

## **A FUZZY LOGIC BASED METHOD FOR ANALYSING TEST RESULTS**

**Le Xuan Vinh\* and Tran Hong Quan**

Research Institute of Posts and Telecoms – PTIT  
122 Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam

Received 10 January 2006

---

### **ABSTRACT**

Network operators must perform many tasks to ensure smooth operation of the network, such as planning, monitoring, etc. Among those tasks, regular testing of network performance, network errors and troubleshooting is very important. Meaningful test results will allow the operators to evaluate network performance of any shortcomings and to better plan for network upgrade. Due to the diverse and mainly unquantifiable nature of network testing results, there is a need to develop a method for systematically and rigorously analysing these results.

In this paper, we present STAM (System Test-result Analysis Method) which employs a bottom-up hierarchical processing approach using Fuzzy logic. STAM is capable of combining all test results into a quantitative description of the network performance in terms of network stability, the significance of various network errors, performance of each function blocks within the network. The validity of this method has been successfully demonstrated in assisting the testing of a VoIP system at the Research Institute of Post and Telecoms in Vietnam.

The paper is organized as follows. The first section gives an overview of fuzzy logic theory the concepts of which will be used in the development of STAM. The next section describes STAM. The last section, demonstrating STAM's capability, presents a success story in which STAM is successfully applied.

### **1. FUZZY LOGIC THEORY**

Mathematical models have been helping us solve many problems, however, classical models often require assumptions which must be unambiguous, exact and complete. In real life, information that can be obtained is usually incomplete and uncertain, and yet right decisions must be made upon these information. To help with modelling under such circumstances, fuzzy set logic is developed as a flexible inference methodology [6].

#### **1.1 Fuzzy set**

Fuzzy set theory is an extension of the classical set theory. In this theory, any element in space  $U$  can belong to a fuzzy set  $\tilde{A}$  at some level of certainty. The level of certainty is presented by a function of the fuzzy set. A function of a fuzzy set  $\tilde{A}$  is denoted by  $\mu_{\tilde{A}}(x)$  where  $x$  is an element in space  $U$  that determines  $\tilde{A}$ . The values of a function is within a closed range  $[0,1]$ . Value of 0

---

\* Corresponding author e-mail: chuyenmach@gmail.com

means that  $x$  cannot belong to  $\tilde{A}$ ; whereas value of 1 means that  $x$  must belong to  $\tilde{A}$ . Values between 0 and 1 represents the uncertainty of an element  $x$  belonging to  $\tilde{A}$ . Hence, unlike the 0 or 1 mapping in classical set theory, fuzzy set function is a mapping  $\mu_{\tilde{A}}(x) : U \rightarrow [0, 1]$ .

A fuzzy set is represented as follows:

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in U, 0 \leq \mu_{\tilde{A}}(x) \leq 1\} \tag{1}$$

When space  $X$  is discrete, a fuzzy set if represented as:

$$\tilde{A} = \mu_1 / x_1 + \mu_2 / x_2 + \dots + \mu_n / x_n \tag{2}$$

where symbol "/" and "+" are not the normal algebraic operators;  $\mu_i / x_i$  meaning function  $\mu_i$  of  $x_i$  and "+" meaning the combination of all pairs  $(\mu_i, x_i)$ .

When space  $X$  is continuous, fuzzy set is represented by:

$$\tilde{A} = \int_x \mu(x) / x \tag{3}$$

A membership function represents the fuzzy part of the fuzzy set concept. Numerical values of a membership function is an convenient way to represent the relationship level of an element to a fuzzy concept. A membership function can also be thought of as a quantitative constraint for a fuzzy concept. Values of a membership cannot be exactly determined and often are subjective.

### 1.2 Fuzzy inference system

Fuzzy inference system performs inference to make decisions from ambiguous, incomplete and inexact. Fuzzy inference process has the following steps:

1. Fuzzyfication: determine base fuzzy sets and their membership functions.
2. Rule definition: determine composite rules based on the nature of an application and use them to combine base fuzzy sets.
3. Integration: Combine the composite rules.
4. Fuzzy range: fuzzy range of values of the resulting fuzzy sets.

Figure 1 illustrates this process.

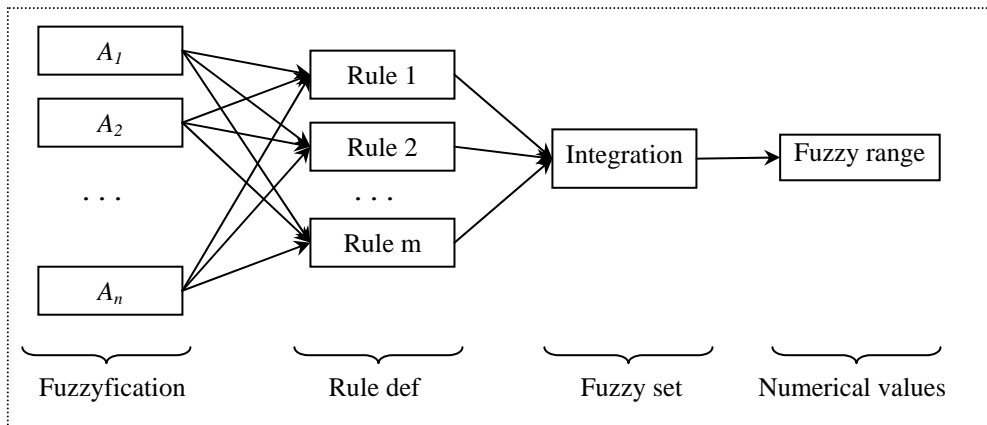
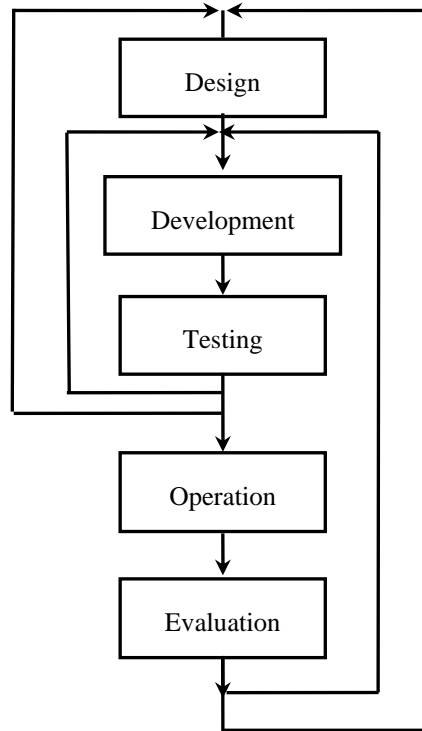


Fig. 1: Fuzzy inference system construction procedure

## 2. STAM

### 2.1 Network development life cycle

The lifecycle of a network consists of a number of stages which may include design, development, testing, operation, evaluation, as illustrated in Fig. 2.



*Fig. 2: Development life cycle of a typical system*

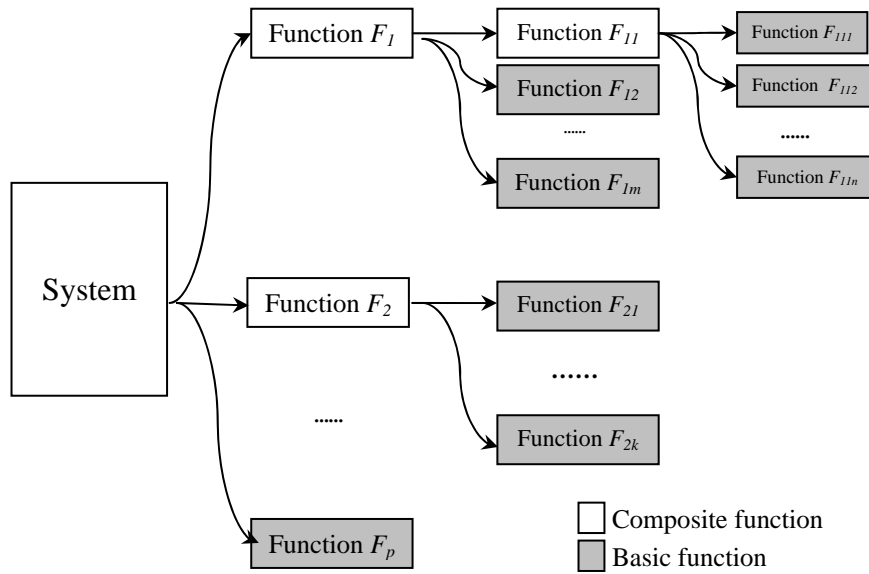
### 2.2 Test result analysis method

System testing during development and operation stages is repeated for each cycle, and its results must be converted into input to network development process that helps network operators to manage the network during its development.

For real life networks, test results are diverse and complex in nature. In many cases, test results are no more than a description of the errors detected during network operation. These results are mostly descriptive and qualitative. STAM is capable of processing these various results into something concrete and quantitative in 3 steps:

1. Determine, classify and define sources of errors.
2. Examine, build statistics for all errors.
3. Analyse all errors found in step 2, provide overall results.

To illustrate the STAM operation, let us examine a system shown in Fig. 3. Each function block of the system is considered as a potential source of errors. The sources are considered to be mutually independent.



**Fig. 3:** System functions

The test must be performed carefully with various test cases. The test cases are designed such that each and every function block would be activated, and must be repeated many times to unearth all possible errors. The errors must be logged in details. Denote the set of errors found during the test by  $E = \{E_i \mid i=1 \dots n\}$ , where  $n$  is a number of errors. Statistical report of the errors is presented in the format shown in Table 1.

**Table 1:** Error statistics

Error	Frequency of occurrence	Level of impact on the overall system	Duration of test
$E_1$	$Ne_1$	Description of error and its impact on the overall system operation.	Operating duration

As discussed earlier, the source of errors may involve different function blocks, hence, to determine the exact source of errors is a non-trivial task. Processing test results is also another concern as the errors are often descriptive and quantitative. In order to diagnose the source of errors, to evaluate the level of impact of various errors, and especially to come up with a quantitative measure of the system, STAM is designed to iteratively process each level using fuzzy logic inference.

Each function block must be tested and test results are collected according to the following principles:

- Input data is collected at each block.
- Process data from the bottom layer (basic functions) up to the top layer (system level).
- At each layer, data is processed independently for each composite block. The input data for a composite block is the data for all function block at the lower layer.

### 3. USING STAM FOR TEST RESULT ANALYSIS OF A VoIP SYSTEM

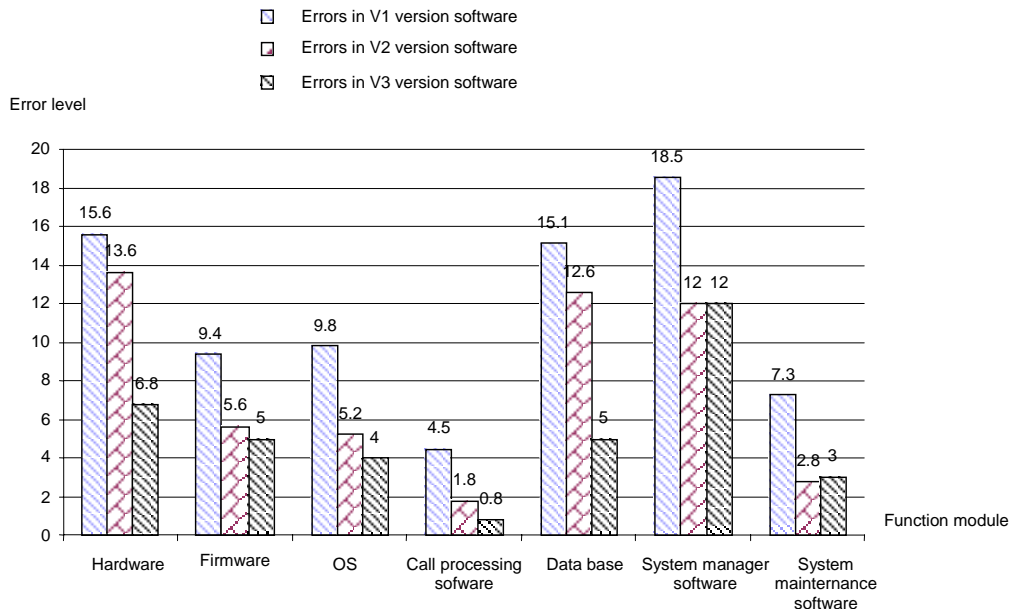
STAM method is used as a QA tool for a VoIP system development at RIPT. The test is done on three software versions denoted by  $V_1$ ,  $V_2$ , and  $V_3$ .

The whole system is broken down into 7 basic blocks which are further divided into sub modules at various level of details. The first level of the system hierarchy is summarised in Table 2.

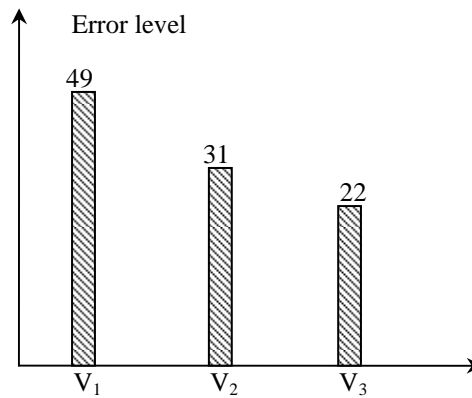
**Table 2:** Top level function blocks

Top level function blocks	
1	Hardware
2	Firmware
3	Operating system
4	Call processing applications
5	Database
6	System management applications
7	System maintenance applications

The test results using STAM method for the three versions are illustrated in Fig. 4 and Fig. 5. The former shows the level of development of each major function block in the system across different version. As can be seen from the graph, the number of errors is significantly reduced in V2 as compared to that of V1; the graph also shows great improvement on hardware, OS, call processing software and database management system in V3, while the remaining blocks do not received the same boost. Figure 5 reveals a continual enhancement of the overall quality as the number of errors in the system is steadily reduced in later version, which can be explained by the fact that all significant problems in earlier version have been resolved.



**Fig. 4:** Error lever of function blocks



*Fig. 5: Error level of the overall system*

#### 4. CONCLUSIONS

This paper has presented a new method to be used in analysing system test results. In this method, fuzzy inference system theory is applied in a bottom-up fashion. The method has been employed to evaluate and analyse the development of a VoIP system, which yields the following results:

1. Gives developers of each function block a prioritized list of errors found by the test.
2. Gives the system manager a overview of the state of development of the system and the improvement of each function block.

STAM method has therefore the potential to be used for network testing and network management.

#### REFERENCES

1. Bui Cong Cuong, Nguyen Doan Phuoc (2001), Fuzzy System, Neural Network and Applications. Science and Technology Publishing House, Hanoi, Vietnam.
2. John B. Bowles and Colon E. Pelaez (1998), Application of Fuzzy Logic to Reliability Engineering. *Proceedings of the IEEE*, vol. 83, no. 3, March.
3. Kazuo Tanaka and Hua O. Wang (2001), Fuzzy Control Systems Design and Analysis. A Wiley-Interscience Publication, John Wiley & Son, Inc..
4. Leon Osterweil (1981), A Strategy for Integrating Program Testing and Analysis. *Computer Program Testing*, North Holland, pp. 187-229.
5. Martin L. Katcher (1984), Integration and System Testing: a Methodology. ISS Session 14 C, Paper 1, pp. 1-5, 7-11 May.
6. Pham Xuan Minh, Nguyen Doan Phuoc (2002), Fuzzy Control Theory. Science and Technology Publishing House, Hanoi, Vietnam.
7. William E. Howden (1981), Errors, Design Procedures and Functional Program Tests. B. Chandrasekaran and S. Radicchi, *Computer Program Testing*, North Holland, pp. 115-127.
8. Witold Pedrycz, Fernando Gomide (1998), An Introduction to Fuzzy Sets. The MIT Press, Apr.