Component-based Software Safety Assessment Method

Tao GU¹, Qing ZHOU²*, Dong-yi LING³ and Shi-hai WANG³

¹China National Aeronautical Radio Electronics Research Institute, Shanghai, China
²Science and Technology on Avionics Integration Laboratory, Shanghai, China
³School of Reliability and Systems Engineering, Beihang University, Beijing, China

*Corresponding author

Keywords: Component-based software, Software architecture, Safety, Fuzzy set.

Abstract. Along with the thought of component based software development and with the software has been widely employed in safety-critical systems; an effective early evaluation of the software on the safety property of a safety-critical system is needed for the system which is developed by the component based software development. The safety of software mainly depends on the early stage of the system design, rather than after the implementation of the software. Software architecture is always designed at the early stage of development of a system, which involves the basic and significant information for safety analysis. In order to solve the problem faced by safety prediction and analysis for large-scale complex software system, in this paper we propose for safety degree assessment based on the component-based software architecture by using fuzzy mathematics. Based on the system architecture safety analysis and the architecture analysis, the safety prediction of one single component can be gain. With a consideration for the architecture dynamic behavior, a connection between the safety prediction for a software system and the dynamic behavior of the ones is made in our approach, which can guidance to software safety design effectively.

Introduction

Complex safety-critical systems (SCS) exhibit much more initial failure rate, failure consequence than could be predicted. And these factors determines the safety degree of the system. The assessment result of safety degree are a consequence of the design defects, process flows.

The traditional measurement of software safety work is mainly reflected in two aspects, the method based on the risk and based on the defect. Fenton 1) think that measuring the software safety should consider the factors effecting the risk such as control events and trigger events, and should consider the effect of consequences of mitigation measures for safety. Fenton He developed the AgenaRisk safety assessment system. But this model needs to input some many software failure probabilities and control events success probabilities which will bring subjective judge inescapable. In The 2), the authors analyses the defects in the process of risk. It developed a SCS framework to measure the software safety degree based on the McCall³)model. But it is worth to point out the software safety is not certain inversely proportional with the number of defects existing in the system, which rather than relates to the software operation profiles.

There is no quantitative assessment method to the software safety for the safety assessment often consider as an subjective work. The safety degree may be gain when the system has been developed. And the motivation of this paper is to have a feasible safety evaluation method for system at design period of the component-based software engineering. To deal with the subjective judge issue, the method employs the fuzzy sets and the D-S theory to develop evidence combine algorithms for combining the subjective decision and the exact method of dealing with mathematics to assessment software safety degree.
A Software Safety Degree Assessment Method Based on Fuzzy Sets

Generally, a safety analyze analysis method for software always starts with the requirement specifications analysis. Because the safety analysis is subjective understanding, a quantitative safety analysis is needed to evaluate uncertain potential hazards and assess assesses by the safety property associated with the safety requirement specifications. Considering the static requirement specification of each component, this kind of analysis cannot reflect the whole software safety in real operating conditions of the component-based software. Then we propose the software running path information in safety assessment to simulate dynamic conditions of the software.

![Diagram](image)

**Figure 1. A diagram for synthesizing the safety expression to calculate safety degree.**

**Component Safety Fuzzy Sets Comprehensive Method**

At this stage, a safety assessment method based on the component-based software architecture for one component is proposed. It uses fuzzy sets methodology.

Step 1: Follow the simple safety assess computational model 4):

\[
Risk = \sum_{hazard} \rho(hazard) \times \varepsilon(hazard)
\]

Risk represents the safety variable, \( \rho(hazard) \) and \( \varepsilon(hazard) \) represents the hazard likelihood, and possible consequences of the hazard respectively. For a safety fuzzy set, there are three basic parameters 5): failure likelihood (FL), consequence severity (CS), and failure consequence probability (FCP), which are usually used to assess the safety of a system with an event on a subjective basis.

The components safety degree can described by those parameters of fuzzy sets (respectively L, C, E means hazard likelihood and possible consequences). In the basis of fuzzy three quantities, then fuzzy safety degree can use three synthesis 12):

\[
S = C \circ E \times L
\]

\[
\mu_S = \mu_C \circ \mu_E \times \mu_L = (\mu_C^1 \circ \mu_E^1 \times \mu_L^1)
\]

where symbol \( \circ \) represents the composition operation, \( \times \) represent Cartesian product, \( \mu_S^j \) represents the safety degree belonging to the class \( j \), \( S \) and \( \mu_i \) represents a fuzzy description which under one of the failure mode leading by risk.
Step 2: Using the optimal fitting method, to calculate the safety $S$ and every safety expression Minkowski distance between two fuzzy sets, namely Euclidean distance:

$$d_y(S, H) = \sum_{i=1}^{n} (\mu_i^j - \mu_i^S)^2$$

In this formula $d_y$ represent the assurance degree of the fuzzy linguistic variables, the smaller the values, means the closer the fuzzy degree of the safety $S_i$ distance of the $j$ fuzzy language variables. In special circumstance, when $d_y = 0$, represents $S_j$ belong to the expression of $j$ fuzzy language variables, $H$ represents the safety expression. Set $d_j(J = 1, \ldots, n)$ is the smallest distance to $S_i$, then set $a_j$ represents the reciprocal of relative distance:

$$a_j = \frac{1}{d_j} (j = 1, \ldots, n)$$

Standard the $a_j$ as follow:

$$\beta_j = \frac{a_j}{\sum_{k=1}^{n} a_k} (j = 1, \ldots, n)$$

Step 3: We assume set $H = \{H_1, H_2, \ldots, H_n\}$ represent a serial set of language variables, which use to represent safety expression and assessment, $H_j$ represent the $j$ language variable, $n$ represent the total number of language variable ,then the final safety assessment to the component $i$ can described as:

$$S_{i(j)} = \{(H_j, \beta_j) | j = 1, 2, \ldots, n\}$$

In the formula, $\beta_j$ represent the confidence level of this assessment, and it satisfied with $\beta_j \geq 0, \sum_{j=1}^{n} \beta_j \leq 1$.When $\sum_{j=1}^{n} \beta_j = 1$, name $S_{i(j)}$ is safe assessment; when $\sum_{j=1}^{n} \beta_j < 1$, name $S_{i(j)}$ is unsafe assessment.

The Running Path and the System Safety Calculating Integrated

Assume subsystem in the running system own $i$ running path, definite $c_{i_k}$ is component $c^e$ associate with path $i$ , then the components set of running path $i$ can defined as:

$$P_i = \{c_{i_1}, c_{i_2}, \ldots, c_{i_n}\}$$

Let $m'_i = m(H_j / e_j)(m'_i \leq 1)$ be the real value, which represents a basic allocate function, this function is used to represent the $i$ operating path for the safety assessment of the value of the degree support for the hypothesis. Then gain $m'_i$ as below:

$$m'_i = \lambda_i \beta_i$$

Where the value of $\beta_i$ has been given by the component $c^e$.

The D-S theory is a kind of method to deal with evidence reasoning of uncertainty which can solve the synthesis problem of fuzzy and uncertain effectively when multiple factors need to be considered. In the D-S theory, a sample space is called a “frame of discernment”, defined as $\Theta$. A basic hypothesis (singleton) in $\Theta$ is denoted by $H_z$. In $\Theta$, all basic hypertheses are required to be mutually exclusive and exhaustive.

Follow the D-S theory. Given two different evidence mass function $m_1, m_2$, according to the D-S combination rule, can be combined with mass function, namely:

$$m(A) = m_1 \oplus m_2 = \begin{cases} 0, A = \phi & \text{if } k \neq \phi \\ \sum_{B \cap C = \phi} (m_1(B)m_2(C)) / 1 - k, A \neq \phi & \text{if } k \neq \phi \\ \sum_{B \cap C = \phi} m_1(B)m_2(C), k < 1 & \text{if } k \neq \phi \end{cases}$$
Suppose \( m_{j}^{k} \) is confidence level of component \( k \)'s safety degree which assess to \( H_{j} \), then \( m_{j}^{k} \) can integrate according to basic allocate function \( m(P / c_{j}) \) and the following comprehensive algorithm of evidential reasoning.

Suppose \( \varphi \) is sub-set of safety expression \( H \), \( \varphi \subseteq H \). Define sub-set \( p_{i} \) which is path set \( P_{i} \), and define combined probability allocate function \( m_{p_{i}(i)}^{e} \) as follow:

\[
P_{p_{i}(i)} = \{p_{1}, \cdots, p_{i}\}, 1 \leq i \leq p_{i}
\]

\[
m_{p_{i}(i)}^{e} = m(\varphi_{p_{i}(i)})
\]

Then the hierarchical algorithm steps are as follows (13) (14) :

\[
\{H_{j}\} : m_{l(i)}^{l} = K_{l(i)} \{m_{l(i)}^{l} + m_{l(i)}^{\mu} + m_{l(i)}^{m} \}, j = 1, \cdots, N
\]

\[
\{H_{j}\} : m_{l(i)}^{\mu} = K_{l(i)} m_{l(i)}^{\mu}
\]

\[
K_{l(i)} = [1 - \sum_{j=1}^{N} \sum_{l(i), \varphi_{l(i)}} m_{l(i)}^{\varphi}]^{-1}
\]

\[
n = 1, \cdots, n - 1
\]

To any \( \psi \subseteq H, m_{p_{i}(i)}^{e(\psi)} \) is the overall probability distribution function of \( \psi \), which confirmed by \( V_{k} \) and \( m_{p_{i}(i)}^{e(\psi)} = 0 \). So the safety degree of \( k \) th component \( c_{k} \) assess to \( H_{j} \) through a confidence level to \( (m_{j}^{k}) S_{(\psi)} = \{m_{j}^{k}, H_{j}\}, j = 1, \cdots, N \}

For any \( \varphi \subseteq H, m_{p_{i}(i)}^{e(\varphi)} \) is overall probability allocate function, which confirm by \( c_{n} \) and \( m_{p_{i}(i)}^{e(\varphi)} = 0 \). Thus component \( k \) safety degree of one software operation path will assess to \( H_{j} \) as a confidence level \( (m_{j}^{k}) \). This kind of assessment can realize through given safety degree of related path, express as follow:

\[
S_{(\varphi)} = \{m_{j}^{k}, H_{j}, i = 1, \cdots, N\}
\]

In the same way, each path safety assess value can calculate. A further problem is integrate the safety degree of the operation path which the component is in. Assume a number of \( l \) path exist in software system \( l \), then the set of software operation path transaction weight \( l \) define as follow:

\[
F_{l} = (p_{1}, \cdots, p_{l})
\]

In a software operation path the safety degree of component \( k \) assess through a confidence level to \( (m_{j}^{k}) \), then these component assess result can be consider to the evidence of the \( i \) operation path safety degree assess to \( H_{j} \) \( (j = 1, \cdots, i) \), the \( i \) operation path safety degree can be get through evidential reasoning theory. The problem converse to how to get \( m_{l}^{k} \) and calculate \( m_{l}^{k} \) \( (j = 1, \cdots, N; k = 1, \cdots, L) \). To solve these problems, we can just take \( e_{i} \) instead of \( c_{ik} \), and take \( \beta_{j} \) and \( m_{l}^{e_{i}} \) instead of \( m_{l}^{k} \) and \( m_{l}^{k} \). Then the safety degree of \( i \) software operating path can be assess by the follow formula:

\[
S_{(p_{i})} = \{m_{l}^{e_{i}}, H_{j}, j = 1, \cdots, i\}
\]

After these work, each safety expression of a subsystem of software have got.

Then the result of the subsystem weight segment to \( W = (w_{1}, w_{2}, \cdots, w_{k}) \), \( \sum_{j=1}^{k} w_{j} = 1 \), which get from the software operating analysis, comprehensive operation with the safety degree of each software operating path. Take \( w_{i} \) express in \( \lambda \) of each path, then the overall software safety degree is:

\[
S_{(\lambda)} = \{m_{l}^{e_{i}}, H_{j}, j = 1, 2, \cdots, N\}
\]

The result of \( S_{(\lambda)} \) can map into rule express of safety in DO-178B (5 level of software level). Finally we can realize the safety assessment of the software system.
A Case Study

In this paper a simple flight management system is employed to validate the method. The system consists of three degrees (rm). The subsystem consists of flight state selector (CDS), build in test (BIT) system and several other main modules.

Take $c_i$ expresses in Operating, $c_j$ expresses in startlog, $c_k$ expresses in BIT, $c_l$ expresses in RM, $c_m$ expresses in CDS.

And the running path of this architecture has been analyzed, the first running path is $c_1 - c_2 - c_3$; the second path is $c_1 - c_2 - c_4 - c_5$. Assume $p_1$ expresses the first path, $p_2$ expresses the second path. According to the operating profile analyzes the occurrences frequency of paths is 2, 1.

First we assume safety analyst of component $c_i$ may assign failure likelihood $L_i$, consequence severity $C_i$, and failure consequences probability $E_i$ as follows:

$L_i = \{1/0, 0.25, 3/1, 4/0.75, 5/0.6, 7/0\}$

$C_i = \{1/0, 0.2, 3/0.4, 4/0.75, 5/1.6, 25/0.25, 7/0\}$

$E_i = \{1/0, 0.2, 0.3, 0.4, 0.5/0.8, 6/0.2, 7/0\}$,

The fuzzy set of the safety description $S_i$ can be obtained as follows:

$S_i = C_i \times E_i \times L_i = \{1/0, 0.2, 0.25, 3/0.8, 4/0.75, 5/0.6, 7/0\}$, Mapping back to the safety expressions, the safety degree can be obtained as follows:

$S_i = (c_1\times c_2\times c_3) = \{(0.1452, \text{Catastrophic}), (0.1797, \text{Hazardous}), (0.2212, \text{Major}), (0.2235, \text{Minor}), (0.2304, \text{Noeffect})\}$

Follow the same way, the safety degree of component $c_2, c_3$ can be obtained as follows:

$S_{(c_2)} = \{(0.1375, \text{Catastrophic}), (0.1785, \text{Hazardous}), (0.1473, \text{Major}), (0.3213, \text{Minor}), (0.1151, \text{Noeffect})\}$

$S_{(c_3)} = \{(0.1195, \text{Catastrophic}), (0.1740, \text{Hazardous}), (0.1277, \text{Major}), (0.3557, \text{Minor}), (0.1231, \text{Noeffect})\}$

The path $p_1$ safety assessment $S_{(p_1)}$, associated with $S_{(c_1)}, S_{(c_2)}, S_{(c_3)}$ are obtained as follows:

Analysis the software operating path, the transact probability of $c_1, c_2, c_3$ is $[1, 0.5, 1]^T$ suppose $\lambda_1, \lambda_2, \lambda_3$ is calculate as $[0.995, 0.4975, 0.995]$.

$S_{(p_1)} = \{(0.1020, \text{Catastrophic}), (0.1130, \text{Hazardous}), (0.3207, \text{Major}), (0.3351, \text{Minor}), (0.1250, \text{Noeffect})\}$

Follow the same step, the safety degree of $p_2$ calculates as

$S_{(p_2)} = \{(0.1103, \text{Catastrophic}), (0.1051, \text{Hazardous}), (0.3002, \text{Major}), (0.3571, \text{Minor}), (0.1273, \text{Noeffect})\}$

Suppose $\lambda_{p_1}, \lambda_{p_2} = [2, 1]^T$, then the $\lambda_{p_1}, \lambda_{p_2}$ is calculate as $[0.9901, 0.4951]$.

The safety assessment associated with the software operating path is finally obtained as follows:

$S_{(\text{system})} = \{(0.1054, \text{Catastrophic}), (0.1101, \text{Hazardous}), (0.3148, \text{Major}), (0.3436, \text{Minor}), (0.1259, \text{Noeffect})\}$

According to the result, the safety level of this software system is level D, and the degree of confidence is 34.36%.

Acknowledgements

This research has been supported by the Defense Basic Research Program, and the fund ID is JCKY2013205C003.

References


