A Risk Measurement Model of Software Project Based on Grey Variable Weight Clustering analysis

Ning He*, Xin Mao

*Department of electrical engineering, Chongqing Water Resources and Electric Engineering College, 402160, China
bState Grid Chongqing electric power company Yongchuan branch, 402160, China
hn_mx@126.com

Effectively identifying and analyzing risks are critical to ensuring the quality of software project. Risk measurement analysis of software project is a process of multi-factor, fuzziness, hierarchy and complexity, in response to which we carry out a research on the multi-level risk measurement of software project, and propose a multi-level risk measurement model of software project based on grey variable weight clustering analysis. The model first analyzes factors influencing risk measurement analysis of software project, and on this basis, an evaluation index system of multi-level risk measurement analysis is established. Then, considering that evaluation indexes may exert different influences on risk measurement, we propose a risk measurement model of software project based on grey variable weight clustering analysis. Finally, a gray close-degree model is established between software project and various risk categories, and the rank for risk measurement of software project is confirmed based on close degrees. Additionally, a practical case analysis is presented to illustrate and verify the proposed model and algorithms.

1. Introduction

In the process of software development, there are many unknown and uncertain factors posing multiple kinds of risks to software development or products, which may lead to slow progress of software design, low execution efficiency of software system, and added cost of software development. As a result, software project suffers serious losses (Wu, 2015; Yin, 2013; Gu, 2016). Consequently, the risk measurement analysis and control of software project have gradually evolved into a management science, emerging as a research highlight in computer engineering (Luo, 2011; Mu, 2007; Bi, 2014). To this end, many researchers have analyzed, studied and explored risk measurement of software project from different perspectives and starting points. A series of research results have been made in such respects as evaluation indexes of risk measurement of software project (LI, 2013; GE, 2009), risk control methods (Zhao, 2007; Wang, 2015) and risk evaluation models (Yang, 2010; Pang, 2009). Nonetheless, the available models and methods have some limitations, because risk measurement analysis of software project is a process of multi-factor, fuzziness, hierarchy and complexity, and fuzzy factors that may have some gray information often exert diverse effects. Hence, from analysis gray system theory (Liu, 2012; Tsai-Fu, 2014), this paper adopts an improved method of gray clustering analysis (Ker-Tah, 2011; Liu, 2012; Alexi, 2016) and establishes an evaluation system for risk measurement of software subject, so as to present a multi-level risk measurement model of software project based on grey variable weight clustering.

2. An index system for multi-level risk measurement of software project

2.1 Risk analysis of software project

Software project is under the combined influence of various factors, so there is a need for a targeted analysis of such factors in software development to work out effective solutions and methods. After making generalizations and analyses of related research results, the influencing factors of software project risks in this paper are considered to generally include:
(1) Technical factor: mainly concern the influences of development capabilities and technical standards and specifications on software project development.
(2) Management factor: mainly deal with the influences of management and organization ability, coordination and communication ability as well as schedule planning ability of managers or regulatory authority.
(3) Economic factor: mainly relate to the influences of cost control as well as reasonability and effectiveness of resource utilization.
(4) Human factor: mainly concern the influences of team building and development ability, as well as mobility and sustainability of developers.
(5) Quality factor: mainly concern the influences of team building and development ability, as well as mobility and sustainability of developers.
(6) Demand factor: mainly concern the influences of cognitive, converted and participatory requirements in software development.
(7) Market factor: mainly involve the influences of changes in market environment.
(8) Environmental factor: mainly relate to the influences of hardware platform construction and design specifications.

2.2 Establishment of an index system for multi-level risk measurement
Based on the comprehensive analysis of various influencing factors, we establish an improved index system for multi-level risk measurement of software project by generalizing and extracting key features of influencing factors, and Table 1 shows the specific architecture and index content.

### Table 1: Index system for multi-level risk measurement of software project

<table>
<thead>
<tr>
<th>Factor level</th>
<th>Index level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical factor $S_1$</td>
<td>Technology maturity $S_{11}$</td>
</tr>
<tr>
<td></td>
<td>Technical specification $S_{12}$</td>
</tr>
<tr>
<td>Management factor $S_2$</td>
<td>Progress controllability $S_{21}$</td>
</tr>
<tr>
<td></td>
<td>Organization and management ability $S_{22}$</td>
</tr>
<tr>
<td>Economic factor $S_3$</td>
<td>Earnings ratio $S_{31}$</td>
</tr>
<tr>
<td>Human factor $S_4$</td>
<td>Personnel stability $S_{41}$</td>
</tr>
<tr>
<td>Quality factor $S_5$</td>
<td>Availability $S_{51}$</td>
</tr>
<tr>
<td></td>
<td>Reliability $S_{52}$</td>
</tr>
<tr>
<td></td>
<td>Safety $S_{53}$</td>
</tr>
<tr>
<td>Demand factor $S_6$</td>
<td>Demand adaptability $S_{61}$</td>
</tr>
<tr>
<td>Market factor $S_7$</td>
<td>Market adaptability $S_{71}$</td>
</tr>
<tr>
<td>Environmental factor $S_8$</td>
<td>Platform construction ability $S_{81}$</td>
</tr>
</tbody>
</table>

3. Grey variable weight clustering analysis of risk measurement of software project

3.1 Weight function
Weight function is a central part of grey clustering analysis. Choosing suitable weight functions according to various forms of grey clustering analysis can effectively determine the grey category of target object. Suppose the target object is divided into $k$ grey categories based on $j$ indexes, and $a_1^j(1)$, $a_2^j(2)$, $a_3^j(3)$ and $a_4^j(4)$ are the turning points of weight function $f_j(a)$, then Graph 1 shows a typical weight function structure.

![Figure 1: Typical weight function](image)

Its model can be expressed as:
In particular, if $a_k^j(2)=a_k^j(3)$, $f_k^j(a)$ is a moderate weight function, which can be represented as:

$$f_k^j(a) = \begin{cases} \frac{a-a_k^j(1)}{a_k^j(2)-a_k^j(1)} & a \in \left[a_k^j(1), a_k^j(2)\right] \\ 0 & a \notin \left[0, a_k^j(4)\right] \\ \frac{a_k^j(4)-a}{a_k^j(4)-a_k^j(3)} & a \in \left[a_k^j(3), a_k^j(4)\right] \\ 1 & a \in \left[0, a_k^j(3)\right] \end{cases}$$

(2)

If $a_k^j(1)=a_k^j(2)=0$, $f_k^j(a)$ is a lower-measurement weight function, which can be expressed as:

$$f_k^j(a) = \begin{cases} 0 & a \notin \left[0, a_k^j(4)\right] \\ \frac{a_k^j(4)-a}{a_k^j(4)-a_k^j(3)} & a \in \left[a_k^j(3), a_k^j(4)\right] \\ 1 & a \in \left[0, a_k^j(3)\right] \end{cases}$$

(3)

If $a_k^j(3)=a_k^j(4)=1$, $f_k^j(a)$ is an upper-measurement weight function, which can be formulated as:

$$f_k^j(a) = \begin{cases} 1 & a \in \left[a_k^j(2), 1\right] \\ \frac{a-a_k^j(1)}{a_k^j(2)-a_k^j(1)} & a \in \left[a_k^j(1), a_k^j(2)\right] \\ 0 & a \in \left[0, a_k^j(1)\right] \end{cases}$$

(4)

3.2 Classification of gray clustering category

As shown in the hierarchical structure of Table 1, system level is decided by factor level, while factor level relies on index level. For this reason, reasonable classification of gray categories at the index level is critical to determining the risk measurement grade of software project. If the gray categories are classified too narrowly, it is not easy to distinguish; if the classification is too wide, it affects the accuracy of measurement analysis. This paper divides the gray categories of risk measurement index of software project into low ($L_l$), secondary ($L_s$), high ($L_h$) and higher ($L_{bh}$), and the gray category set $L$ is given as follows:

$$L = \{L_l, L_s, L_h, L_{bh}\}$$

(5)

3.3 Model of gray variable weight clustering analysis

Indexes corresponding to different gray categories often have different weights, and weight $w_j$ of index $j$ corresponding to gray category $k$ can be expressed as:

$$w_j = f_k(a)$$

(6)

where $f_k(a)$ is a typical weight function, weight $w_j$ can be calculated as:

$$w_j = \frac{a_k^j(2)+a_k^j(3)}{2}$$

(7)

If $f_k(a)$ is a lower-measurement weight function, weight $w_j$ can be expressed as:
Under other circumstances, it can be expressed as
\[ w_j = d_j. \]  
(8)

After analysis, turning points of weight functions for various gray categories are chosen to form the corresponding models, namely the gray-category weight function \( f_j^1(a) \) with higher risks is a lower-measurement one, which can be expressed as:
\[
f_j^1(a) = \begin{cases} 
0 & a \notin [0,0.7] \\
0.7-a & a \in [0.6,0.7] \\
1 & a \in [0,0.6]
\end{cases}.
\]  
(10)

The gray-category weight function \( f_j^2(a) \) with high risks is a moderate one, which can be formulated as:
\[
f_j^2(a) = \begin{cases} 
a-0.6 & a \in [0.6,0.7] \\
0 & a \notin [0.6,0.8] \\
0.8-a & a \in [0.7,0.8] \\
n/1 & a \in [0,0.1]
\end{cases}.
\]  
(11)

The gray-category weight function \( f_j^3(a) \) with secondary risks is a moderate one, which can be expressed as:
\[
f_j^3(a) = \begin{cases} 
a-0.7 & a \in [0.7,0.8] \\
0 & a \notin [0.7,0.9] \\
0.9-a & a \in [0.8,0.9] \\
n/1 & a \in [0,0.1]
\end{cases}.
\]  
(12)

The gray-category weight function \( f_j^4(a) \) with low risks is an upper-measurement one, which can be represented as:
\[
f_j^4(a) = \begin{cases} 
1 & a \in [0.9,1] \\
a-0.8 & a \in [0.8,0.9] \\
0 & a \in [0,0.9] \\
n/1 & a \in [0,0.1]
\end{cases}.
\]  
(13)

### 3.4 Model and algorithms implementation

On the basis of weight and weight function, the relative gray close degree \( \zeta^k \) can be calculated as:
\[
\zeta^k = \sum_{j=1}^{n} \left( w_j^k \ast f_j^k(a) \right)
\]  
(14)

Upon normalization, \( \zeta^k \) can be expressed as:
\[
\zeta^k = \zeta^k / \sum_{k=1}^{m} \zeta^k
\]  
(15)

In this way, the gray category of software project risks can be determined based on the magnitude of gray close degree \( \zeta^k \). The following describes the detailed implementation steps.
(1) Analyze the influencing factors of software project risks through investigation;
(2) Solicit experts for advice to form an index set of risk measurement analysis of software project;
(3) Establish a gray category set of risk measurement of software project;
(4) Establish a weight function of gray clustering analysis based on diverse gray categories;
(5) Obtain the weight of corresponding gray categories of various measurement indexes;
(6) Calculate the gray close degree between analysis object and diverse gray categories, and on this basis, determine the risk category of software project.

4. Case analysis

This paper employs a newly-developed software system of a software company to further illustrate and analyze the proposed model. Evaluation value (with a ratio scale of 0-1) of various measurement indexes is obtained through expert grading, as shown in Table 2.

<table>
<thead>
<tr>
<th>Risk measurement indexes</th>
<th>Evaluation value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology maturity $S_{11}$</td>
<td>0.85</td>
</tr>
<tr>
<td>Technical specifications $S_{12}$</td>
<td>0.95</td>
</tr>
<tr>
<td>Progress controllability $S_{21}$</td>
<td>0.95</td>
</tr>
<tr>
<td>Organization and management ability $S_{22}$</td>
<td>0.90</td>
</tr>
<tr>
<td>Earnings ratio $S_{31}$</td>
<td>0.70</td>
</tr>
<tr>
<td>Personnel stability $S_{41}$</td>
<td>0.60</td>
</tr>
<tr>
<td>Availability $S_{51}$</td>
<td>0.85</td>
</tr>
<tr>
<td>Reliability $S_{61}$</td>
<td>0.80</td>
</tr>
<tr>
<td>Safety $S_{51}$</td>
<td>0.85</td>
</tr>
<tr>
<td>Demand adaptability $S_{61}$</td>
<td>0.50</td>
</tr>
<tr>
<td>Market adaptability $S_{71}$</td>
<td>0.60</td>
</tr>
<tr>
<td>Platform construction ability $S_{81}$</td>
<td>0.85</td>
</tr>
</tbody>
</table>

Based on the presented weight functions of various gray categories, gray category weight of different measurement indexes can be worked out, namely $w=(0.200,0.233,0.267,0.300)$, and Table 3 shows the gray clustering coefficient.

<table>
<thead>
<tr>
<th>Risk measurement indexes</th>
<th>Risk categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology maturity $S_{11}$</td>
<td>Higher</td>
</tr>
<tr>
<td>Technical specifications $S_{12}$</td>
<td>0</td>
</tr>
<tr>
<td>Progress controllability $S_{21}$</td>
<td>0</td>
</tr>
<tr>
<td>Organization and management ability $S_{22}$</td>
<td>0</td>
</tr>
<tr>
<td>Earnings ratio $S_{31}$</td>
<td>0</td>
</tr>
<tr>
<td>Personnel stability $S_{41}$</td>
<td>1.00</td>
</tr>
<tr>
<td>Availability $S_{51}$</td>
<td>0</td>
</tr>
<tr>
<td>Reliability $S_{61}$</td>
<td>0</td>
</tr>
<tr>
<td>Safety $S_{51}$</td>
<td>0</td>
</tr>
<tr>
<td>Demand adaptability $S_{61}$</td>
<td>1.00</td>
</tr>
<tr>
<td>Market adaptability $S_{71}$</td>
<td>1.00</td>
</tr>
<tr>
<td>Platform construction ability $S_{81}$</td>
<td>0</td>
</tr>
</tbody>
</table>

Hence, the gray close-degree sequence can be derived, namely $\xi=(0.191,0.074,0.256,0.279)$. It can be seen that the software project has low risks, which is consistent with the result of later software project implementation and demonstrates the validity of the proposed model.
5. Conclusions

This paper analyzes the influencing factors of software project risks and establishes an index system for risk measurement. The corresponding gray categories set is established concerning diverse measurement analysis indexes. Moreover, weight function is introduced, and on this basis, this paper presents a risk measurement model of software project based on grey variable weight clustering analysis. Then, gray category of software project risks is determined according to the gray close degree. Finally, we demonstrate the validity of the proposed model with a practical case.

References