

A Comparative Study of Mamdani and Sugeno Fuzzy Models for Quality of Web Services Monitoring

Mohd Hilmi Hasan, Izzatdin Abdul Aziz, Jafreezal Jaafar, Lukman AB Rahim, Joseph Mabor Agany Manyiel
Computer and Information Sciences Department,
Universiti Teknologi PETRONAS,
32610 Seri Iskandar, Perak, Malaysia

Abstract—This paper presents a comparative study of fuzzy inference system (FIS) with respect to Mamdani and Sugeno FISs to show the accuracy and precision of quality of web service (QoWS) compliance monitoring. We used these two types of FIS for designing the QoWS compliance monitoring model. Clustering validity index is used to optimize the number of clusters of both models. Then both models are constructed based on Fuzzy C-Means (FCM) clustering algorithm. Simulation results with a Mamdani model, a Sugeno model and a crisp-based model for benchmark are presented. We consider different levels of noise (to represent uncertainties) in the simulations for comparison and to analyze the performance of the models when applied in QoWS compliance monitoring. The results show that Sugeno FIS outperforms Mamdani FIS in terms of accuracy and precision by producing better total error, error percentage, precision, mean squared error and root mean squared error measurements. The advantage of using fuzzy-based model is also verified with benchmark model.

Keywords—Quality of web service (QoWS) monitoring; fuzzy inference system; QoS

I. INTRODUCTION

Quality of web service (QoWS) compliance monitoring is an important component in web service architecture as it evaluates whether or not services are delivered according to agreed performance. That means, it becomes a mechanism to detect requirements violations, and hence, can be used by users to decide whether or not to continue subscribing the services [1], [2], [3]. Furthermore, QoWS compliance monitoring also affects significantly the quality of service delivery in a way that it creates business commercial and user-provider relationship effects [4].

QoWS compliance monitoring is performed by comparing delivered QoWS with requirements defined in the agreement between a user and a provider. Existing QoWS compliance monitoring models are implemented based on precise logic computing and precise definition of QoWS requirements. In this paper, these precise computing and precise requirements definition are referred as crisp method.

An example of QoWS compliance monitoring model is SALMon [5]. The research on SALMon is carried out because it argues that previous works only developed systems for monitoring specific activities. Instead SALMon has the capability to monitor the whole service-based system's lifecycle. SALMon carries out its monitoring tasks based on crisp

method. Another work proposes a model for monitoring web service composition [6]. In the work, a monitoring model is proposed to monitor the process of combining web service functionalities in delivering service to users. The model is able to detect the violations of agreed composition requirements, formulate new composition requirements and select a new alternate service. This model carries out its monitoring tasks by using crisp method.

Another work also investigates the monitoring for service level agreement (SLA) violations in composite web service [7]. The work proposes a model that manages not only at instance level of web service composition, but also at the level of a group of instances. This monitoring functionality also includes QoWS monitoring. This model is similar to the above models in a way that it performs its QoWS monitoring tasks based on crisp method. In different context, Haiteng et al. (2012) proposes SLA derivation based on historical data to ensure the requirements represent recent actual QoWS values. This is done by QoWS monitoring model [8]. This model monitors the QoWS and supplies the information for SLA derivation. The framework proposed in the work is also based on crisp method.

Overall, the reviewed previous works show that the existing models perform their QoWS monitoring by crisp computation and crisp QoWS requirements definition. We argue that crisp method cannot effectively handle the uncertain nature of QoWS, hence, reducing the accuracy of QoWS compliance monitoring. These uncertainties have caused QoWS values to constantly change over time [9], [10], [11]. Crisp method is based on rigid QoWS requirement values, hence, it has less ability to handle these changes. For example, we conduct a preliminary experiment by clustering one type of QoWS parameter, namely availability, based on crisp algorithm (K-Means). Three clusters are used, namely, *Good*, *Moderate* and *Poor*. Table 1 shows the results, which indicate that the borders of *Good-Moderate* and *Moderate-Poor* clusters for the original dataset are 78.36% and 48.52%, respectively. The original dataset is then imposed with random errors in order to represent uncertainties. The dataset containing these uncertainties is known as synthetic dataset. Two errors are used, namely $\pm 0.5\%$ and $\pm 10\%$, hence two synthetic datasets are constructed. As shown in Table 1, the border values of both synthetic models are different from the original dataset described above. This means that, the level of QoWS cannot precisely be determined due to the occurrence of uncertainties.

TABLE I. CRISP CLUSTERING OF AVAILABILITY DATASET UNDER UNCERTAINTY

Dataset	Good-Moderate Cluster(%)	Moderate-Poor Cluster(%)
Original	78.36	48.52
Synthetic +0.5% error	77.64	47.74
Synthetic +10% error	77.43	49.67

For example, an availability value of 78.00% is considered as *Moderate* if it is based on the original dataset, but it is considered as *Good* if both synthetic datasets are used.

Another problem with crisp method is that QoWS requirements must be defined rigidly in SLA using precise values. However, generally, users are not aware of the realistic QoWS values [12], [13]. They may specify the expected QoWS values either lower or higher than the correct values, which will respectively result in getting poor service or not getting any matching service at all.

Hence, we proposed in our previous work a QoWS compliance monitoring model using fuzzy logic. The model has been found to be able to handle uncertainties better than that of crisp model, by producing monitoring results with high accuracy and precision. Furthermore, the model also allows users to specify their expected QoWS requirements using linguistic values such as “Response time is *Good*” and “Availability is *Moderate*”. Hence, it becomes more user-centric and solve the second problem described above.

In this research, we focus on improving our fuzzy model as described above by investigating its performance when different FISs are used. Specifically, the model is implemented using Mamdani and Sugeno FIS. Their performance in carrying out QoWS compliance monitoring tasks is evaluated in terms of accuracy and precision of the monitoring results. The performance of the two models are also compared with the benchmark model, which is developed using crisp method. In summary, the objectives of this paper are to present our methodology in conducting this research and to present the comparative study among the three models as described above.

The remaining of this paper is organized as follows. Section 2 contains the basic concepts of fuzzy and FIS. Section 3 contains the development of the model. Section 4 presents the experimental setups and evaluation parameters used in this research. Section 5 contains results and discussion. Finally, Section 6 concludes the findings of the paper.

II. BASIC CONCEPT OF FUZZY INFERENCE SYSTEM

A. Fuzzy Logic

Fuzzy logic is a soft computing method that involves uncertainty in carrying out its results inferring. The fundamental concept of fuzzy logic is that it converts a set of input into an output by using if-then fuzzy rules. These rules are evaluated on fuzzy sets. In contrast with classical set theory, this fuzzy set theory assigns elements with a partial set membership degree, which means that an element holds a value in the range between 0 and 1 [14]. This membership is known as membership degree.

Assuming X is the universe containing x objects, a fuzzy set F can be defined as follows [14]:

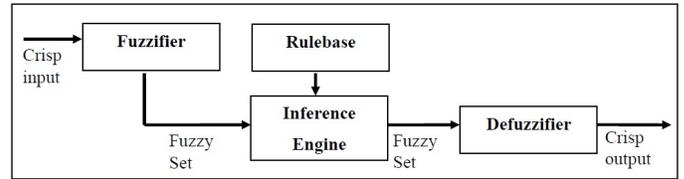


Fig. 1. Fuzzy inference system components.

$$F = (x, \mu_F(x)) | x \in X, \mu_F(x) \in [0, 1] \quad (1)$$

The notation $\mu_F(x)$ in (1) is an element’s membership degree in the universe X . As mentioned earlier, its value is ranging from 0 to 1.

B. Fuzzy Inference System

An FIS is a set of processes that applies fuzzy logic in mapping inputs to outputs. FIS comprises four components, namely fuzzifier, inference engine, rulebase and defuzzifier, as shown in Fig. 1.

The fuzzifier reads the inputs, which are normally crisp value, and evaluates their membership degree to each MF in the input MFs. This is done based on the fuzzification equation in (1). Then, the fuzzifier passes its fuzzifying results to the inference engine for rules inferring. The inference engine deduces results based on sets of rules from the rulebase. Two processes are performed by the inference engine, namely implication and aggregation. The former converts a fuzzified input into an output known as a rule’s consequent, while the latter sums up all the consequents since there is a possibility for a system to evaluate a number of rules. The aggregation process produces another fuzzy set, which will be defuzzified in order to transform it into a crisp value. This defuzzification process is carried out by the defuzzifier component by implementing centroid, largest of maximum, or some other methods.

There are two main types of FIS, namely, Mamdani and Sugeno. Both FIS are similar in many respects, for example, their fuzzifier process is the same. The main difference between the two FISs is that Mamdani’s output MFs are fuzzy sets, while Sugeno’s output MFs are either liner or constant. For example, rule such as *if A is R_1 , and B is R_2 , then C is R_3* is used in Mamdani FIS, where R_{1-3} are fuzzy sets. The same rule is implemented in Sugeno FIS, for example, as *if A is R_1 , and B is R_2 , then C is $ar1 + br2 + c$* . Numerous researches have been carried out to compare the performance of Mamdani and Sugeno FIS in the areas such as real-time scheduling system [15], prediction [16], antenna frequency [17] and water flow rate control [18]. In general, the motivations for conducting comparative study between Mamdani and Sugeno are to investigate their accuracy and computational efficiency. In this paper, we focus on the accuracy comparison between the two FISs in performing QoWS compliance monitoring.

III. DEVELOPMENT OF THE MODEL AND EXPERIMENTAL SETUP

Fig. 2 shows the research methodology, which contains activities involved in the development of the model and ex-

TABLE II. XB INDEX VALIDATION RESULTS

QoWS	Number of Clusters			
	2	3	4	5
Response time	2476.10	263.28	148.44	2365.10
Availability	18.81	15.82	59.28	16.04
Latency	6535.20	537.76	958.07	26566.00

perimental setup.

A. Dataset Preparation

The development of the model begins with identifying QoWS datasets. We use the datasets provided by Al-Masri and Mahmoud (2007) because it contains real QoWS data which were captured by their Web Service Crawler Engine (WSCE) [19], [20]. Moreover, the WSCE also has the capability to determine whether or not a QoWS data is valid, hence able to be monitored [19], [20]. In this reasearch, we use three types of QoWS data; latency, response time and availability. There are 1500 data points in each of these datasets. Response time and latency datasets contains data points of milliseconds (ms) in unit while the data points' unit in availability dataset are in percentage (%).

B. Clustering Validation

Based on the data sets, clustering validation is carried out to identify the optimal number of clusters for the FIS's MFs. A clustering validity index (CVI) is used to carry out this clustering validation process. Two considerations, namely compactness and separation, can be used by a CVI to determine the optimal number of clusters [21], [22]. Some CVIs use either one of these two considerations in their validation process. However, optimal clustering is reached when validation process uses both considerations and produces high degrees of compactness and separation. Due to this fact, we use Xie and Beni (XB) index in this research. Moreover, XB index is also selected because it is capable to perform well for the number of clusters' candidates is in the range 2 - 10 [23]. The candidate number used in this research is 2 - 5, which is within this range.

In this research, we use the candidates number of clusters from two to five. In XB index, the optimal number of clusters is determined by the minimum validation result value. Table 2 shows the XB index validation results for the three datasets used in this research. Based on the results, the optimal number of clusters is four for response time, and three for the other two datasets, availability and latency.

C. Data Clustering and FIS Construction

In general, there are two ways that can be implemented to develop MFs for an FIS, namely automatic development based on historical data or manual development using expert knowledge. It is found that the implementation based on expert knowledge may result in loss of accuracy [24] and may not always available [25]. Therefore, in this research, we develop the QoWS compliance monitoring model based on automatic approach using clustering of historical data.

We use Fuzzy C-Means (FCM) algorithm to cluster the QoWS datasets because its results can be used to construct

TABLE III. FCM RESULTS (CENTER OF EACH CLUSTER)

Cluster	Cluster Center Response time (ms)	Cluster	Cluster Center	
			Availability (%)	Latency (ms)
<i>Good</i>	174.44	<i>Good</i>	90.69	12.11
<i>Moderate High</i>	491.35	<i>Moderate</i>	65.43	95.86
<i>Moderate Low</i>	1438.44	<i>Poor</i>	28.12	392.20
<i>Poor</i>	3516.57			

both types of FIS, namely, Mamdani and Sugeno. As shown in Fig. 2, data clustering is needed before the MFs of the models can be constructed. FCM creates a number of clusters, and assigns a cluster's membership degree to each of data points [26]. To support its nature of fuzzy clustering, FCM assigns each data point to more than one cluster. These assignments to different clusters means that a single data point may fall in more than one cluster with different membership degrees. FCM clusters data by iteratively executing several processes, which eventually minimizing an objective function.

FCM requires the results of the optimal number of clusters shown in Table 2 as its inputs. Hence, in this research, the response time comprises *Good*, *Moderate High*, *Moderate Low* and *Poor* clusters. Meanwhile, the availability and latency consist of *Good*, *Moderate* and *Poor* clusters. FCM produces center of each cluster, which are shown in Table 3.

Furthermore, FCM also produces a matrix of membership degrees, U . This matrix contains each data point's membership degree of each of the clusters. This means that, there are *number of data points* \times *number of clusters* membership degree value for each dataset. Hence, in U , response time contains 1500×4 membership degree values, while availability and latency have 1500×3 membership degree values.

D. FIS Construction

The cluster centers, c (Table 3), and matrix of membership degrees, U , that are produced by FCM are used to construct input MFs of the QoWS compliance monitoring models. In this research, Gaussian-typed MF is implemented because its constructs match with the two outputs produced by FCM as mentioned above. Gaussian fuzzy sets used in input MFs is based on (2) as the following [27], [28]:

$$f(x; w, c) = e^{-\frac{(x-c)^2}{2w^2}} \quad (2)$$

MF width, w , is determined by solving (2) as follows:

$$w_{i=g,m,p} = \frac{\sum_{n=1}^{1500} \sqrt{-(X_n - C_i)^2} / (2 * \log(U_n))}{1500} \quad (3)$$

Or,

$$w_{i=g,mh,ml,p} = \frac{\sum_{n=1}^{1500} \sqrt{-(X_n - C_i)^2} / (2 * \log(U_n))}{1500} \quad (4)$$

where g , m and p in (3) denote *good*, *moderate* and *poor* clusters, and g , mh , ml and p in (4) respectively denotes *good*, *moderate high*, *moderate low* and *poor* clusters. Therefore, w

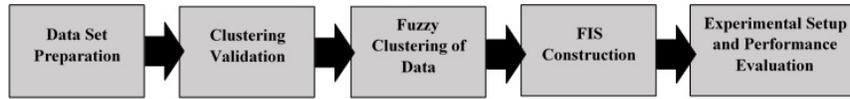


Fig. 2. Methodology of the research.

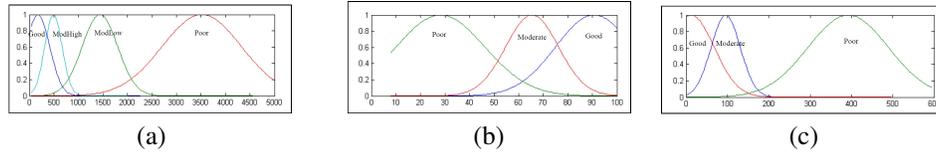


Fig. 3. Input MF of (a)Response time (b)Availability (c)Latency

of latency and availability data sets are generated based on (3), while (4) generates w values for response time data set.

Fig. 3 shows the three input MFs constructed using c values from Table 3 and w values from (3) and (4). Meanwhile, we implement 36 fuzzy rules in the proposed model. This number of fuzzy rules is chosen based on the number of clusters of each of QoWS parameters, i.e. $4 \times 3 \times 3$. All of these rules are shown in Fig. 4.

IV. EXPERIMENT SETUP AND PERFORMANCE EVALUATION

The experiment is conducted in Matlab simulation environment. The Mamdani and Sugeno models that are constructed based on the original QoWS datasets are known as main model, i.e. $Main_M$ and $Main_S$.

Then we construct 30 synthetic models for each Mamdani and Sugeno FISs. Synthetic models are constructed based on synthetic datasets, which are the original datasets that have been imposed with random errors. These random errors represent uncertainties in web service environment. For response time, six ranges of random errors are imposed, which are +5ms, +10ms, +20ms, +30ms, +40ms and +50ms. Similar to availability and latency, six ranges of random errors are imposed, which are +1%, +2%, +3%, +4%, +5% and +6% (availability) and +1.6ms, +1.7ms, +1.8ms, +1.9ms, +2.0ms and +2.5ms (latency). For each of these ranges, we construct five different synthetic datasets. Overall, there are 6 ranges \times 5 synthetic datasets, i.e. 30 synthetic models for each models. That means, we construct 30 Mamdani synthetic models and 30 Sugeno synthetic models.

Then all of the main models and synthetic models are executed to monitor a QoWS input dataset. This input dataset contains 27540 data points, which are 9180 data points of each response time, availability and latency. These input data points comprise values that have high probability to be evaluated differently by different monitoring models. That means, they are the data points that exist near to the area where two or more clusters intersect each other. This is shown in areas B and C in Fig. 5. These two areas have high probability for different monitoring models to generate different monitoring results as compared to area A. That means, the effects of uncertainties are most likely occurring in areas B and C.

The monitoring results between the main model and each of its synthetic models are compared to identify error, e . Hence,

assuming R is monitoring result, $Main$ is main model, Syn is synthetic model, e can be defined in the following:

$$e = \begin{cases} 0, & \text{if } R_{Main} = R_{Syn} \\ 1, & \text{otherwise} \end{cases} \quad (5)$$

Based on e , we compare the performance of the models in terms of accuracy, precision, mean squared error (MSE) and root mean squared error (RMSE). Accuracy is computed based on total error, $Accuracy_{TotalError}$, and percentage of error, $Accuracy_{ErrorPercentage}$ as the following:

$$Accuracy_{TotalError} = \sum e \quad (6)$$

$$Accuracy_{ErrorPercentage} = \frac{Accuracy_{TotalError}}{Number\ of\ data} \times 100 \quad (7)$$

Meanwhile, precision measures the robustness of the models under the state of uncertainties, by determining the closeness of two or more values to each other. In this research, precision is determined based on standard deviation of total errors generated from the comparison among the main and synthetic models' monitoring results [29], [30]. The following equation defines the precision measurement:

$$Precision = \sqrt{\frac{\sum (x - \bar{x})^2}{(n - 1)}} \quad (8)$$

where x is the total error as in (6); \bar{x} is the mean of total errors; and n is the number of sample.

Furthermore, MSE and RMSE are evaluated as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^n e^2 \quad (9)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n e^2} \quad (10)$$

1. If (*rt is Good*) and (*av is Good*) and (*lat is Good*) then (*output is Good*)
2. If (*rt is Good*) and (*av is Good*) and (*lat is Moderate*) then (*output is Good*)
3. If (*rt is Good*) and (*av is Moderate*) and (*lat is Good*) then (*output is Good*)
4. If (*rt is ModHigh*) and (*av is Good*) and (*lat is Good*) then (*output is Good*)
5. If (*rt is Good*) and (*av is Good*) and (*lat is Poor*) then (*output is Moderate*)
6. If (*rt is Good*) and (*av is Moderate*) and (*lat is Moderate*) then (*output is Moderate*)
7. If (*rt is Good*) and (*av is Poor*) and (*lat is Good*) then (*output is Moderate*)
8. If (*rt is ModHigh*) and (*av is Good*) and (*lat is Moderate*) then (*output is Moderate*)
9. If (*rt is ModHigh*) and (*av is Moderate*) and (*lat is Good*) then (*output is Moderate*)
10. If (*rt is ModHigh*) and (*av is Moderate*) and (*lat is Moderate*) then (*output is Moderate*)
11. If (*rt is ModLow*) and (*av is Good*) and (*lat is Good*) then (*output is Moderate*)
12. If (*rt is ModLow*) and (*av is Good*) and (*lat is Moderate*) then (*output is Moderate*)
13. If (*rt is ModLow*) and (*av is Moderate*) and (*lat is Good*) then (*output is Moderate*)
14. If (*rt is ModLow*) and (*av is Moderate*) and (*lat is Moderate*) then (*output is Moderate*)
15. If (*rt is Poor*) and (*av is Good*) and (*lat is Good*) then (*output is Moderate*)
16. If (*rt is Good*) and (*av is Moderate*) and (*lat is Poor*) then (*output is Poor*)
17. If (*rt is Good*) and (*av is Poor*) and (*lat is Moderate*) then (*output is Poor*)
18. If (*rt is Good*) and (*av is Poor*) and (*lat is Poor*) then (*output is Poor*)
19. If (*rt is ModHigh*) and (*av is Good*) and (*lat is Poor*) then (*output is Poor*)
20. If (*rt is ModHigh*) and (*av is Moderate*) and (*lat is Poor*) then (*output is Poor*)
21. If (*rt is ModHigh*) and (*av is Poor*) and (*lat is Good*) then (*output is Poor*)
22. If (*rt is ModHigh*) and (*av is Poor*) and (*lat is Moderate*) then (*output is Poor*)
23. If (*rt is ModHigh*) and (*av is Poor*) and (*lat is Poor*) then (*output is Poor*)
24. If (*rt is ModLow*) and (*av is Good*) and (*lat is Poor*) then (*output is Poor*)
25. If (*rt is ModLow*) and (*av is Moderate*) and (*lat is Poor*) then (*output is Poor*)
26. If (*rt is ModLow*) and (*av is Poor*) and (*lat is Good*) then (*output is Poor*)
27. If (*rt is ModLow*) and (*av is Poor*) and (*lat is Moderate*) then (*output is Poor*)
28. If (*rt is ModLow*) and (*av is Poor*) and (*lat is Poor*) then (*output is Poor*)
29. If (*rt is Poor*) and (*av is Good*) and (*lat is Moderate*) then (*output is Poor*)
30. If (*rt is Poor*) and (*av is Good*) and (*lat is Poor*) then (*output is Poor*)
31. If (*rt is Poor*) and (*av is Moderate*) and (*lat is Good*) then (*output is Poor*)
32. If (*rt is Poor*) and (*av is Moderate*) and (*lat is Moderate*) then (*output is Poor*)
33. If (*rt is Poor*) and (*av is Moderate*) and (*lat is Poor*) then (*output is Poor*)
34. If (*rt is Poor*) and (*av is Poor*) and (*lat is Good*) then (*output is Poor*)
35. If (*rt is Poor*) and (*av is Poor*) and (*lat is Moderate*) then (*output is Poor*)
36. If (*rt is Poor*) and (*av is Poor*) and (*lat is Poor*) then (*output is Poor*)

Fig. 4. Fuzzy rules for QoS compliance monitoring model.

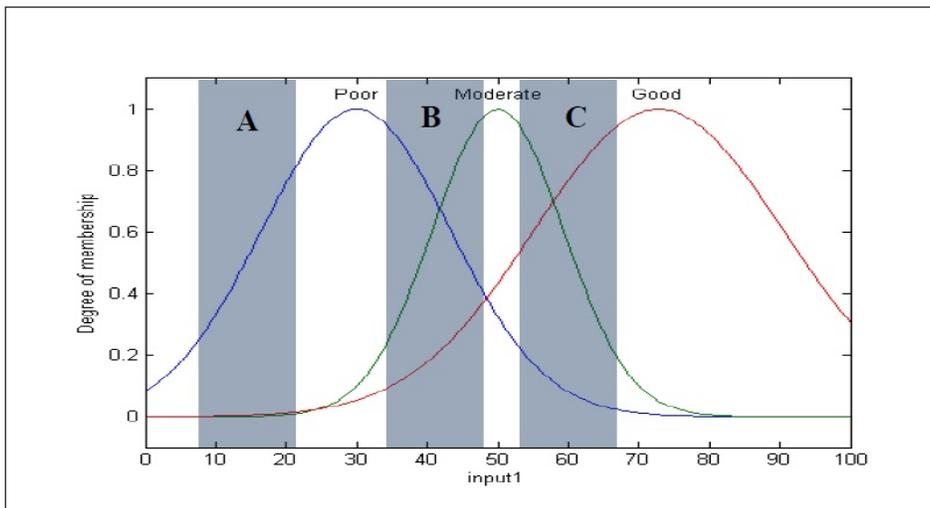


Fig. 5. Cluster intersection areas in input MFs.

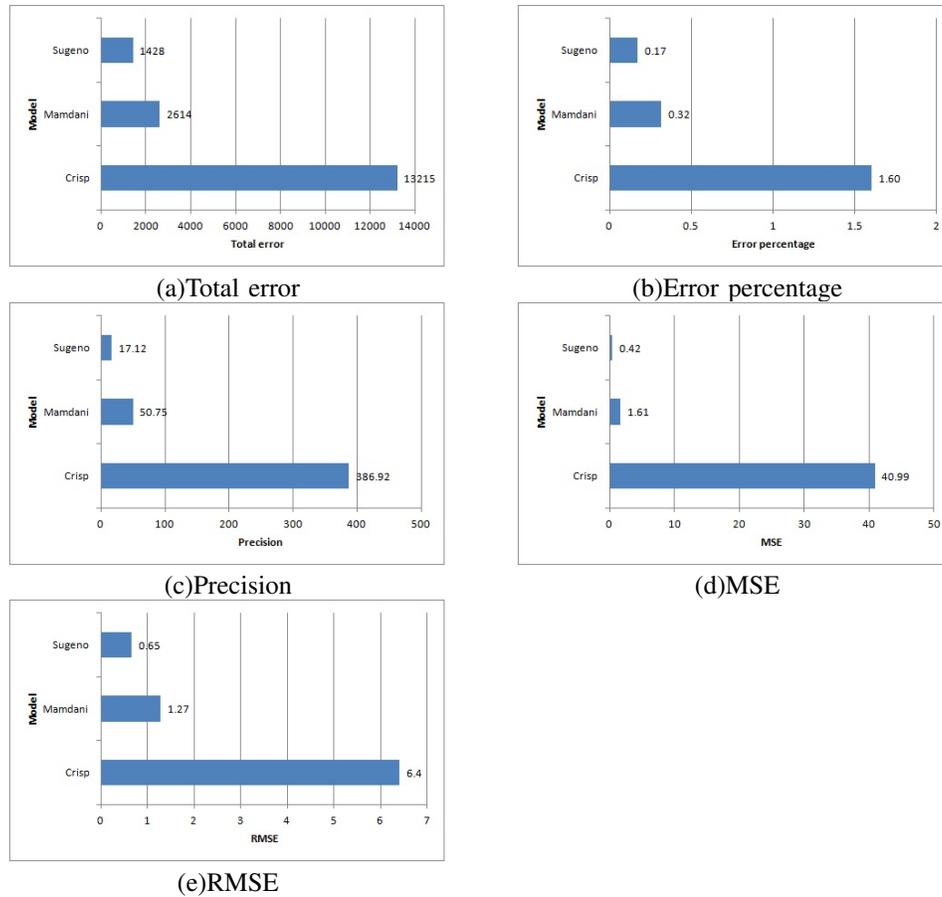


Fig. 6. Performance evaluation results.

V. RESULTS AND DISCUSSION

The comparative results of performance evaluation between Mamdani and Sugeno models are presented in Fig. 6. In all evaluations, the two models are also compared with the benchmark model that represents existing QoWS compliance monitoring model, namely, crisp model. Crisp model performs QoWS compliance monitoring based on hard computation, and it is constructed using K-Means clustering algorithm. This algorithm is selected because it works similar to FCM algorithm in performing data clustering. Unlike FCM, it produces rigid/hard clusters. The similarity of these two algorithms ensures the fairness in the conducted performance evaluations.

The results show that, in all of the five measurements, both of the fuzzy models produce better performance than the crisp model. These results support the theory; which states that fuzzy logic is tolerant of imprecision and uncertainty, hence can handle uncertainties better than crisp method. Less errors mean uncertainties are better handled, as shown in both fuzzy models. These results also support the findings of the previous research which argued that the use of crisp technique for QoWS compliance monitoring is unrealistic. This is because QoWS are uncertain, hence cannot be handled effectively by crisp technique.

Furthermore, the results also show that Sugeno model has outperformed Mamdani model in all of the five measurements. This shows that Sugeno structure is more robust than Mamdani

TABLE IV. PERFORMANCE EVALUATION RESULTS

Model	Total Error	Error Percentage	Precision	MSE	RMSE
Crisp	13215	1.60	386.92	40.99	6.40
Mamdani	2614	0.32	50.75	1.61	1.27
Sugeno	1428	0.17	17.12	0.42	0.65

in dealing with uncertainties. It is important for a QoWS compliance monitoring model to produce minimum number of errors under the state of uncertainties. This is because QoWS monitoring has become the reference for the users to evaluate the providers. Moreover, the results also suggest that Sugeno model is more stable than Mamdani model when performing monitoring task under the state of uncertainties. This can be seen from its smaller precision result than that of Mamdani, which shows that the the number of errors does not deviate much when different values of uncertainties are imposed. The summary of the performance evaluation results are shown in Table 4.

Overall, the experiments show that Sugeno model is not affecting by the noise (errors) as much as the Mamdani model. This shows that Sugeno is better than Mamdani especially in dealing with problems that contain high degree of uncertainty. Another advantage of a Sugeno FIS is that its consequents can have as many input parameters per rule. This allows more flexibility to developers to design Sugeno-based FIS. However, despite these advantages, Mamdani FIS is more widely used than Sugeno FIS. The main reason is Mamdani can offer good

results through a relatively simple structure. Mamdani FIS is also more intuitive in terms of rule base.

VI. CONCLUSION

This research involves a case study of three original QoWS datasets and 30 synthetic QoWS datasets. The synthetic datasets are based on the original datasets but they have been imposed with random errors in order to represent uncertainties. The original datasets are used to construct the main models of Sugeno, Mamdani and crisp. Similarly, the synthetic datasets are used to develop synthetic models of Sugeno, Mamdani and crisp. Overall, this research comprises a case study of 90 synthetic models (30 Sugeno, 30 Mamdani and 30 crisp models) and three main models.

This research shows that for this case study, Sugeno FIS does not only work better in terms of accuracy, but also performs better in terms of precision than Mamdani FIS. This shows that Sugeno FIS has better ability than Mamdani FIS to handle uncertainties by producing less number of errors and being more stable through better precision measurement results. Furthermore, this research also shows that Sugeno FIS outperforms Mamdani FIS in MSE and RMSE measurements, which shows that it is better in producing monitoring results that are close to the expected values. However, overall, both FIS have performed better than the benchmark model, i.e. crisp model in all of the five measurements, namely total error, error percentage, precision, MSE and RMSE.

To conclude, we may say that fuzzy-based model is better than crisp model in handling uncertainties in QoWS compliance monitoring. Moreover, between the two fuzzy-based models, Sugeno should be used whenever the QoWS compliance monitoring model is developed based on historical data clustering. For future work we are now studying the comparison of the different types of type-2 fuzzy FIS models as well as the different types of FIS's MFs, which are not only using Gaussian MFs, but also triangular and trapezoidal MFs.

REFERENCES

- [1] Liu, J-x., He, K-q., Wang, J., Ning, D.: A Clustering Method for Web Service Discovery. IEEE International Conference on Services Computing, 729-730 (2011)
- [2] Chhetri, M.B., Vo, Q.B., Kowalczyk, R.: AutoSLAMA policy-based framework for automated SLA establishment in cloud environments. Concurrency and Computation: Practice and Experience 27(9)(2013)
- [3] Palacios, M., Garcia-Fanjul, J., Tuya, J., Spanoudakis, G.: Coverage-Based Testing for Service Level Agreements. IEEE Transactions on Services Computing 8(2), 299-313 (2015)
- [4] Teixeira, M., Ribeiro, R., Oliveira, C., Massa, R.: A quality-driven approach for resources planning in Service-Oriented Architectures. Expert Systems with Applications 42 (12), 5366-5379 (2015)
- [5] Oriol, M., Franch, X. and Marco, J.: Monitoring the service-based system lifecycle with SALMon. Expert Systems with Applications 42(19), 6507-6521 (2015)
- [6] Shanmuga Priya, R., and Kanchana, R.: AOP based QoS monitoring of dynamic web service compositions. IEEE 2014 International Conference on Advanced Communication Control and Computing Technologies (ICACCT), 818-822 (2014)
- [7] Karthikeyan, J. and Kumar, M.S.: Monitoring QoS Parameters of Composed Web Services. 2014 International Conference on Information Communication and Embedded Systems (ICICES), 1-7 (2014)
- [8] Haiteng, Z., Zhiqing, S., Hong, Z. and Jie, Z.: Establishing Service Level Agreement Requirement Based on Monitoring. 2012 Second International Conference on Cloud and Green Computing (2012)
- [9] Dutta, M., Bhowmik, S. and Giri, C.: Fuzzy Logic Based Implementation for Forest Fire Detection Using Wireless Sensor Network. Advanced Computing, Networking and Informatics, 1 (The series Smart Innovation, Systems and Technologies), 319-327 (2014)
- [10] Martin, A., Lakshmi, T.M. and Venkatesan, V.P.: An information delivery model for banking business. International Journal of Information Management: The Journal for Information Professionals archive, 34(2) 139-150 (2014)
- [11] Prenesti, E. and Gosmaro, F.: Trueness, precision and accuracy: a critical overview of the concepts as well as proposals for revision. Accreditation and Quality Assurance, 20 33-40 (2015)
- [12] Benouaret, K., Benslimane, D., Hadjali, A., Barhamgi, M., Maamar, Z., Sheng, Q.Z.: Web Service Compositions with Fuzzy Preferences: A Graded Dominance Relationship Based Approach. ACM Transactions on Internet Technology, 13 (4)1-34 (2014)
- [13] Nagesha and Manvi, S.S.: QoS mapping from user to network requirements in WMSN: A fuzzy logic based approach. 2014 IEEE International Advance Computing Conference (IACC) (2014)
- [14] Zadeh, L.A.: Fuzzy sets. Information and Control, 8(3) 338-353 (1965)
- [15] Blej, M. and Azizi, M.: Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Fuzzy Real Time Scheduling. International Journal of Applied Engineering Research, 11(22) 11071-11-75 (2016)
- [16] Zaher, H., Kandil, A.E. and Fahmy, R.: Comparison of Mamdani and Sugeno Fuzzy Inference Systems for Prediction (With Application to Prices of Fund in Egypt), British Journal of Mathematics and Computer Science, 4(21) 3014-3022 (2014)
- [17] Guney, K. and Sarikaya, N.: Comparison of Mamdani and Sugeno Fuzzy Inference System Model for Resonant Frequency Calculation of Rectangular Microstrip Antennas. Progress in Electromagnetic Research B, 12 81-104 (2009)
- [18] Kansal, V. and Kaur, A.: Comparison of Mamdani-type and Sugeno-type FIS for Water Flow Rate Control in Rawmill. International Journal of Scientific and Engineering Research, 4(6) (2013)
- [19] Al-Masri, E. and Mahmoud, Q.H.: Discovering the best web service. 16th International Conference on World Wide Web (WWW)(2007)
- [20] Al-Masri, E. and Mahmoud, Q.H.: QoS-based Discovery and Ranking of Web Services. IEEE 16th International Conference on Computer Communications and Networks (ICCCN)(2007)
- [21] Berry, M.J.A. and Linoff, G.: Data Mining Techniques for Marketing, Sales and Customer Support. John Wiley and Sons, Inc., USA (1996)
- [22] Halkidi, M., Batistakis, Y. and Vazirgiannis, M.: On Clustering Validation Techniques. Journal of Intelligent Information Systems, 17 (2-3), 107145 (2001)
- [23] Pal, N.R. and Bezdek, J.C.: On cluster validity for the fuzzy c-means model. IEEE Transactions on Fuzzy Systems, 3 (3), 370-379 (1995)
- [24] Guillaume, S.: Designing Fuzzy Inference Systems from Data: An Interpretability-Oriented Review. IEEE Transactions on Fuzzy Systems, 9 426-443 (2001)
- [25] Jang, J-S.R.: Self-Learning Fuzzy Controllers Based on Temporal Back Propagation. IEEE Transactions on Neural Networks, 3 714-723 (1992)
- [26] Wang, L. and Wang, J.: Feature Weighting fuzzy clustering integrating rough sets and shadowed sets. International Journal of Pattern Recognition and Artificial Intelligence, 26(4)(2012)
- [27] Castillo, O. and Melin, P.: Design of Intelligent Systems with Interval Type-2 Fuzzy Logic. Type-2 Fuzzy Logic: Theory and Applications - Studies in Fuzziness and Soft Computing, 223 53-76 (2008)
- [28] Tay, K.M. and Lim, C.P.: Optimization of Gaussian Fuzzy Membership Functions and Evaluation of the Monotonicity Property of Fuzzy Inference Systems. 2011 IEEE International Conference on Fuzzy Systems (2011)
- [29] Prenesti, E. and Gosmaro, F.: Trueness, precision and accuracy: a critical overview of the concepts as well as proposals for revision. Accreditation and Quality Assurance 20(1) 33-40 (2015)
- [30] Wilrich, P-T.: Robust estimates of the theoretical standard deviation to be used in interlaboratory precision experiments. Accreditation and Quality Assurance 12(5) 231240 (2007)