CHEMICAL ENGINEERING TRANSACTIONS
VOL. 51, 2016

A publication of AIDIC
The Italian Association of Chemical Engineering
Online at www.aidic.it/cet

Guest Editors: Tichun Wang, Hongyang Zhang, Lei Tian

ISBN 978-88-95608-43-3; **ISSN** 2283-9216
DOI: 10.3303/CET1651092

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

Ning He*[a], Xin Mao[b]

[a]Department of electrical engineering, Chongqing Water Resources and Electric Engineering College, 402160, China
[b]State 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:

Please cite this article as: He N., Mao X., 2016, A risk measurement model of software project based on grey variable weight clustering analysis, Chemical Engineering Transactions, 51, 547-552 DOI:10.3303/CET1651092

(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 relate to the influences of performances of system, modules, interfaces and components.
(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

System level	Factor level	Index level
Index system for multi-level risk measurement of software project *S*	Technical factor S_1	Technology maturity S_{11}
		Technical specification S_{12}
	Management factor S_2	Progress controllability S_{21}
		Organization and management ability S_{22}
	Economic factor S_3	Earnings ratio S_{31}
	Human factor S_4	Personnel stability S_{41}
	Quality factor S_5	Availability S_{51}
		Reliability S_{52}
		Safety S_{53}
	Demand factor S_6	Demand adaptability S_{61}
	Market factor S_7	Market adaptability S_{71}
	Environmental factor S_8	Platform construction ability S_{81}

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

3.1 Weight function

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

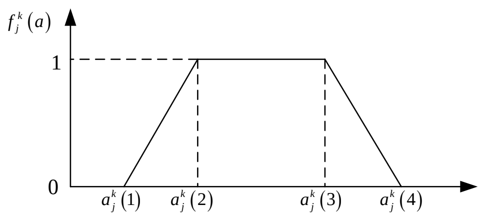

Figure 1: Typical weight function

Its model can be expressed as:

$$f_j^k(a)=\begin{cases}0 & a\notin\left[a_j^k(1),a_j^k(4)\right]\\ \dfrac{a\text{-}a_j^k(1)}{a_j^k(2)\text{-}a_j^k(1)} & a\in\left[a_j^k(1),a_j^k(2)\right]\\ 1 & a\in\left[a_j^k(2),a_j^k(3)\right]\\ \dfrac{a_j^k(4)\text{-}a}{a_j^k(4)\text{-}a_j^k(3)} & a\in\left[a_j^k(3),a_j^k(4)\right]\end{cases} \tag{1}$$

In particular, if $a_j^k(2)=a_j^k(3)$, $f_j^k(a)$ is a moderate weight function, which can be represented as:

$$f_j^k(a)=\begin{cases}\dfrac{a\text{-}a_j^k(1)}{a_j^k(2)\text{-}a_j^k(1)} & a\in\left[a_j^k(1),a_j^k(2)\right]\\ 0 & a\notin\left[a_j^k(1),a_j^k(4)\right]\\ \dfrac{a_j^k(4)\text{-}a}{a_j^k(4)\text{-}a_j^k(2)} & a\in\left[a_j^k(2),a_j^k(4)\right]\end{cases} \tag{2}$$

If $a_j^k(1)=a_j^k(2)=0$, $f_j^k(a)$ is a lower-measurement weight function, which can be expressed as:

$$f_j^k(a)=\begin{cases}0 & a\notin\left[0,a_j^k(4)\right]\\ \dfrac{a_j^k(4)\text{-}a}{a_j^k(4)\text{-}a_j^k(3)} & a\in\left[a_j^k(3),a_j^k(4)\right]\\ 1 & a\in\left[0,a_j^k(3)\right]\end{cases} \tag{3}$$

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

$$f_j^k(a)=\begin{cases}1 & a\in\left[a_j^k(2),1\right]\\ \dfrac{a\text{-}a_j^k(1)}{a_j^k(2)\text{-}a_j^k(1)} & a\in\left[a_j^k(1),a_j^k(2)\right]\\ 0 & a\in\left[0,a_j^k(1)\right]\end{cases} \tag{4}$$

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_k) and higher (L_{bh}), and the gray category set L is given as follows:

$$L=\{L_l,L_s,L_h,L_{bh}\} \tag{5}$$

3.3 Model of gray variable weight clustering analysis

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

$$w_j^k=\varpi_j^k/\sum_{k=1}^{m}\varpi_j^k \tag{6}$$

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

$$\varpi_j^k=\frac{a_j^k(2)+a_j^k(3)}{2} \tag{7}$$

If $f_j^k(a)$ is a lower-measurement weight function, weight w_j^k can be expressed as:

$$\varpi_j^k = a_j^k(3) \tag{8}$$

Under other circumstances, it can be expressed as $\varpi_j^k = a_j^k(2)$. (9)

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:

$$f_j^1(a) = \begin{cases} 0 & a \notin [0, 0.7] \\ \dfrac{0.7\text{-}a}{0.1} & a \in [0.6, 0.7] \\ 1 & a \in [0, 0.6] \end{cases} \tag{10}$$

The gray-category weight function $f_j^2(a)$ with high risks is a moderate one, which can be formulated as:

$$f_j^2(a) = \begin{cases} \dfrac{a\text{-}0.6}{0.1} & a \in [0.6, 0.7] \\ 0 & a \notin [0.6, 0.8] \\ \dfrac{0.8\text{-}a}{0.1} & a \in [0.7, 0.8] \end{cases} \tag{11}$$

The gray-category weight function $f_j^3(a)$ with secondary risks is a moderate one, which can be expressed as:

$$f_j^3(a) = \begin{cases} \dfrac{a\text{-}0.7}{0.1} & a \in [0.7, 0.8] \\ 0 & a \notin [0.7, 0.9] \\ \dfrac{0.9\text{-}a}{0.1} & a \in [0.8, 0.9] \end{cases} \tag{12}$$

The gray-category weigh function $f_j^4(a)$ with low risks is an upper-measurement one, which can be represented as:

$$f_j^4(a) = \begin{cases} 1 & a \in [0.9, 1] \\ \dfrac{a\text{-}0.8}{0.1} & a \in [0.8, 0.9] \\ 0 & a \in [0, 0.9] \end{cases} \tag{13}$$

3.4 Model and algorithms implementation

On the basis of weight and weight function, the relative gray close degree ζ^k can be calculated as:

$$\zeta^k = \sum_{j=1}^{n} \left(w_j^k * f_j^k(a) \right) \tag{14}$$

Upon normalization, ζ^k can be expressed as:

$$\xi^k = \zeta^k / \sum_{k=1}^{m} \zeta^k \tag{15}$$

In this way, the gray category of software project risks can be determined based on the magnitude of gray close degree ζ^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-develop 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 2: Value of risk measurement indexes

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

Based on the presented weight functions of various gray categories, gray category weight of different measurement indexes can be worked out, namely *w*=(0.200,233,0.267.0.300), and Table 3 shows the gray clustering coefficient.

Table 3: Gray clustering coefficient of risk measurement

Risk measurement indexes	Risk categories			
	Higher	High	Secondary	Low
Technology maturity S_{11}	0	0	0.50	0.50
Technical specifications S_{12}	0	0	0	1.00
Progress controllability S_{21}	0	0	0	1.00
Organization and management ability S_{21}	0	0	0	1.00
Earnings ratio S_{31}	0	1.00	0	0
Personnel stability S_{41}	1.00	0	0	0
Availability S_{51}	0	0	0.50	0.50
Reliability S_{51}	0	0	1.00	0
Safety S_{51}	0	0	0.50	0.50
Demand adaptability S_{61}	1.00	0	0	0
Market adaptability S_{71}	1.00	0	0	0
Platform construction ability S_{81}	0	0	0.50	0.50

Hence, the gray close-degree sequence can be derived, namely $\xi=\left(0.191,0.074,0.256,0.279\right)$. 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

Bi M., 2014, Present situation analysis and Countermeasure of software project management [J]. Computer CD Software and Applications, (21): 71-72.

Chuang T.F., Chang Y.H., 2014, Comparison of physical characteristics between Rana latouchtii and Rana adenopleura using grey system theory and Artificial Neural Network [J]. Ecological Engineering, 68(7): 223-232.

Delgado A., Romero I., 2016, Environmental conflict analysis using an integrated grey clustering and entropy-weight method: A case study of a mining project in Peru [J]. Environmental Modelling & Software, 77(3): 108-121.

Ge F.Z., Peng T.L., Guo G.F., 2009, Application of grey clustering method in software risk assessment [J]. Computer Engineering and Applications. 45(20): 78- 80.

Hsu K.T., 2011, Using a back propagation network combined with grey clustering to forecast policyholder decision to purchase investment-inked insurance [J]. Expert Systems with Applications, 38(6): 6736-6747.

Liu K., Shen X.L., Tan Z.F., Guo W.Y., 2012, Grey Clustering Analysis Method for Overseas Energy Project Investment Risk Decision [J]. Systems Engineering Procedia, 3: 55-62.

Liu S.F., Fang Z.G., Yang Y.J., et al., 2012, General grey numbers and its operations [J]. Grey Systems: Theory and Application, 2(3): 4-15.

Li W.T., Zhang J., Wang Y.X., 2013, Research on Software Risk Assessment Based on Grey System Theory [J]. COMPUTER TECHNOLOGY AND DEVELOPMENT, 23(5): 124-126.

Luo X., NI J.J., 2011, Software Project Risk Management Overview [J]. Modern Computer, (9): 16-19.

Mu L.L., Chen L.W., Zhang J.J., 2007, Survey on software project risk management [J]. Computer Engineering and Applications, 43(10): 202-205.

Pang Q.H., 2009, Grey Evaluation Method of Software project Risk and its Application [J]. Science technology and Engineering, 9(23): 7162-7165.

Wang L., Xie K., Wu G.J., et al., 2015, Risk Assessment for Software Project using Fuzzy Logic [J]. Aeronautical Computing Technique, 45(1): 108-110.

Wang L.T., 2016, Research on software project risk management model [J]. INFORMATION & COMMUNICATIONS, (6): 127.

Wu D.S., Li J.P, Sun X.L., et al., 2015, Multi-object risk control model for software project considering risk correlation [J]. Systems Engineering-Theory & Practice, 35(3): 578-586.

Yang L., Li N., He Y.Y., 2010, Application of triangular fuzzy number multi -attribute decision making method in software project risk evaluation [J]. Computer Engineering and Applications, 46(11): 246-248.

Yin Q.L., 2013, Risk analysis and control of software development [J]. Electronic Technology & Software Engineering, 2013, (10): 28-29.

Zhao H.X., Yang J.P., 2007, Software project risk assessment based on fuzzy recognition [J]. Techno economics & Management Research, (4): 52-53.