# The Application of Fault Tree Analysis in Software Project Risk Management

Xia zhang Benhai Yu Jinlong Zhang
(School of Management, Huazhong University of Science and Technology, Wuhan 430074, China)
E-mail:ybh68@163.com

***Abstract:*** *The fault tree model has great significance on software project risk management. According to the standard fault-tree model, this paper establishes the corresponding mathematical model and sets up the software fault tree model of software project, analyzes project risk probability and influence coefficient combined with the actual software project risk management; sequentially lays a theoretical foundation for better controlling software project risk management.*

***Key words: Fault Tree Model; Software Project; Risk Management***

## I. INTRODUCTION

The software development tools and technologies have made great progress in recent years. But the software project development cost overruns, schedule overtimes, can not satisfy user needs, therefore no practical use is still abound. There have long been all kinds of uncertainty, which have a serious impact on the completion and delivery of the projects, in software development and management. However, there is still a lack of adequate attention and systematic research regarding the risks [1]. Until the 1980's, Boehm discussed the risks of software development in detail, and proposed software risk management approach. Boehm believes that software risk management refers to the "as an attempt to formalize the risk-oriented correlates of success into a readily applicable set of principles and practices", its aim is to "identify, address, and eliminate risk items before they become either threats to successful software operation or major sources of software rework ".

Fault tree analysis (FTA) is widely used as a kind of system security analysis methods [2]. The method originated in the American Bell Telephone Laboratories. In 1961, Watson firstly proposed this method for the United States Air Force's Minuteman System security evaluation, and then A· B· Clemens et al had contributed to the prediction of missile launch accidents by improving the methods. Afterward, the Boeing Company reformed the FTA to use computer simulation extensively. In 1974, the U.S.A Atomic Energy Commission evaluated the fatalness of commercial nuclear power station incidents, published the famous Rasmussen report, and attracted attention from all over the world. At present, many countries are studying the application of this approach [3].

## II. CONSTRUCT THE FAULT TREE

Fault Tree Analysis (FTA) is an important method for analyzing and estimating system reliability and availability [4]. In the system design process, after analyzing various failure factors(such as hardware, software, environment, individual factors) , we can construct the logic diagram(viz. fault tree) , sequentially identify all causes with probabilities of system failures of 0 or more to predict the probability of system failure and take relative corrective measures to improve system reliability and security.

It's the key of analysis to construct the fault tree rightly. In the process of fault tree construction, an undesired effect is taken as the target of logic analysis, known as 'top event'; and then find out all the possible direct causes of this undesired effect, known as 'middle event'[5]; sequentially find out all the possible direct causes of these middle events, known as 'bottom event'[6]. The tree logic diagram, which is usually written out using corresponding logic gate symbols and representatives of the top events, middle events and bottom events, is called 'Fault Tree' [7]. In general, The construction of fault tree is divided into four steps: ① Select and obtain the top event. The top event is the undesired event, or the fault event of the logical analysis of appointed line. ②Analyze the top event. Find out all the direct, necessary and sufficient causes of the top event. Take the top event as the output event and all direct reasons as the input event, and interconnect these events using appropriate logic gates according to their actually logic relations. ③Analyze every input event directly linked to the top event. Take it as output event of the next level if the event can been further disassembled, as in the case of ②. ④Repeat the above steps, disassemble it down step by step until all the input events cannot been disassembled or need not break up, that is a complete inverted fault tree, as shown in Figure 1:

978-1-4244-4639-1/09/$25.00 ©2009 IEEE

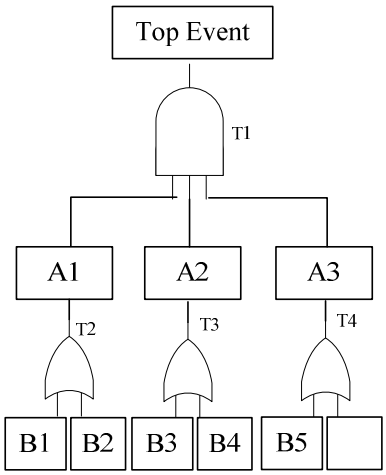


Figure1: The Fault Tree

## III. MATHEMATICAL MODEL OF FAULT TREE

Let x(t) be a state of the bottom event M(i＝1, 2, ⋯, n) at moment t, according to the distribution of '0-1', the bottom event can be defined as follows:

The bottom event B occurs at the moment t

$$x_i = \begin{cases} 1 & \text{bottom event } B_i \text{ occurs at the moment t} \\ 0 & \text{bottom event } B_i \text{ not occur at the moment t} \end{cases} \quad (1)$$

At the moment t, the occurrence probability of the bottom event equals the expectation of random event B.

$$p(t) = E[x(t)] = p[x(t) = 1] \quad (2)$$

In the fault tree, the state of the top event M must be a function of the bottom event B, we denote M[x(t)] as the state of the top event at the moment t, i.e.

The top event occurs at the moment t

$$M[x(t)] = \begin{cases} 1 & \text{top event occurs at the moment t} \\ 0 & \text{top event not occur at the moment t} \end{cases} \quad (3)$$

X(t)＝{x(t), x(t), ⋯, x(t)} is the time vector, the occurrence probability of the top event M at the moment t:

$$p_i = E\{M[x(t)]\} = p[\{M[x(t)] = 1\}] \quad (4)$$

We usually describe fault tree using the structural function to facilitate qualitative analysis and quantitative calculation. We can find the structural function of any given fault tree and take it as the combination of the AND and OR gates. The fault tree as shown in Figure 1:

$$\begin{aligned} T_4 &= B_5 + B_6 \\ T_3 &= B_3 + B_4 \\ T_2 &= B_2 + B_1 \\ T_1 &= T_2 T_3 T_4 \end{aligned} \quad (5)$$

If let p be the occurrence probability of the top event B, and we denote 'OR' and 'AND' as OR gate and AND gate respectively then

$$p_{AND} = \prod_{i=1}^{n} p_i \quad (6)$$

$$p_{OR} = 1 - \prod_{i=1}^{n} (1 - p_i) \quad (7)$$

## IV. EXPERIMENTS

The project schedule sometimes requires more stringent than cost estimate in software project management. Because the increased software cost can be compensated by improving product price or through sales in large quantities, however improper project scheduling will cause customer dissatisfaction and impact on the markets.

Take the Three Gorges Power Station automatic meter system improvement software project as an actual example, this paper constructs the fault tree model of the whole project schedule risk combined with the practical operation of the project, as shown in Figure 2.

China Yangtze Power Co., Ltd. Three Gorges hydroelectric power plant, whose annual average capacity of generating electricity is 84.6 billion kilowatt hours, is known as the world's largest hydro-electric power plant.

The Three Gorges hydroelectric power plant is the world's largest hydro-electric power plant by total capacity, which reaches 22,500MW. It has 34 generators. 32 are main generators, each with a capacity of 700 MW, and the other 2 are planted power generators, each with capacity of 50 MW. Among those 32 main generators, 14 of them are installed in the left side of the dam, 12 in the right side and the remaining 6 in the underground power plant. The automatic meter system of the Three Gorges Power Station uses DF6100 as the master station system and DF6201, CHL064-1j as collection terminal. It provides comprehensive coverage of all units of the Three Gorges hydro-electric power plant: generators exports, excitation variables, high-voltage transformer and 10KV system Watthour Meter. Its density of data acquisition and storage reaches 1 minute, and achieves accurate statistics data analysis of generator units' power output, power plant electricity consumption and line losses etc, and then provides comprehensive technical support to reduce

electricity price, enhance economic efficiency and reduce management loss.

The electric energy data on both sides of the Three Gorges have been collected and sent to the dispatch center and the country center. The remote terminals of the electric energy are Landis+Gyr company's products.

The measurement status quo of both sides of the Three Gorges and power generators' electricity exports and power plant electricity consumption as follows: the watt-hour meters send pulse volume to the monitoring system, and then the monitoring system accumulates electricity power, realize real-time monitoring. From the practical operation of the left side in recent years, it can be seen that the data of generators exports, excitation variables, high-voltage transformer, the monitoring system and the data of the Watthour Meter are consistent. In the past, the 10KV system Watthour Meter used mechanical meter on the left side, pulse signal have poor anti-jam capability, and hence the data from the monitoring system is not accurate. They have been replaced by electronic multifunctional energy meter now.

There are questions: on a daily basis, the rosters need to print data from the monitoring system screen before 24:00, and then fill out the daily and monthly statements manually. It is a heavy workload, moreover, cannot achieve accurate statistics analysis of electric energy.

According to the overall plan of the project, the system put into trial operation during mid-August 2009 and acceptance test during mid-September 2009

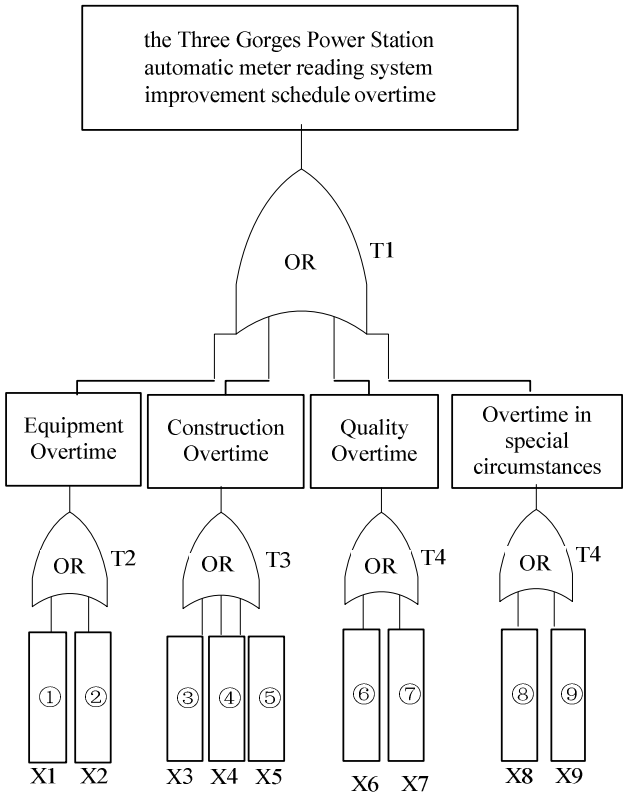
Case analysis as follows:

*A. The fault tree of the Three Gorges Power Station automatic meter system improvement schedule risk*

Based on the foregoing theory, the fault tree of the Three Gorges Power Station automatic meter system improvement schedule risk, as shown in Figure 2

*B. The probability of the bottom event progress overtime and important degree.*

The probability of the bottom event progress overtime and important degree, as shown in table 3:




①Procurement Overtime ②FAT Overtime ③Technological Backwardness ④Backward Management

⑤ Failure to Communicate ⑥ Hardware Quality ⑦ Software Quality ⑧ Complex Construction ⑨ Strict Acceptance Criteria

Figure 2: The Fault Tree Schematic diagram of the Three Gorges Power Station Automatic Meter System Improvement Schedule Risk

Table 3 The Probability of the Bottom Event Progress Overtime and important degree

| Bottom Event | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | X9 |
|---|---|---|---|---|---|---|---|---|---|
| Occurrence Probability | 0.01 | 0.03 | 0.01 | 0.1 | 0.2 | 0.02 | 0.2 | 0.2 | 0.01 |
| Impact Factors | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 1 | 1 |
| Influence Coefficient | 0.00429 | 0.01315 | 0.00429 | 0.04723 | 0.10626 | 0.00867 | 0.10626 | 0.10626 | 0.00429 |
| $I_i$ | 0.4293 | 0.4382 | 0.4293 | 0.4723 | 0.5313 | 0.4337 | 0.5313 | 0.5313 | 0.4293 |

### *C. The probability of the top event progress and the influence coefficient of the bottom event*

The fault tree of the project schedule risk is built using OR gates. According to the parameters of (7) and Table 3, we can calculate the probability of the project progress overtime.

$$p_{OR} = 1 - \prod_{i=1}^{n} (1 - p_i) = 0.595044 \qquad (7)$$

We denote $I_i$ as the probability importance degree of the bottom event. The definition is given as follows:

$$I_i = \frac{\tilde{\sigma}(p)}{\tilde{\sigma}^{p_i}} = \frac{1 - \prod_{i=1}^{n} (1 - p_i)}{\tilde{\sigma}^{p_i}} = \prod_{j=1}^{n} (1 - p_j)(i \neq j) \qquad (8)$$

For the bottom event with large impact factor and small probability of occurrence, we cannot see its influence on the entire project, so we used to define the influence coefficient F (u):

$$F_i(u) = I_i p_i \qquad (9)$$

According to Definition (8) and Definition (9), we can calculate the influence coefficient, as shown in table 3.

## V. RESULT ANALYSES

From the above-mentioned calculated analysis, it can be seen that lots of risk factors affect the whole project schedule. But the influence coefficients of various bottom events in the fault tree depend upon their probabilities of occurrence and impact factors, Moreover, it will influence the progress of the entire project once any bottom event occurs. Therefore, in order to insure that the progress of the entire project is in line with customer requirements, we must ensure that the progress of each event does not delay, especially focus on the bottom event with relatively large influence coefficients.

This paper supported by: Key Program of National Natural Science Foundation of China (No: 70731001); Surface Program of National Natural Science Foundation of China (No: 70571025) ; The plan project of Country safety production science and technology develops(No: 05-190); Science and Technology Research and Development Projects of Shandong Province (No: 2006GG2301002)

## REFERENCES

[1] Plexousakis, M. K. D. A formal framework for business process modeling and design[J]. Information Systems Research, 2006, 27(5) : 299-319.

[2]James.J.Jiang, G. K., Hsin-Ginn Hwang, Jack Huang, Shin-Yuan Hung. An exploration of the relationship between software development process maturity and project performance[J].Information & Management, 2004, 41 : 279-288.

[3]Fortune, J., White, D. Framing of project critical success factors by a systems model[J]. International Journal of Project Management, 2006, 24 : 53-65.

[4]IEEE.IEEE standard for software life cycle processes- risk management [R]. New York: IEEE Inc, 2001.

[5]Roy, S., Kalle, L., Mark, K., et al. Identifying software project risks: an international delphi study [J]. Journal of Management Information Systems, 2001, 17(4): 5-36.

[6]Kumar, R. L. Managing risk in IT project: an options perspective [J]. Information and Management, 2002, 40: 63-74.

[7]Erdogmus, H. Valuation of learning options in software development under private and market risk [J]. The Engineering Economist, 2002, 47(3): 308-353.