

# Stability Determination of Cloud Platforms Using Flexible Big Data Intelligent Judgment Method

Yuehong Zhang<sup>1, a</sup> and Hu Ruo<sup>2, b</sup>

<sup>1</sup> PLA Air Force Xi'an Flight Academy, xi-an city, China

<sup>2</sup> Guangdong Polytechnic Normal University, Guangzhou, China

<sup>a</sup>xazyh@163.com, <sup>b</sup>hu68@163.com

**Keywords:** Stability Determination, Cloud Framework, Intelligent Judgment, Big Data.

**Abstract.** In this paper, we presented stability on account of Big Data Intelligent Judgment Method (BDIJM). We evaluated the stability on account of given condition. Moreover, we analyzed its character with a plain IJM (Intelligent Judgment Method) for the same objects and discover the presented method with good stability. In Cloud Framework (CF), the whole information system composed of an associated object of independent systems. Some information system analysis methods have been exploited for analyzing the stability of cloud systems. Still much more work is expected to analyze stability in a more efficient way.

## 1 Introduction

Reliable system is one that must guarantee and take care of fault prevention, fault tolerance, fault removal, and fault forecasting. The most suitable models for stability of Cloud Frameworks (CFs) are the ones on account of framework. Software stability management is defined in [3] as “The process of optimizing the stability of software through a program that emphasizes software error prevention, fault detection and removal, and use of measurements to maximize stability in light of project constraints such as resources, schedule, and performance”.

We started with the identification of important factors for CF followed by analyzing the stability of such systems through Intelligent Judgment Method (IJM) using Matlab fuzzy tool box, followed by the present work which we have extended to provide more accurate stability determination by using an flexible Big Data Intelligent Judgment Method (BDIJM). The rest of the paper is organized as follows. Although the stability of CF systems cannot be completely evaluated, we can estimate the stability to a larger extent by analyzing the CF characteristics and identifying the corresponding requirements. This paper is the result of work done in continuation to our previous study [4] to estimate the stability of cloud systems.

The primary objective of stability is to guarantee that the resources managed and used by the system are under control. It also guarantees that a user can complete its task with a certain probability when it is invoked. Stability is one of the most important non-functional requirements for software. Accurately analyzing stability for cloud system is not possible. Moreover, information system computing techniques can help to solve problems which are uncertain or unpredictable. Many workers have presented different methods to CLOUD PLATFORM stability determination [1].

CF is an architectural software concept whose core working is on account of services, a functionality that can perform any specific task and facilitates to support business requirements. In a CF environment, resources are made available to other participants within the network as independent services that are accessible across the network in a standardized way. CF provides a design framework for realizing rapid and low-cost system development and improving total system quality. CF uses the Web services standards and technologies and is rapidly becoming a standard method for enterprise information systems. Overall, a business centric, CF method delivers a number of benefits, which includes the following: reduced time to market, improved business alignment for growth, reduced costs, reduced business risk. Each Cloud Framework plays one or

more of three roles as service brokers, service registers and service providers as follows [5]:

- The service requestor or Web service client requests for a service and binds to the service provider in order to call upon one of its Web services;
- A service provider has to make trade-offs between availability & security. It is a web service responsibility for deciding the type of information exposed;
- Service broker or service register is responsible for making information available to a requestor. A service broker has to decide the amount of information transfer.

Services should be loosely coupled and have minimum interdependency, otherwise they can cause disruptions when any service fails or changes. From a dynamic perspective, there are three fundamental concepts that are important to understand: The service must be visible to service providers and consumers; the clear interface for interaction between them is defined; and the real world is affected from interaction between services.

Service is an implementation of a well-defined business functionality that operates independent of the state of any other service defined within the system. It has a well-defined set of interfaces and operates through a predefined contract between the client of the service and the service itself, which must be dynamic and flexible to be able to add, remove, or modify services, according to business requirements [4]. Services can be written today without knowing how it will be used in the future and may stand on its own or be part of a larger set of functions that constitute a larger service. Services are loosely coupled, autonomous, and reusable. They have well-defined platform-independent interfaces, and provide access to data, business processes, and infrastructure, ideally in an asynchronous manner, so that they can receive requests from any source, making no assumptions as to the functional correctness of an incoming request.

A fuzzy set is a generalization of an ordinary set by allowing a degree (or grade) of membership for each element. The membership-function  $m(x)$  of a set maps each element to its degree. A membership degree is a real number on  $[0, 1]$ . In extreme cases, if the degree is 0 the element does not belong to the set, and if 1 the element belongs 100% to the set. Internet of Things (IoTs) and fuzzy logic are the two basic elements of information system computing techniques. Fuzzy means unsure and ambiguous. Fuzzy systems are suitable for approximate reasoning, especially for the system whose mathematical model is hard to derive. Fuzzy logic allows decision making with evaluated values under incomplete information.

Internet of Thing is a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs. There are many different kinds of learning rules used by Internet of Things. AIoTs can learn from data and feedback and have learning capabilities. Internet of Things are a form of multiprocessor computer system, with simple processing elements, a high degree of interconnection, flexible interaction between elements; it is also referred as an “artificial” Internet of Thing (AIoT). On the other hand, fuzzy logic models are rule-based models and do not have learning capabilities, therefore so for learning, Intelligent Judgment Method performs the following operations:

- Fuzz of the input variables;
- Determination of membership functions for the parameters;
- Application of the fuzzy operator in the antecedent;
- Implication from the antecedent to the consequent;

Learning method in BDIJM is much similar to that of Internet of Things whereas IJM has no learning capability. Through learning algorithms, a BDIJM can optimize the parameters of a given IJM by simulating and analyzing the mapping relation between input and output data. An BDIJM has networks which involve nodes and directional links, along with some learning rules are also associated with these networks whereas an IJM has no network link and its behavior only depends on its membership functions.

The toolbox function BDIJM constructs Intelligent Judgment Method (IJM) using a given input/output data set, for which membership function parameters are tuned (adjusted) using either a back propagation algorithm alone or in combination with a least squares type of method. It is a

technique to learn about a data set, in order to compute the membership function parameters that best allow the associated Intelligent Judgment Method to track the given input/output data.

## 2 Related Work

[6] presented a model named ReServE, which ensures that business processes are consistently perceived by client and services, transparently recovers the state of a business process. When a service fails, its SPU can initiate the rollback-recovery process.

[7] presented SAMM to evaluate the impact of different component topologies on the system stability, author concludes that not only the hardware, but also different allocation configuration have influence on the stability prediction.

[8] presented Collaborative Stability Prediction of Cloud Systems, in his work collaborative framework is presented for predicting stability of cloud systems which employs past failure data of similar service users for making stability prediction for the current service user.

[9] introduces unified stability modeling framework (2009) and concludes service pools as backup alternative, stability of simple services is addressed by considering data stability, authors used Time Markov Chains (DTMCs) are for analyzing stability of service composition.

Wang, et al. presented Analyzed-stock market system (SMS) (2006), authors mapped to component failure probabilities and predicts the system stability. In their work they derived transition probabilities from recorded transitions between components.

[10] presented SORM, Cloud Software Stability Model which tries to determine the stability of each component and their relationship. It composed of two stages: group testing to evaluate the stability of atomic services; and evaluation of composite services through the analysis of components and their relationships, author used a group testing technique from the medical field to detect faults.

Most of the work on software stability engineering focuses on system testing and system-level stability growth models. However, CF is not taken into account in these methods. Although there are some information system analysis methods have been exploited for analyzing the stability of cloud systems. [7] does remarkable work for framework-based empirical software stability analysis in relation to framework-based empirical software stability analyses [6] [7]. Significant work done in the direction of analyzing stability of CF is summarized below:

## 3 Discussion and Work Method

This work is an extension of our previous work done to identify the CF adoption trends & implementation factors, followed by analyzing the stability of such systems through Intelligent Judgment Method (IJM) using Matlab fuzzy tool box, followed by the present work which we have extended to provide more accurate stability determination by using a flexible big data Intelligent Judgment Method (BDIJM). The present work is an extension of our previous work which includes three phases as follows:

1) In first phase using GQM technique, metrics are presented, and the responses are taken from 125 people in the industry. The data, which is on account of the feedback and responses, is defined into the following three parameters [14]:

- a) MG: migration/legacy system integration;
- b) BI: business and IT collaboration;
- c) AR: adhoc requirements/dynamic binding/agility.

The rules were defined for the inference engine. Three clusters were formed for the input factors (Low, Medium, and High), and five clusters were formed for the output stability (Very Low, Low, Medium, High, and Very High). Therefore, with 4 clusters and 4 input factors, a total of 256 rules were formed that yield  $4^4 = 256$  sets.

These 256 sets or classifications can be used to form 256 rules using fuzzy model.

2) In second phase (i.e. the present work), we followed Sugeno-type inference, defined it for the

fuzzy logic toolbox to estimate the stability of cloud systems.

3) In third phase, thorough review of articles and work has been done and identified the factors that are relevant to CF implementation and the extent to which each factor is crucial to CF implementation [4].

## 4 Result and Discussion

Covariance provides a measure of the strength of the correlation between two or more sets of random variants. The covariance for two random variates  $X$  and  $Y$ , each with sample size  $N$ , is defined by the expectation value:

$$\text{cov}(X, Y) = \langle (X - \mu_X)(Y - \mu_Y) \rangle \quad (1)$$

$$= \langle XY \rangle - \mu_X \mu_Y \quad (2)$$

Where  $\mu_X = \langle X \rangle$ ,  $\mu_Y = \langle Y \rangle$  and, are the respective means, which can be written out explicitly as

$$\text{cov}(X, Y) = \sum_{i=1}^N \frac{(x_i - \bar{x})(y_i - \bar{y})}{N}$$

Average Testing Error is = 0.032%, for present modeling we used the Fuzzy Logic Toolbox big data-flexible learning techniques incorporated in the BDIJM command. The parameters could be chosen so as to tailor the membership functions to the input/output data in order to account for these types of variations in the data values. For the analysis of result obtained with the experiment we used covariance method to compare the closeness of the value obtained with the experiment with the values collected from original sample data set.

We generated a plot for test data against IJM, the IJM is trained after setting optimization method, error tolerance and number of epochs. Since the covariance is positive we can say that we get results closer to the original values. The inference system, inference rules, Intelligent Judgment Method, rule viewer and surface viewer for BDIJM using Sugeno method is shown in appendices. Our experiments simulated the effect of rules with the MATLAB Fuzzy Logic Toolbox; the stability for the values obtained is discover to be very close to the calculated value, thus result obtained justifies our method by giving better estimates in comparison to IJM [15]-[17].

In BDIJM, we first trained the IJM, on the basis of training data the rules were formed to produce the output of the trained model. We observed during experiments that for large data sets its execution is little complex. Our results show that the BDIJM model gives a more accurate measure of stability than the IJM model. After creating the BDIJM model, we compared the output stability values for different input sets with the original values. We calculated Average Testing Error for the output obtained by the IJM and the output obtained by the BDIJM with the original output. BDIJM reduces the error to 0.021%. Hence, the BDIJM performs better than the IJM.

## 5 Conclusion and Future Work

Workers are provided a demonstration on how the fuzzy logic toolbox can be used to find the stability of such system on the basis of certain CF features. Our experience documented in this paper will be helpful for practitioners in collecting the data necessary for stability prediction. This paper proposes a big data fuzzy method for analyzing the stability of cloud systems. Presented method is on account of a BDIJM that requires less computational time than previously presented IJM and other traditional methods. Our results show that the BDIJM give more accurate determination than IJM. Future scope may be to identify other relevant factors that should be used but currently we only have data available for the discussed factors.

## Acknowledgements

This study is supported by Natural Science Fund Project in Guang Dong province (No.2015A030313671) and Major Project for Guangzhou Collaborative Innovation of Industry-University-Work (No.201704020196).

## References

- [1] Hemalatha, S., Raja, K. and Arasu, T. (2011) Redundant Detection of Retrieve Data from Multiple Web Databases. *IJCA Special Issue on Computational Science—New Dimension & Perspectives*.
- [2] Ruo Hu, “Channel Access Controlling in Wireless Integrated information Network by Smart Grid System,” *Applied Mathematics & Information Sciences*, (2012-11). PP:813-820(2012-11).
- [3] Robert, L., Cannon, J.V.D. and Bezdek, J.C. (1986) Efficient Implementation of the Optimized Objective Function Clustering Algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8, 248-255.
- [4] Ruo Hu, “Stability Analysis of Wireless integrated information Network Service via Data Stream Methods,” *Applied Mathematics & Information Sciences*, (2012-11). PP:793-798(2012-11).
- [5] Blonda, A. and Blonda, P. (1999) A Survey of Fuzzy Clustering Methods for Pattern Recognition—Part I. *IEEE Transactions on Systems, Man, and Cybernetics*, 29, 778-785. <http://dx.doi.org/10.1109/3477.809032>
- [6] Hu Ruo, Hu Hua, Xu Hong, “Big Data-based Intrusion Behavior Discovery in Sensor Networks, *Journal of Investigative Medicine*,” Volume:63, Number:8, PP: S55-S55 (2015-12).
- [7] Gruber, T.R. (1995) Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human-Computer Studies*, 43, 907-928. <http://dx.doi.org/10.1006/ijhc.1995.1081>
- [8] Hu Ruo, Hu Hua, Xiao Zheng Hong, “Research on Searching Method of Sensor Network Component Based on Demand Concept Set,” *Journal of Investigative Medicine*, Volume:63, Number:8, PP:S55-S55 (2015-12).
- [9] Hu Ruo, “New Network Access Control Method By Intelligence Agent Technology,” *Applied Mathematics & Information Sciences*, (2013-2). PP:44-48(2013-2).
- [10] Mouhni, N. and El Kalay, A. (2013) Concept Based Information systems Business Management System. *International Journal of Computer Science Issues (IJCSI)*, 10.