1 standards for SRGM performance evaluation, including *M* commonly used fitting standards, *N* less commonly used fitting standards, and 1 *RE* prediction standard. In this way, the initial evaluation matrix DM composed of *K* SRGM weight vectors can be obtained, as shown in (3):

$$DM = \begin{bmatrix} fv_{1,1} & \dots & fv_{1,M+N} & pv_{1,M+N+1} \\ fv_{2,1} & \dots & fv_{2,M+N} & pv_{2,M+N+1} \\ \dots & \dots & \dots & \dots \\ fv_{K,1} & \dots & fv_{K,M+N} & pv_{K,M+N+1} \end{bmatrix}_{K \times (M+N+1)}$$
(3)

In this matrix, each row has a total of M+N+1 elements, the first M elements $fv_{i,j}$, $1 \le i \le K$, $1 \le j \le M$ are the key fitting standard values of the model, the last element $pv_{i,M}+N+1$, $1 \le i \le K$ is the prediction standard value, and the middle N elements $fv_{i,j}$, $1 \le i \le K$, $M+1 \le j \le M+N$ are the noncritical fitting standard values.

To obtain a uniform positive result and facilitate subsequent calculations, it is necessary to use conversion rules to convert the original fitting and prediction results into normalized data with positive growth in the interval [0, 1]. Take the conversion form of formula (4) for different fitting standards:

$$mfv_{i,j} = \begin{cases} \left(\frac{fv_{j}^{\max} - fv_{i,j}}{fv_{j}^{\max} - fv_{j}^{\min}}\right), \\ fv_{j}^{\max} = \max v_{i,j}, fv_{j}^{\min} = \min fv_{i,j}, \\ 1 \le j \le M + N, for \ MSE, MEOP, \\ Variation, RMS - PE, TS, BMMRE \\ \left(1 - \frac{|1 - fv_{i,j}|}{|1 - fv_{i,j}|_{j}^{\max}}\right), |1 - fv_{i,j}|_{j}^{\max} = \max |1 - fv_{i,j}|, \\ 1 \le j \le M + N, for \ R - square \end{cases}$$

For prediction evaluation, the faster the prediction standard *RE* curve approaches the 0 standard horizontal line, the better the prediction performance of the model. Different from the fitting standard that can be quantitatively compared, the prediction curve mainly uses the observation method to distinguish the prediction performance difference of each model and has certain subjective fuzzy information. For this reason, for the predictive performance of the model, it is possible to give a ranking of each other's superiority and inferiority and convert it into a quantitative performance comparison value with the help of uncertainty methods such as AHP and information entropy. For this reason, it is necessary to quantify the qualitative curve results (Zhang, Cui, Liu, Meng & Wu, 2015; Saaty, 1990), as shown in Table 1.

(4)

Volume 14 • Issue 2

Table 1. Classification table

Description	Worst	Worse	Worse Fair		Best
quantized value	1	3	5	7	9

In this way, the *RE* prediction value pv_iM+N+1 , $1 \le i \le K$ in the last column of DM can be taken from the values in Table 1 according to the actual situation, and then the numerical description of the prediction results can be obtained:

$$\begin{split} mpv_{i,M+N+1} &= mfv_{i,M+N+1} \\ &= \frac{pv_{i,j}}{pv_j^{\max}}, where \ pv_j^{\max} \\ &= \max_{1 \leq i \leq K} pv_{i,j}, for \ RE \\ &\stackrel{1 \leq i \leq K}{\longrightarrow} \end{split}$$
(5)

To date, all elements in the DM have been standardized into a form in which the greater the positiveness is, the better the effect:

$$v_{i,j} = \frac{mfv_{i,j}}{\sqrt{\sum_{i=1}^{K} mfv_{i,j}^2}}, 1 \le i \le K, 1 \le j \le M + N$$
(6)

In this way, the standardized SDM= $[v_{i,i}]_{K \times (M+N+I)}$ is obtained.

Thus far, substituting the weight vector in formula (2) into SDM, the standardized evaluation matrix WSDM with weight can be obtained:

$$\begin{split} WSDM &= \\ \begin{bmatrix} v_{1,1} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} & \dots & v_{1,M} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,1} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} & \dots & v_{2,M} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,M+1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & v_{2,M+N} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,M+1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & v_{2,M+N} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,M+1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & v_{2,M+N} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,M+1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & v_{2,M+N} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{2,M+1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & \dots & \dots \\ v_{K,1} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} & \dots & v_{K,M} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} \\ v_{K,1} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} & \dots & v_{K,M} \times \begin{pmatrix} W_{fit} \times w_{cc1} \end{pmatrix} \\ v_{K,1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & \dots & \dots \\ v_{K,1} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} & \dots & v_{K,M+N} \times \begin{pmatrix} W_{fit} \times w_{ncn} \end{pmatrix} \\ v_{K,N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \times \begin{pmatrix} W_{fit} \times W_{N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \times \begin{pmatrix} W_{K,N+N+1} \end{pmatrix} \\ v_{K,N+N+1} \end{pmatrix} \\ v_{K,N+1} \end{pmatrix}$$

TE-WSDM and SRGM-WSDM are evaluation matrices for TE and SRGM, respectively, on the fitting and prediction standards. All elements in the matrix were preprocessed to obtain normalized dimensionless data.

SRGM Evaluation System Structure Tree

For the evaluation of SRGM considering TE, the fitting and prediction performance evaluation of TE should be performed first, and then the evaluation of SRGM should be performed. In this way, considering this order, the decision problem can be abstracted as a tree, as shown in Figure 1.





The top root node in Figure 1 indicates that the tree is used to evaluate SRGM, and the second layer nodes indicate different SRGMs, which can be divided into two categories, namely, SRGM without TE and SRGM with TE. Branches of a tree at the bottom layer represent multiple fitting standard nodes and a prediction node. SRGM_k in Figure 1 represents an SRGM node that does not consider TE, and SRGM1 represents an SRGM node that considers TE. It is easy to see that the evaluation of SRGMK only needs to evaluate the fit and prediction in Figure 1; in contrast, SRGM1 needs to be implemented by comprehensive evaluation of part , and part f in the figure.

The leaf node in Figure 1 can be represented by a five-tuple, namely, *<TYPE*, *ID*, *FAID*, *FIT_INFO*, *PRE_INFO>*, where:

- 1. TYPE means TE or SRGM.
- 2. *ID* is the unique number of the node.
- 3. If the *TYPE* of the node is *TE* type, there is a parent node SRGM, and *FAID* is used to represent the parent node; otherwise, it is empty.
- 4. FIT_INFO is the fitting node corresponding to TYPE, which can be described by a two-tuple, namely, *<CRITERIA*, *FIT_VALUE>*, *CRITERIA* represents the fitting standard, which can be *MSE*, *MEOP*, *Variation*, *RMS-PE*, *R-square*, *TS or BMMRE*, *FIT_VALUE* is the standard value of *CRITERIA*.
- 5. *PRE_INFO* is the prediction node corresponding to TYPE, which can be described by a twotuple, namely, *<RE,PRE_VALUE>*, RE represents the prediction standard, and *PRE_VALUE* is the corresponding prediction result value.

For the middle "TE evaluation" node and SRGM node in Figure 1, a triplet is used to represent *<TYPE, FIT_VALUE,PRE_VALUE>*, which, respectively represent TE or SRGM, as well as the fitting standard value and the prediction standard value. From the perspective of unity, for the two-tuple*<CRITERIA,FIT_VALUE>* in FIT_INFO in the abovementioned triples *<TYPE,FIT_INFO,PRE_INFO>*, if CRITERIA is FIT, it means an intermediate node. For example, *<TE,FIT,0.78,RE,0.65>* means that the fitted value of a TE node in the tree in Figure 1 is 0.78, and the predicted value is 0.65.

In addition, the abovementioned fitting standard values are all dimensionless values within (0,1); the prediction value can be obtained from the *RE* prediction curve using a suitable algorithm strategy, which is omitted here.

SRGM Decision Algorithm Based on TOPSIS

Based on the previous analysis and summary, we propose the *SESABCRC* (SRGM Evaluation and Selection Algorithm Based on Cost-Reliability Criterion), as shown in algorithm 1.

Algorithm 1: SRGM evaluation and selection algorithm SESABCRC considering cost-reliability

```
Input: TE fitting and predicting result value vector:
TEFP_{i,j} = \left[ tfv_{i,1}, tfv_{i,2}...tfv_{i,L-1}, tpv_{i,L} \right], 1 \le i \le K, 1 \le j \le O
SRGM fitting and predicting result value vector:
SRGMFP_{i,j} = [sfv_{i,1}, sfv_{i,2}...sfv_{i,L-1}, spv_{i,L}], 1 \le i \le K, 1 \le j \le O
weight vector W, FDS set DSSet
Output: SRGM partially ordered set SRGMOP
1: Initialize:
   For each DS in (DSSet) {
      Based on formulas (3) \sim (7), the weighted standardized
evaluation decision matrix for TE and SRGM is constructed,
respectively:TE-WSDM和SRGM-WSDM
Construct SRGM evaluation system structure tree//Get structure
tree under standardized data }
2: EvaluateSRGMWithTE:
   For each DS in (DSSet) {
      For each subtree {
          If(Node is of TE type) {
         TEValue[TE node ID]=CaculateEvaValue(TE node ID)
             Get the SRGM parent node corresponding to the TE node
             SRGMValue[SRGM node ID]=CaculateEvaValue(SRGM node ID)
             SRGMFinalOrder[SRGM节点ID]=GetFinalValue(TEValue[TE节点
ID], SRGMValue[SRGM node ID]) }
   EvaluateSRGMNoTE:
           Else {
             SRGMValue[SRGM node ID]=CaculateEvaValue(SRGM node ID)
             SRGMFinalOrder[SRGM node ID]=GetFinalValue(SRGMValue]
             SRGM node ID])
           } } }
   Return SRGMFinalOrder
```

On the whole, the *SESABCRC* algorithm finds the SRGM parent node from the TE at the bottom to the upper layer, calculates the evaluation values of TE and SRGM, and finally outputs the partial order relationship of each model. The algorithm consists of 3 parts:

- Initialization part: For each failure data set, (1) Based on the fitting results of TE and SRGM on FDS, obtain the fitting and prediction result data of both, and use formulas (3) ~ (7) to preprocess the data and obtain the weighted standardized evaluation decision matrix of TE and SRGM: TE-WSDM and SRGM-WSDM; ,Using TE-WSDM and SRGM-WSDM to construct an SRGM evaluation system structure tree.
- 2. EvaluateSRGMWithTE Evaluate the SRGM containing TE: On each failure data set, for the constructed SRGM evaluation system structure tree, start the following operations from each subtree: If it is a TE node, first use *CaculateEvaValue()* Function to calculate the TE to obtain the evaluation result; ,Based on this, the TE node finds its corresponding SRGM parent node, and calculates the evaluation result based on the parent node's vector in SRGM-WSDM; fUsing the *GetFinalValue()* function, for this node performs corresponding calculations, and then obtain the final SRGM ranking value.

3. **EvaluateSRGMNoTE - Evaluate SRGM without TE:** For SRGM without TE, directly use the *CaculateEvaValue()* and *GetFinalValue()* functions to obtain the final SRGM ranking value.

The **SESABCRC** algorithm has a certain framework; that is, the specific content of the two key evaluation functions *CaculateEvaValue()* and *GetFinalValue()* can be determined according to the actual situation.

About the Functions CaculateEvaValue () and GetFinalValue ()

The function *CaculateEvaValue()* is used in the algorithm to evaluate the evaluation results of TE and SRGM, and its core is mainly to calculate the distance between the model and a specific solution. According to needs, a simple weighting method, Euclidean distance, etc. can be used. Obviously, the literature only selects the model with the minimum distance to the positive ideal solution *POP* as optimal, but it may not ensure the maximum distance to the negative ideal solution *NOP*. In contrast, the ordinal preference method/prosperity solution distance method TOPSIS (Saaty, 2008; Shih, Shyur & Lee, 2007) calculates the distance D+ and D- from the model to the *POP* and *NOP* and then calculates the degree of closeness H from the model's solution to the ideal solution to sort each model. This method has more comprehensive trade-offs. The following shows the *CaculateEvaValue()* function based on TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution).

Algorithm 2: CaculateEvaValue () function based on the ordinal preference method TOPSIS

```
CaculateEvaValue (ID)
```

{ The positive ideal solution POP and the negative ideal solution NOP are constructed based on the evaluation value of the model, as shown in formula (8). The former is composed of the best elements of the L evaluation indicators in all models, and the latter is composed of the worst elements:

$$\begin{cases} POP(ID) = \{\max_{D} x_{ID,j} \mid 1 \le j \le (M+N+1)\} \\ NOP(ID) = \{\min_{D} x_{ID,j} \mid 1 \le j \le (M+N+1)\} \\ NOP(ID) = \{\min_{D} x_{ID,j} \mid 1 \le j \le (M+N+1)\} \\ NOP(ID) = \{\min_{D} x_{ID,j} \mid 1 \le j \le (M+N+1)\} \end{cases}$$
(8)

Obtain the distance D+(ID) and D-(ID) between the model and POP and NOP:

$$\begin{cases} D^{+}(ID) = \sqrt{\sum_{j=1}^{(M+N+1)} \left(x_{ID,j} - x_{j}^{\max}\right)^{2}}, where \ x_{j}^{\max} = \max_{1 \le i \le M+N} x_{ID,j}, 1 \le i \le K \\ D^{-}(ID) = \sqrt{\sum_{j=1}^{(M+N+1)} \left(x_{ID,j} - x_{j}^{\min}\right)^{2}}, where \ x_{j}^{\max} = \min_{1 \le i \le M+N} x_{i,j}, 1 \le i \le K \end{cases}$$
(9)

Calculate how close the model is to the ideal solution, H, as a measure of the performance of the model:

$$H(ID) = \frac{D^{-}(ID)}{D^{+}(ID) + D^{-}(ID)}, 1 \le ID \le kH(ID) = \frac{D^{-}(ID)}{D^{+}(ID) + D^{-}(ID)}, 1 \le ID \le k$$
(10)

The *GetFinalValue()* function realizes the final sorting calculation of SRGM, including SRGM with and without TE. For the former, we must first evaluate TE and then SRGM. From the perspective of "cost-reliability", the evaluation results of the TE and SRGM shall be considered comprehensively, and the final evaluation value of SRGM shall be determined in an appropriate manner. The latter does not include TE; it is only a comprehensive evaluation of the fitting and prediction of the SRGM. For SRGM evaluation that includes TE, you can set the weight vector wTE of TE and the weight vector w_{SRGM} of SRGM. These two vectors satisfy the condition $w_{TE}+w_{SRGM}=1$. At this time, the degree of closeness *H* between the model's solution and the ideal solution can be calculated as follows:

$$H(TEID, SRGMID) = w_{TE} \left(\frac{D^{-}(TEID)}{D^{+}(TEID) + D^{-}(TEID)} \right)$$
(11)
$$+w_{SRGM} \left(\frac{D^{-}(SRGMID)}{D^{+}(SRGM) + D^{-}(SRGM)} \right)$$

EVALUATION OF NUMERICAL EXAMPLES

In this section, based on the aforementioned design and analysis, we use the *SESABCRC* evaluation algorithm in the previous section to verify the fitting and prediction result values of the selected SRGMs on a given failure dataset.

SRGM Example and Data Set Selection and Parameter Setting

To explain this calculation example more concretely, we selected 6 typical SRGMs with TEF for experimental and evaluation analysis, as shown in Table 2.

The two data sets used in this paper, DS1 (Huang, Kuo, Lyu, 2007) and DS2 (Ohba, 1984), are the failure data sets recorded during the testing of large computer projects released by IBM and Tandem, respectively. These two data sets were recorded for 19 weeks and 20 weeks, and both included the cumulative test workload and the number of failures.

SRGM Performance Evaluation Benchmark Value Considering TE

First, based on the parameter fitting of the model in Table 2 on the two failure data sets, we calculated the corresponding fitting result values, as shown in Table 3. The results listed here are raw data without preprocessing. Among them, standards such as MSE correspond to DS1 and DS2. The two rows DS1 and DS2 are divided into two subrows: the upper row of data is the m(t) standard value of SRGM, and the lower row is the corresponding TE W(t) standard value.

On the other hand, the RE prediction curves of the 6 models on the 2 datasets are shown in algorithms 1 and 2. Algorithm 1 is the RE curve of the TE model, which represents the prediction of test resource consumption, and Algorithm 2 is the prediction curve of m(t) of SRGM and future failure data. The abscissa in the algorithm is the test time in weeks, and the ordinate is the numerical value that characterizes the prediction performance. The smaller the absolute value of the ordinate, the better. In particular, in view of the limitation of page space, the prediction curves of all models are drawn in one place. In fact, the *RE* curves of each model can be drawn separately, which has a more direct display distinguishing ability.

Experimental Data Processing and Result Analysis

Model Fitting and Prediction Data Processing

First, standardize the fitting data in Table 3 according to the data processing method in Section 2, and then obtain the value inside the [0,1] interval (positive growth is preferred), as shown in Table 4. On

Description	The function m(t) corresponding to the model used to represent the cumulative number of failures and the test workload function TEF contained in it
Model1:Huang(Shih, Shyur & Lee, 2007)	$\begin{split} m\left(t\right) &= \frac{a}{1-b} \Big[1 - e^{-bW^*(t)} \Big] m\left(t\right) = \frac{a}{1-b} \Big[1 - e^{-bW^*(t)} \Big];\\ W^*\left(t\right) &= \frac{W}{1+Ae^{-\alpha t}} - \frac{W}{1+A} W^*\left(t\right) = \frac{W}{1+Ae^{-\alpha t}} - \frac{W}{1+A};\\ W\left(t\right) &= \frac{W}{1+Ae^{-\alpha t}} W\left(t\right) = \frac{W}{1+Ae^{-\alpha t}} \end{split}$
Model2:Yamada(Huang, & Kuo, 2002)	$\begin{split} m\left(t\right) &= a \Big[1 - e^{-bW^*(t)}\Big] m\left(t\right) = a \Big[1 - e^{-bW^*(t)}\Big];\\ W^*\left(t\right) &= W \Big[1 - e^{-\beta t^{\sigma}}\Big] W^*\left(t\right) = W \Big[1 - e^{-\beta t^{\sigma}}\Big];\\ W\left(t\right) &= W \Big[1 - e^{-\beta t^{\sigma}}\Big] W\left(t\right) = W \Big[1 - e^{-\beta t^{\sigma}}\Big] \end{split}$
Model3:N. Ahmad(Yamada & Hishitani, 1993)	$ \begin{split} m\left(t\right) &= a \Big[1 - e^{-bW(1 - e^{-\beta t^2})^{\theta}} \Big] m\left(t\right) = a \Big[1 - e^{-bW(1 - e^{-\beta t^2})^{\theta}} \Big];\\ W\left(t\right) &= W \Big[1 - e^{-\beta t^2} \Big]^{\theta} W\left(t\right) = W \Big[1 - e^{-\beta t^2} \Big]^{\theta} \end{split} $
Model4:LTEFID (Ahmad, Khan, Quadri & Kumar, 2009)	$\begin{split} m\left(t\right) &= \frac{a}{1-r} \Big[1 - e^{-b(1-r)W^*(t)} \Big] m\left(t\right) = \frac{a}{1-r} \Big[1 - e^{-b(1-r)W^*(t)} \Big];\\ W^*\left(t\right) &= \left(\frac{W}{1+Ae^{-\alpha t}} - \frac{W}{1+A}\right) W^*\left(t\right) = \left(\frac{W}{1+Ae^{-\alpha t}} - \frac{W}{1+A}\right);\\ W\left(t\right) &= \frac{W}{1+Ae^{-\alpha t}} W\left(t\right) = \frac{W}{1+Ae^{-\alpha t}} \end{split}$
Model5:SEWTEFID(Huang, 2005)	$\begin{split} m\left(t\right) &= \frac{a\left[1-e^{-bW(t)}\right]}{1+\left[\left(1-\gamma\right)/\gamma\right]e^{-bW(t)}} m\left(t\right) = \frac{a\left[1-e^{-bW(t)}\right]}{1+\left[\left(1-\gamma\right)/\gamma\right]e^{-bW(t)}};\\ W\left(t\right) &= W(1-e^{-\varphi t^{\sigma}})^{\theta}W\left(t\right) = W(1-e^{-\varphi t^{\sigma}})^{\theta};\\ W\left(t\right) &= W(1-e^{-\varphi t^{\sigma}})^{\theta}W\left(t\right) = W(1-e^{-\varphi t^{\sigma}})^{\theta} \end{split}$
Model6:SLTEFID (Huang, Kuo & Lyu, 2007)	$ \begin{split} m(t) &= \frac{a}{1-r} \overline{\left\{ 1 - \left[1 + b W^*(t) \right]^{1-r} e^{-b(1-r)W^*(t)} \right\}} m(t) = \frac{a}{1-r} \left\{ 1 - \left[1 + b W^*(t) \right]^{1-r} e^{-b(1-r)W^*(t)} \right\} \\ W^*(t) &= \left(\frac{W}{1+A e^{-\alpha t}} - \frac{W}{1+A} \right) W^*(t) = \left(\frac{W}{1+A e^{-\alpha t}} - \frac{W}{1+A} \right); \\ W(t) &= \frac{W}{1+A e^{-\alpha t}} W(t) = \frac{W}{1+A e^{-\alpha t}} \end{split} $

Table 2. SRGM participating in the comparison and the corresponding test workload function TEF

Standard		Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
		Logistic TEF	Yamada Weibull TEF	General Burr type X TEF	Logistic TEF	Exponential Weibull TEF	Logistic TEF
	DC	114.08513928	116.84307931	114.76144684	0.1313224284	85.71104359	118.37469102
MGE		1.62719973	0.86545734	0.86188618	1.62719973	0.85190168	1.62719973
MSE	DC	21.65633246	16.58786284	14.22923960	21.65633333	12.59373552	21.69495934
		37786.65297525	19229.07905372	17257.32522641	37786.65297525	17031.67779712	37786.65297525
	DC	7.91246649	9.09877368	9.03166155	8.37691648	7.90269823	8.50540865
MEOD		1.09298213	0.88866439	0.88460248	1.09298213	0.91959213	1.09298213
MEOP	DC	3.66275040	2.86014363	2.62509435	3.86647564	2.73194349	3.83647399
		174.59094539	127.85406724	128.41521476	174.59094539	137.37250251	174.59094539
	DC	11.55819508	11.15086683	11.04431078	11.55887634	9.58372810	12.48684981
W		1.32218001	0.95764605	0.95619587	1.32218001	0.94887627	1.32218001
Variation	DC	5.30868439	4.50243209	4.14975167	5.30861126	3.84041274	5.36667151
		202.37883268	145.51956418	136.51080443	202.37883268	135.18914403	202.37883268
	DS ₁	11.73671169	11.16512334	11.05630927	11.73759327	9.60635617	12.87251066
DIGDE		1.32582450	0.95823080	0.95694555	1.32582450	0.94906539	1.32582450
KMS-PE	DS ₂	5.46696384	4.60022178	4.23444189	5.46687031	3.90144871	5.53984461
		203.30129947	146.53315666	137.05438542	203.30129947	135.59613517	203.30129947
	DC	1.06795498	0.96682639	0.96893022	1.06808874	1.01722667	1.08841999
D	DS ₁	0.96803024	1.00379260	1.00487804	0.96803024	1.00011923	0.96803024
K-square	DC	1.19649881	1.14079737	1.12647170	1.19644642	1.10195845	1.20279864
		0.97265132	1.01868839	1.01285861	0.97265132	1.01083721	0.97265132
	DC	4.76242886	4.81964954	4.77652406	4.76243057	4.12793029	4.85113535
70		4.54289345	3.31310311	3.30626057	4.54289345	3.28705414	4.54289345
15	DC	6.19865308	5.42500439	5.02453322	6.19865320	4.72696253	6.20417867
		3.02451452	2.15757402	2.04396382	3.02451452	2.03055697	3.02451452
		0.08580944	0.06669899	0.06639896	0.08583908	0.06448277	0.09504980
		0.10669329	0.07002998	0.07091754	0.10669329	0.06872885	0.10669329
BMMRE	DC	0.19362979	0.17450104	0.14636240	0.19361618	0.12668076	0.19620159
	DS ₂	0.07127320	0.08318321	0.06614712	0.07127320	0.06052061	0.07127320

Table 3. SRGM participating in the comparison and the corresponding test workload function TEF

the 7 fitting standards, the fitting values of m(t) and TEF (i.e., W(t)) of the 6 models were mapped to values inside the [0,1] interval. The pros and cons of the six models in a specific fitting standard are relatively clear, but from the perspective of all seven fitting standards, it is not yet possible to directly give the overall performance of these models.

For ease of processing, in the processing of the data in Table 3, the values are truncated to 3 decimal places.

Figure 3 shows the prediction curves of the six models on two data sets. The prediction performance is based on the speed and closeness of the curve approaching the zero standard. Obviously, the prediction performance has certain qualitative characteristics, and it is necessary to manually compare the performance level and give the judgment result. Based on our previous research results and performance comparison experience in the field of SRGM, through the comprehensive comparison

Figure 2. TE's RE prediction curve on the data set



Figure 3. RE prediction curve of SRGM on the data set



of all models in Figure 2 and Figure 3 and separating them for a single comparison, we determined the prediction performance value of each model. As shown in Table 5:

Standardize the data in Table 5, and finally obtain the prediction performance value (positive growth is preferred) within [0,1].

Thus far, based on the fitting and prediction values obtained by experimenting with 6 models on DS1 and DS2, we have obtained basic data for model performance comparison through data preprocessing. Finally, according to the standardized calculation of formula (6), standardized data and SDM can be obtained.

Experimental Results and Analysis

Based on many existing studies, it can be seen that *MSE*, *R*-square and variation are frequently used to evaluate the fitting performance of SRGM, so they are used as the key attribute of the fitting index, namely, M=3 (N=4). Its weight is more important than other fitting weights. In addition, since prediction and fitting are two aspects that characterize the performance of SRGM, their importance is basically the same, so the weights of the two are set to be the same here. In this way, the fitting and prediction weights shown in Table 6 can be obtained.

Here, we may also set the *MSE*, *R*-square and variation to account for 60% of the weight of the 7 fitting standards, and the three are equal; the remaining 4 fitting standards account for 40% of the weight, and the four are equal. In the weight setting of TE and SRGM, the proportional relationship between cost and reliability is adjusted by the linear relationship of $W_{TE}+W_{SRGM}=1$. Furthermore, these weight values will be adjusted in subsequent experiments to observe their impact on the performance of SRGM.

In the follow-up experiments, we first construct the SRGM evaluation system structure tree and calculate the result according to the *SESABCRC* algorithm flow. Under the current weight parameter setting, the comprehensive performance value and corresponding ranking shown in Table 7 can be obtained.

		Model (SRGM and TEF)								
Standard		Model 1	Model 2	Model 3	Model 4	Model 5	Model 6			
		Logistic TEF	Yamada Weibull TEF	General Burr type X TEF	Logistic TEF	Exponential Weibull TEF	Logistic TEF			
		0.1313249462	0.0468904066	0.1106197398	0.1313224284	1.0	0.0			
MCE	DS_1	0.0	0.98251555	0.9871217269	0.0	1.0	0.0			
MSE	DC	0.0042441413	0.5611439298	0.8202984442	0.0042440457	1.0	0.0			
	DS_2	0.0	0.8941265293	0.9891280318	0.0	1.0	0.0			
	DC	0.9918330737	0.0	0.0561102813	0.6035214584	1.0	0.4960933108			
MEOD	DS_1	0.0	0.9805071656	1.0	0.0	0.8320870104	0.0			
MEOP	DC	0.1641117372	0.8106550486	1.0	0.0	0.9139272189	0.0241679573			
	DS_2	0.0	1.0	0.9879934745	0.0	0.796339943	0.0			
	DC	0.319881432	0.4601884156	0.4968923711	0.3196467674	1.0	0.0			
T 7 • .•	DS_1	0.0	0.9765076557	0.9803923743	0.0	1.0	0.0			
Variation	DS ₂	0.0379929807	0.5662469805	0.797322095	0.0380408953	1.0	0.0			
		0.0	0.8462499178	0.9803294162	0.0	1.0	0.0			
	DS ₁	0.3477480852	0.5227515493	0.5560672024	0.3474781715	1.0	0.0			
		0.0	0.9756730235	0.9790843545	0.0	1.0	0.0			
KMS-PE	DS ₂	0.0444830032	0.5735016976	0.7967565837	0.0445400895	1.0	0.0			
		0.0	0.838461045	0.9784617575	0.0	1.0	0.0			
	DC	0.2314522994	0.624817759	0.6486113604	0.2299395193	0.8051722241	0.0			
D.	DS_1	0.0	0.8813691438	0.8474170591	0.0	0.9962705382	0.0			
R-square	DC	0.0310644588	0.3057282337	0.3763681058	0.0313227939	0.4972429302	0.0			
	DS_2	0.0	0.3166620839	0.5298270337	0.0	0.6037391933	0.0			
	DC	0.1226574521	0.0435364902	0.1031675442	0.1226550876	1.0	0.0			
TC	DS_1	0.0	0.9792577205	0.9847062997	0.0	1.0	0.0			
15	DC	0.0037405427	0.5274612556	0.7985598167	0.0037404614	1.0	0.0			
	DS_2	0.0	0.8722107901	0.9865116473	0.0	1.0	0.0			
	DC	0.3022982606	0.9274963907	0.9373118684	0.3013285884	1.0	0.0			
DMMDE		0.0	0.9657276652	0.9423489455	0.0	1.0	0.0			
DMMKE	DC	0.0369932292	0.3121445759	0.716895785	0.0371889979	1.0	0.0			
	DS ₂	0.525535905	0.0	0.7517270746	0.525535905	1.0	0.525535905			

Table 4. The processing results of the fitted standard values of the SRGMs considering TE on the 2 data sets

It can be seen from the above table that on different data sets, as the weight of TE (W_{TE}) gradually increases, the performance of different SRGMs shows different trends. On DS1, as W_{TE} continues to increase, the comprehensive performance values of each SRGM show a downward trend. On DS2, as W_{TE} continues to increase, the overall performance value of M1 shows a downward trend, while the rest of SRGM has the opposite characterization. In summary, the algorithm used in this paper still maintains the stability of the final sorting results under the different changing trends of the comprehensive performance values of each SRGM.

Standard		Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
		Logistic TEF	Yamada Weibull TEF	General Burr type X TEF	Logistic TEF	Exponential Weibull TEF	Logistic TEF
RE quantized value	DS ₁	7/9	5/9	5/9	7/9	3/9	9/9
		3/9	5/9	7/9	3/9	9/9	3/9
	DS ₂	1/9	7/9	7/9	5/9	9/9	3/9
		7/9	3/9	9/9	7/9	5/9	7/9

Table 5. Quantified value of model RE curve prediction performance

Table 6. Fitting and prediction weight settings

1)-1 <i>MSE</i>	0.2	
1)-2 <i>MEOP</i>	0.1	
1)-3 Variation	0.2	
1)-4 <i>RMS-PE</i>	0.1	0.5
1)-5 R-square	0.2	
1)-6 <i>TS</i>	0.1	
1)-7 <i>BMMRE</i>	0.1	
2)- <i>RE</i>	0.5	

Table 7. Model decision ranking results

Deternt	Weight setting		Com	Donk				
Dataset		Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Капк
	W _{TE} =0.1,W _{SRGM} =0.9	0.30612	0.33464	0.33366	0.30358	0.31625	0.27587	
	W _{TE} =0.2,W _{SRGM} =0.8	0.30259	0.33075	0.32878	0.30033	0.31321	0.27569	
	W _{TE} =0.3,W _{SRGM} =0.7	0.29905	0.32685	0.32390	0.29707	0.31017	0.27552	
	W _{TE} =0.4,W _{SRGM} =0.6	0.29551	0.32296	0.31902	0.29382	0.30713	0.27534	
DS1	W _{TE} =0.5,W _{SRGM} =0.5	0.29198	0.31907	0.31414	0.29056	0.30409	0.27517	M2>M3>M5>M1>M4>M6
	W _{TE} =0.6,W _{SRGM} =0.4	0.28844	0.31518	0.30925	0.28731	0.30105	0.27499	
	W _{TE} =0.7,W _{SRGM} =0.3	0.28490	0.31129	0.30437	0.28406	0.29801	0.27482	
	W _{TE} =0.8,W _{SRGM} =0.2	0.28137	0.30739	0.29949	0.28080	0.29497	0.27464	
	W _{TE} =0.9,W _{SRGM} =0.1	0.27783	0.30350	0.29461	0.27755	0.29193	0.27447	
	W _{TE} =0.1,W _{SRGM} =0.9	0.29025	0.30986	0.29203	0.27628	0.30096	0.27477	
	W _{TE} =0.2,W _{SRGM} =0.8	0.28879	0.32491	0.29234	0.27637	0.30443	0.27502	
	W _{TE} =0.3,W _{SRGM} =0.7	0.28732	0.33997	0.29265	0.27646	0.30790	0.27528	
	W _{TE} =0.4,W _{SRGM} =0.6	0.28586	0.35502	0.29296	0.27655	0.31138	0.27554	
DS2	W _{TE} =0.5,W _{SRGM} =0.5	0.28440	0.37007	0.29327	0.27663	0.31485	0.27580	M2>M5>M3>M1>M4>M6
	W _{TE} =0.6,W _{SRGM} =0.4	0.28293	0.38512	0.29358	0.27672	0.31833	0.27605	
	W _{TE} =0.7,W _{SRGM} =0.3	0.28147	0.40018	0.29389	0.27681	0.32180	0.27631	
	W _{TE} =0.8,W _{SRGM} =0.2	0.28001	0.41523	0.29420	0.27690	0.32527	0.27657	
	W _{TE} =0.9,W _{SRGM} =0.1	0.27855	0.43028	0.29451	0.27699	0.32875	0.27682	

Note: Due to limited space, all SRGM performance comprehensive values retain only five decimal places of the original data.

Comparison With Other Models: Sensitivity Analysis of Base Weight

As a comparison, we implemented the SRGM performance decision method based on Euclidean distance and the SRGM performance decision method based on the simple weighting method. The Euclidean distance method ranks SRGM performance by calculating the shortest distance between each model's solution and the ideal solution. The ideal solution is a virtual solution composed of the optimal value on each standard; the simple weighting rule ranks SRGM performance based on the weighted average.

According to Table 8, among the three calculation methods, the Euclidean distance method performs the worst. The ranking results of this method are confused, and the advantages and disadvantages of the models are not obvious, which is not conducive to ranking the advantages and disadvantages of the models. The simple weighting method performs slightly better than the Euclidean method. It performs better on DS1, and the ranking result is stable, but on DS2, when W_{TE} exceeds W_{SRGM} , the ranking result changes, which shows that different combinations of W_{TE} and W_{SRGM} will affect the stability of the judgment result of the simple weighting method. The ordinal-based preference

Method	Weight setting	Sort results(DS1-DS2)					
	W _{TE} =0.1,W _{SRGM} =0.9						
	W _{TE} =0.2,W _{SRGM} =0.8						
	W _{TE} =0.3,W _{SRGM} =0.7		M5>M3>M2>M1>M4>M6				
	W _{TE} =0.4,W _{SRGM} =0.6						
Simple weighting	W _{TE} =0.5,W _{SRGM} =0.5	M5>M3>M2>M1>M4>M6					
	W _{TE} =0.6,W _{SRGM} =0.4						
	W _{TE} =0.7,W _{SRGM} =0.3		M2: M5: M2: M1: M4: M6				
	W _{TE} =0.8,W _{SRGM} =0.2		M3>M5>M2>M1>M4>M6				
	W _{TE} =0.9,W _{SRGM} =0.1						
	W _{TE} =0.1,W _{SRGM} =0.9	M5>M2>M3>M1=M4>M6					
	W _{TE} =0.2,W _{SRGM} =0.8		M1>M6>M4>M2>M3>M5				
	W _{TE} =0.3,W _{SRGM} =0.7						
	W _{TE} =0.4,W _{SRGM} =0.6	M2>M1=M4>M5>M3>M6					
Euclidean distance method	W _{TE} =0.5,W _{SRGM} =0.5	M1 M45 M25 M65 M25 M5	M1>M6>M2>M4>M5>M3				
	W _{TE} =0.6,W _{SRGM} =0.4	M1=M4>M2>M0>M3>M3					
	$W_{TE} = 0.7, W_{SRGM} = 0.3$	M1=M4>M6>M2>M3>M5	M2>M1>M6>M5>M4>M3				
	W _{TE} =0.8,W _{SRGM} =0.2		M2>M1>M5>M6>M4>M3				
	$W_{TE} = 0.9, W_{SRGM} = 0.1$		M2>M5>M1>M6>M4>M3				
	$W_{TE} = 0.1, W_{SRGM} = 0.9$						
	W _{TE} =0.2,W _{SRGM} =0.8						
	$W_{TE} = 0.3, W_{SRGM} = 0.7$						
	W _{TE} =0.4,W _{SRGM} =0.6						
Ordinal-based preference method: TOPSIS	W _{TE} =0.5,W _{SRGM} =0.5	M2>M3>M5>M1>M4>M6	M2>M5>M3>M1>M4>M6				
	$W_{TE} = 0.6, W_{SRGM} = 0.4$						
	$W_{TE} = 0.7, W_{SRGM} = 0.3$]					
	$W_{TE} = 0.8, W_{SRGM} = 0.2$]					
	W _{TE} =0.9,W _{SRGM} =0.1						

Table 8. Comparison of three decision-making methods

method TOPSIS proposed in this paper performs best, and the ranking results are the most stable on different data sets, which can effectively rank the pros and cons of the model.

CONCLUSION

Aiming at the current situation of SRGM decision-making, this paper proposes a framework model for SRGM performance evaluation and selection covering cost-reliability, establishes an SRGM multiattribute decision-making system tree and gives the corresponding decision-making evaluation algorithm *SESABCRC*. In particular, in terms of evaluation criteria, this article comprehensively considers fitting criteria and prediction criteria, sets reasonable weights based on expert experience and uses TOPSIS to rank SRGMs considering cost. Experimental results and analysis show that the proposed model and algorithm can accurately evaluate and select SRGM performance and provide decision support for actual model selection. In practical applications, as long as a complete set of evaluation criteria and candidate SRGMs are defined, the *SESABCRC* method can be used to efficiently evaluate and select multiattribute decision-making models. This provides important decision support for software development activities, including testing resources and cost control and optimal release time selection.

In fact, the SRGM multiattribute decision-making problem has different results in different situations. For example, although both are no preference algorithms, the results of the maximumminimum method (depending on the worst attribute) and the minimum-maximum method (depending on the best attribute) are very different, or even opposite. Therefore, it is more pertinent to specify the test environment and operating environment to make SRGM decision-making. In addition, from the perspective of conditional attributes and decision attributes, the establishment of a rough set-based SRGM multiattribute decision-making model in subsequent research is also an important research direction.

FUNDING AGENCY

Open Access Funding for this article has been covered by the authors of this manuscript.

Fund Projects: Shandong Province Natural Science Foundation (ZR2021MF067); Weihai Science and Technology Development Program (ITEAZMZ001807); Shanxi Province Basic Research Program (201801D121120).

REFERENCES

Ahmad, N., Bokhari, M. U., Quadri, S., & Khan, M. G. M. (2008). The exponentiated Weibull software reliability growth model with various testing-efforts and optimal release policy: A performance analysis. *International Journal of Quality & Reliability Management*, 25(2), 211–235. doi:10.1108/02656710810846952

Ahmad, N., Khan, M., & Rafi, L. S. (2010). A study of testing-effort dependent inflection s-shaped software reliability growth models with imperfect debugging. *International Journal of Quality & Reliability Management*, 27(1), 89–110. doi:10.1108/02656711011009335

Ahmad, N., Khan, M. G., Quadri, S. M. K., & Kumar, M. (2009). Modeling and analysis of software reliability with Burr type X testing-effort and release-time determination. *Journal of Modelling in Management*.

Anjum, M., Haque, M. A., & Ahmad, N. (2013). Analysis and ranking of software reliability models based on weighted criteria value. *International Journal of Information Technology and Computer Science*, 2(1), 1–14.

Ce, Z., Fan-Chao, M., Yong-Gui, K., Wei-Gong, L., Hong-Wei, L., & Kun, W. (2017). Survey of software reliability growth model. *Journal of Software*, 28(9), 2402–2430.

Ce, Z., Fan-Chao, M., Yong-Gui, K., Wei-Gong, L., Hong-Wei, L., & Kun, W. (2017). Survey of Software Reliability Growth Model. Academic Press.

Dohi, T., Matsuoka, T., & Osaki, S. (2002). An infinite server queuing model for assessment of the software reliability. *Electronics and Communications in Japan (Part III Fundamental Electronic Science)*, 85(3), 43–51. doi:10.1002/ecjc.1078

Garg, R., Sharma, K., Kumar, R., & Garg, R. K. (2010). Performance analysis of software reliability models using matrix method. *World Academy of Science, Engineering and Technology*, *71*, 31–38.

Goel, A. L., & Okumoto, K. (2009). Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE Transactions on Reliability*, *R*-28(3), 206–211. doi:10.1109/TR.1979.5220566

Gupta, A., Gupta, N., & Garg, R. K. (2018). Implementing weighted entropy-distance based approach for the selection of software reliability growth models. *International Journal of Computer Applications in Technology*, 57(3), 255–266. doi:10.1504/IJCAT.2018.092990

Huang, C. Y. (2005). Cost-reliability-optimal release policy for software reliability models incorporating improvements in testing efficiency. *Journal of Systems and Software*, 77(2), 139–155. doi:10.1016/j.jss.2004.10.014

Huang, C. Y., & Kuo, S. Y. (2002). Analysis of incorporating logistic testing-effort function into software reliability modeling. *IEEE Transactions on Reliability*, *51*(3), 261–270. doi:10.1109/TR.2002.801847

Huang, C. Y., Kuo, S. Y., & Lyu, M. R. (2007). An assessment of testing-effort dependent software reliability growth models. *IEEE Transactions on Reliability*, 56(2), 198–211.

Huang, C. Y., & Lyu, M. R. (2005). Optimal release time for software systems considering cost, testing effort, and test efficiency. *IEEE Transactions on Reliability*, 54(4), 583–591. doi:10.1109/TR.2005.859230

Khalid, B., & Sharma, K. (2015, March). Ranking of software reliability growth models using Bacterial Foraging Optimization Algorithm. In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 1643-1648). IEEE.

Lakshmanan, I., & Ramasamy, S. (2015). An artificial neural-network approach to software reliability growth modeling. *Procedia Computer Science*, *57*, 695–702. doi:10.1016/j.procs.2015.07.450

Ohba, M. (1984). Software reliability analysis models. *IBM Journal of Research and Development*, 28(4), 428–443.

Okamura, H., & Dohi, T. (2014). A Novel Framework of Software Reliability Evaluation with Software Reliability Growth Models and Software Metrics. In *IEEE International Symposium on High-assurance Systems Engineering*. IEEE.

Okamura, H., Etani, Y., & Dohi, T. (2011). Quantifying the Effectiveness of Testing Efforts on Software Fault Detection with a Logit Software Reliability Growth Model. In *Joint Conference of the International Workshop on Software Measurement & the International Conference on Software Process & Product Measurement*. IEEE Computer Society.

Rana, R., Staron, M., Berger, C., Hansson, J., Nilsson, M., Törner, F., Meding, W., & Höglund, C. (2014). Selecting software reliability growth models and improving their predictive accuracy using historical projects data. *Journal of Systems and Software*, *98*, 59–78. doi:10.1016/j.jss.2014.08.033

Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, 1(1), 83–98. doi:10.1504/IJSSCI.2008.017590

Saaty, T. L., & Kearns, K. P. (1985). The Analytic Hierarchy Process. Academic Press.

Sharma, K., Garg, R., Nagpal, C. K., & Garg, R. K. (2010). Selection of optimal software reliability growth models using a distance based approach. *IEEE Transactions on Reliability*, 59(2), 266–276. doi:10.1109/TR.2010.2048657

Shih, H. S., Shyur, H. J., & Lee, E. S. (2007). An extension of TOPSIS for group decision making. *Mathematical and Computer Modelling*, 45(7-8), 801–813. doi:10.1016/j.mcm.2006.03.023

Stringfellow, C., & Andrews, A. A. (2002). An empirical method for selecting software reliability growth models. *Empirical Software Engineering*, 7(4), 319–343. doi:10.1023/A:1020515105175

Wood, A. (1996). Predicting software reliability. Computer, 29(11), 69-77.

Yamada, S. (2014). Software Reliability Modeling: Fundamentals and Applications. Springer-Verlag.

Yamada, S., Hishitani, J., & Osaki, S. (1993). Software-reliability growth with a Weibull test-effort: A model and application. *IEEE Transactions on Reliability*, 42(1), 100–106. doi:10.1109/24.210278

Zhang, C., Cui, G., Liu, H., Meng, F., & Wu, S. (2014). A Unified and Flexible Framework of Imperfect Debugging Dependent SRGMs with Testing-Effort. *Journal of Multimedia*, *9*(2). Advance online publication. doi:10.4304/jmm.9.2.310-317

Zhang, C., Cui, G., Liu, H., Meng, F., & Zhongchuan, F. U. (2014). Software test resources and cost control and optimal release policy. *Journal of Harbin Institute of Technology*, *46*(5), 51–58.

Zhang, C., Cui, G., Meng, F. C., Liu, H., & Wu, S. (2013). A study of optimal release policy for SRGM with imperfect debugging. *Journal of Engineering Science and Technology Review*, 6(3), 111–118. doi:10.25103/ jestr.063.20

Zhang, C., Meng, F. C., & Wan, K. (2016). Analysis on SRGM modeling categories and performances. *Journal of Harbin Institute of Technology*, 48(8), 171–178.