

A Novel Financial Decision Support Systems Based on Mendel Type-2 Fuzzy Set Theory

G.S. Ng¹, C. Quek²

¹Information Systems Technology & Design, Singapore University of Technology & Design

²School of Computer Engineering, Nanyang Technological University, Singapore

Abstract— Hedging against potential market risk and maximizing profit from stock trading are some of the driving forces for the development of financial decision support system. Although presently many types of financial decision support available, there are still a lot of room for improvement. Hence, there is a significant importance to identify and develop a reliable financial decision support system with careful analysis on any critical issues. Accuracy has been the main goal of such decision support system, but it has been hampered by the biggest issue which is the uncertainty of the data. Fuzzy Neural Network (FNN) is a hybrid technique that combines fuzzy logic and neural network. FNN exploits the autonomy power of neural network, and therefore frees itself from the time-consuming process of knowledge base building. Despite all its advantages, FNN suffers from uncertainty when dealing with complex and highly uncertain event. The primary aim of this research work is to develop a financial decision support system based on in type-2 fuzzy set. This is based on the knowledge that Gaussian used in type-2 fuzzy set is a good alternative in reducing uncertainty of input data. The final application is an integrated FNN system, which has achieved the objective of reducing the uncertainty of stock market model. Extensive experimental results are presented for performance evaluation. Financial analysis foundations are also discussed in order to give a clearer prelude of the non-mathematical factor of this decision support system.

Keywords – Fuzzy Neural Network, Mendel Type-2 Fuzzy Set Theory, Decision Support Systems

I. INTRODUCTION

Fuzzy set theory was introduced to deal with problems involving knowledge where uncertainty exists. Classically, a set is defined by its members. An object may be either asserted or denied which is the characteristic of traditional (crisp) set. The connected logical proposition may also be true or false. This concept of crisp set may be extended to fuzzy set with the introduction of the idea of partial truth. Any object may be a member of a set 'to some degree'; and a logical proposition may hold true 'to some degree' between $[0,1]$. Fuzzy set theory offers a precise mathematical form to describe such fuzzy terms in the form of fuzzy sets of a linguistic variable.

Fuzzy set of type-2 is a fuzzy set where the membership grades themselves are fuzzy sets. Fuzzy set of type-2 allows for the idea that the members of a fuzzy set may not have a crisp value in the membership function but the degree of membership for the member is a fuzzy set. These seem to present an interesting approach to the problem of having to attach numeric grades to members of fuzzy sets and merit more detailed consideration than they have already been given.

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANN, like people, learns by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning

process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true for ANN as well.

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if" questions.

Accurate predictions of stock market indexes are important for many reasons. Among these are the need for the investors to hedge against potential market risks and the opportunities for market traders and arbitrageurs to derive profit by trading indexes. Clearly being able to accurately forecast stock market index has profound implications and significance to both researchers and practitioners.

II. LITERATURE REVIEW

Research in the area of neural network technique has shown that neural network possess the properties required for relevant applications, such as non-linear and smooth interpolation, ability to learn non-linear mappings, and self-adaptation for different statistical distribution (Ng, Wahab & Shi, 2003).

However, neural network cannot be used to explain the causal relationship between input and output variables. This is because of the essentially black box like nature of the many existing neural network algorithms. A neural

network cannot be initialized with prior knowledge and usually must learn from scratch. The learning process itself can take very long with no guarantee of success.

On the other hand, the fuzzy expert system approach has been applied to different forecasting problems, whereby the operator’s expert knowledge is used for prediction (Bolloju, 1996; Kaneko, 1996; Shaout & Al-Shammari, 1998). Although the fuzzy-logic based forecasting shows promising results, the process to construct a fuzzy logic system is subjective and depends on somewhat heuristic processes. The choices of membership functions and rule base have to be developed heuristically for each scenario. The rules fixed in this way may not always yield the best forecast, and the choice of membership functions still depend on trial and error.

With all these advantages and disadvantages of neural network and fuzzy logic, a neuro-fuzzy framework has emerged by combining the learning ability of the neural network and the functionality of the fuzzy system (Tan, Quek & Ng, 2007; Tan, Quek & Ng, 2005; Sharaf & Lie, 1995; Studer & Masulli, 1997; Padmakumari, Mohandas & Thiruvengadam, 1999).

Some researchers have made the interesting claim that the stock market is not an ordered system that can be explained by simple rules, nor it is a totally random system which no predictions are possible. In fact, they claim that the market is a complex system, in which only portions of the system’s behavior could be explained and predicted by a set of complex relationship among the variables.

Recognizing the complex characteristics of the stock market leads to the question if the predictability of various indexes could be improved by using models with endogenous learning capabilities in a fuzzy real world environment. The answer turns out to be ‘yes’, and the specific modeling approach used in this application demonstrates the advantages of neuro-fuzzy technique.

According to Tan (2015), Fuzzy Neural Network (FNN) is a hybrid technique that combines fuzzy logic and neural network. FNN exploits the autonomy power of neural

network, and therefore frees itself from the time-consuming process of knowledge base building. Despite all its advantages, FNN suffers from uncertainty when dealing with complex and highly uncertain event. Thus, it is proposed to implement Mendel Type-II fuzzy logic as the base of the fuzzy neural system.

The resulting FNN based on Type-II fuzzy logic is able to handle uncertainties. Type-II fuzzy logic provides an extra mathematical dimension, thereby enabling the uncertainties to be represented. It is believed that implementation of Type-II fuzzy set is a promising and competent candidate in realizing a better financial decision support system.

III. DEVELOPMENT OF THE FINANCIAL DECISION SUPPORT SYSTEM

A. Fuzzy logic system

A type-2 Fuzzy Logic System (FLS) as shown in Fig. 1 is constructed by the same structure of type-1 IF-THEN rules, which is still dependent on the knowledge of experts. Expert knowledge, however, is always represented by linguistic terms and implied uncertainty, which leads to the rules of type-2 FLSs having uncertain antecedent part and/or consequent part, then translate into uncertain antecedent or consequent membership functions. The structure of rules in the type-2 FLS and its inference engine is similar to those in type-1 FLS. The inference engine combines rules and provides a mapping from input type-2 fuzzy sets to output type-2 fuzzy sets. To achieve this process, union and intersection of type-2 sets are needed, as well as compositions of type-2 relations.

The output of the type-2 inference engine is a type-2 set. Using Zadeh’s extension principle (Zadeh, 1975), type-1 defuzzification can derive a crisp output from type-1 fuzzy set. Similarly, for higher type set such as type-2 fuzzy set, there is a process to derive the type-2 sets to a type-1 set. This process is called “type reduction” which will be detailed in a later section.

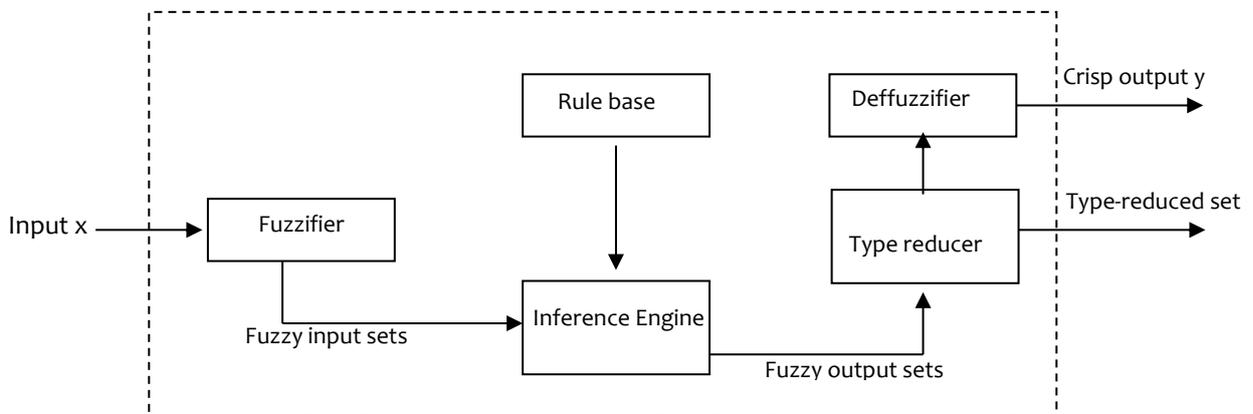


Fig. 1 Type-2 fuzzy logic system

B. Inference process for general type-2 fuzzy logic system

A structure of a single type-2 fuzzy rule is:

$$R: F_1 \times F_2 \times \dots \times F_n \rightarrow G \tag{1}$$

where R is the rule, F_i is the respective antecedents and G is the consequent. The type-2 fuzzy rule can be expressed by membership function as

$$\begin{aligned} \mu_R(\mathbf{x}, y) &= \mu_{F_1 \times F_2 \times \dots \times F_n \rightarrow \Theta}(\mathbf{x}, y) \\ &= \mu_{F_1}(x_1) \Pi \dots \Pi \mu_{F_n}(x_n) \Pi \mu_G(y) \\ &= \left[\prod_{k=1}^n \mu_{F_k}(x_k) \right] \Pi \mu_G(y) \end{aligned} \quad (2)$$

where Π denotes the meet operation. Type-2 union and intersection operations with its related join and meet operations are briefly explained as follows. Let two type-2 fuzzy sets be:

$$A = \int_{x \in X} \frac{\mu_A(x)}{x} = \int_{x \in X} \frac{\left[\int_{u \in J_x^u} \frac{f_x(u)}{u} \right]}{x} \quad J_x^u \subseteq [0,1] \quad (3)$$

and

$$B = \int_{x \in X} \frac{\mu_B(x)}{x} = \int_{x \in X} \frac{\left[\int_{w \in J_x^w} \frac{g_x(w)}{w} \right]}{x} \quad J_x^w \subseteq [0,1] \quad (4)$$

where u and w are just dummy variables used to differentiate between the different secondary membership (i.e. through membership functions f_x and g_x) of x in A and B , respectively.

The union of secondary membership function of the two fuzzy sets is

$$\mu_{A \cup B} = \int_{u \in J_x^u} \int_{w \in J_x^w} \frac{f_x(u) * g_x(w)}{u \vee w} \quad x \in X \quad (5)$$

where \vee means maximum and $*$ means minimum or product. The intersection of secondary membership function is

$$\mu_{A \cap B} = \int_{u \in J_x^u} \int_{w \in J_x^w} \frac{f_x(u) * g_x(w)}{u \wedge w} \quad x \in X \quad (6)$$

where \wedge (or $*$) means minimum or product.

C. Type reduction of general type-2 fuzzy logic system

The defuzzifier of type-1 FLS combines all fired output sets to obtain a crisp output set. For type-1 defuzzification, all the antecedent and consequent sets are type-1 set. However, for type-2 defuzzification, some or all of the antecedent and consequent sets are type-2 sets. Similar to the defuzzifier of type-1 FLS, type-2 FLS combines all type-2 rule output sets and then followed by type reduction to compute output set centroid.

The centroid of a type-1 fuzzy set A can be computed as follows:

$$C_A = \frac{\sum_{i=1}^N x_i \mu_A(x_i)}{\sum_{i=1}^N \mu_A(x_i)} \quad (7)$$

Similarly, for type-2 fuzzy set A , its domain is discretized into points with N equal intervals:

$$A = \frac{\sum_{i=1}^N \left[\int_{u \in J_{x_i}} \frac{f_{x_i}(u)}{u} \right]}{x_i} \quad (8)$$

Then the centroid of type-2 fuzzy set A can be computed as follows:

$$C_A = \frac{\left\{ \int_{\theta_1 \in J_{x_1}} \dots \int_{\theta_N \in J_{x_N}} [f_{x_1}(\theta_1) * \dots * f_{x_N}(\theta_N)] \right\} \sum_{i=1}^N \theta_i}{\sum_{i=1}^N x_i \theta_i} \quad (9)$$

Many applications of FLS to forecasting do not account for noise in training data. In forecasting, since antecedents and consequents are the same variable, the uncertainty during training exists on both the antecedents and consequents. If information about the level of uncertainty is available, it can be used to model antecedents and consequents as type-2 sets (Karnik, Liang & Mendel, 2001).

A fuzzy relation of higher type (e.g., type-2) has been regarded as one way to increase the fuzziness of a relation and, according to Hisdal (1981): "Increased fuzziness in a description means increased ability to handle inexact information in a logically correct manner."

D. Takagi-Sugeno fuzzy logic system

Takagi-Sugeno fuzzy architecture performs an identification starting from a set of N input-output samples. The Takagi-Sugeno fuzzy model (also known as the TSK fuzzy model) was proposed by Takagi, Sugeno and Kang (Sugeno & Kang, 1988; Takagi & Sugeno, 1985) in an effort to develop a systematic approach to generating fuzzy rules from a given input-output data set. A TSK fuzzy inference system is a set of rules. The first part (antecedent) of each rule is defined as a fuzzy and the second part (consequent) is a crisp function.

The characteristic of TSK where the output of its logic system is a crisp value is in accordance with the purpose of this decision support system that attempts to predict the actual value of the stock trading data. Based on the extension principle by Mendel there is no type-reduction needed for a type-2 TSK FLS, just as there is no defuzzification needed for a type-1 TSK FLS.

The TSK inference system uses the weighted mean criterion to combine all the local presentations as a global approximator:

$$y = \frac{\sum \mu^i y^i}{\sum \mu^i} \quad (10)$$

E. Fuzzy clustering

Clustering involves the task of dividing data points into homogeneous clusters so that items in the same class are as similar as possible. Clustering can also be thought of as a form of data compression, where a large number of samples are converted into a small number of representative prototypes or clusters. Depending on the data and the application, different types of similarity measures may be used to identify classes, where the

similarity measure controls how the clusters are formed. Some examples of values that can be used as similarity measures include distance, connectivity, and intensity.

In non-fuzzy or hard clustering, data is divided into crisp clusters, where similar data objects are assigned to the same cluster whereas dissimilar data objects should belong to different clusters. In fuzzy clustering, there is very often no sharp boundary between clusters so that the data points can belong to more than one cluster, and associated with each of the points are membership grades, which indicate the degree to which the data points belong to the different clusters (Hoppner, 2015).

Fuzzy clustering or cluster analysis divides data into groups (clusters) in order to improve understanding or to find structure in data. The most prominent fuzzy clustering algorithm is the fuzzy c-means (FCM), a fuzzification of K-Means.

The FCM algorithm is one of the most widely used fuzzy clustering algorithms. This technique was originally introduced by Bezdek (1981). Its advantages include a straightforward implementation, fairly robust behavior, applicability to multi-channel data, and the ability to model uncertainty of the data although it is quite sensitive to noise (Pham, 2001). The FCM algorithm attempts to partition a finite collection of elements $\mathbf{x} = (x_1, x_2, \dots, x_N)$ into a collection of c fuzzy clusters with respect to some given criterion.

Given a finite set of data, the algorithm returns a list of c cluster centers V , such that

$$V = V_i, \quad i = 1, 2, \dots, c \quad (11)$$

and a partition matrix U such that

$$U = U_{ij}, i = 1, \dots, c, j = 1, \dots, N \quad (12)$$

where U_{ij} is a numerical value in $[0, 1]$ that tells the degree to which the element x_j belongs to the i th cluster.

FCM is a more appropriate clustering technique for this work. Basic TSK architecture uses FCM as the clustering method, where it is designed to handle mostly a 'single input single output' function, whereas for stock data the input can be of multiple variables.

F. Structural parameters

A basic analysis of stock market excess return data shows both linear and non-linear dependence present and can be explained by simply allowing the mean and variance of Gaussian noise to be modulated.

The stock market is not random, it's not Gaussian. As anyone who has repeatedly put money at risk in a market over a long period of time knows, market behavior does not reflect the well behaved Gaussian models proposed by economists in the 1970s and 1980s. There are far more market bubbles and market crashes than these models suggest.

The change in market prices does not follow a Gaussian distribution in a reliable fashion. Like income distribution, market statistics frequently follow a power law. When a graph is made of market returns (e.g., profit and loss) the

curve will not fall toward zero as sharply as a Gaussian curve (Fama, 1963).

One example of a power law curve with "fat tails" is the Cauchy Gaussian distribution, which is shown in Fig. 2 (Johnson, Kotz & Balakrishnan, 1994). In the case of the normal curve, the tails are close to zero at -3.5 and 3.5. In the case of the Cauchy Gaussian distribution the curve is still not close to zero at -5 and 5.

For all the above facts, it shows that type-2 Gaussian fuzzy membership function would be one of the suitable candidates for the shape of the antecedents in analyzing the stock market.

When the secondary membership function of a type-2 fuzzy set is type-1 Gaussian membership function, the type-2 fuzzy set is called Gaussian type-2 set (Karnik, Liang & Mendel, 2001). Consider a primary membership function $\mu_{\tilde{A}_1}$ and a secondary membership function $\mu_{\tilde{A}_2}$, then they type-2 fuzzy set membership function can be expressed as,

$$\mu_{\tilde{A}}(x_1, x_2) = \mu_{\tilde{A}_1}(x_1) \prod \mu_{\tilde{A}_2}(x_2) \quad (13)$$

where both the membership functions are Gaussian.

As for the parametric form of the consequent model, it can be constant,

$$y_i = p_i \quad (14)$$

or it can be linear,

$$y_i = \sum_{j=1}^n p_{ij} x_j + p_{io} \quad (15)$$

where p_i contains the parameters of the consequent model of the rules.

G. Proposed system architecture

The proposed system architecture is shown in Fig. 3. Its detailed function of each layer is as follows:

- Layer-1 (input layer): No computation is done. Each node in this layer corresponds to one input variable.
- Layer-2 (fuzzification layer): Each node corresponds to one fuzzy set. The output link represents the

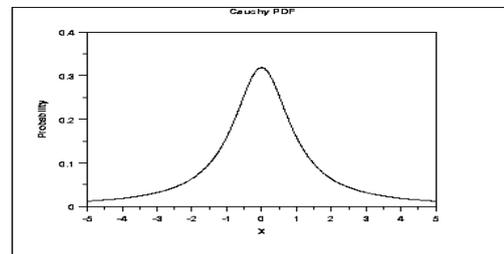


Fig. 2 Cauchy Gaussian distributions

- membership value, which specifies the degree of the fuzziness of an input relative to a fuzzy set.
- Layer-3 (rule antecedent layer): Each node represents the antecedent part of a rule. The output of this layer represents the firing strength of the corresponding fuzzy rule.

- Layer-4 (rule strength normalization): Each node calculates the ratio of the i th rule's firing strength to the sum of all rules firing strength.
- Layer-5 (rule consequent layer): Every node corresponds to a node function.
- Layer-6 (rule inference layer): The single node computes the overall output as the summation of all incoming signals.

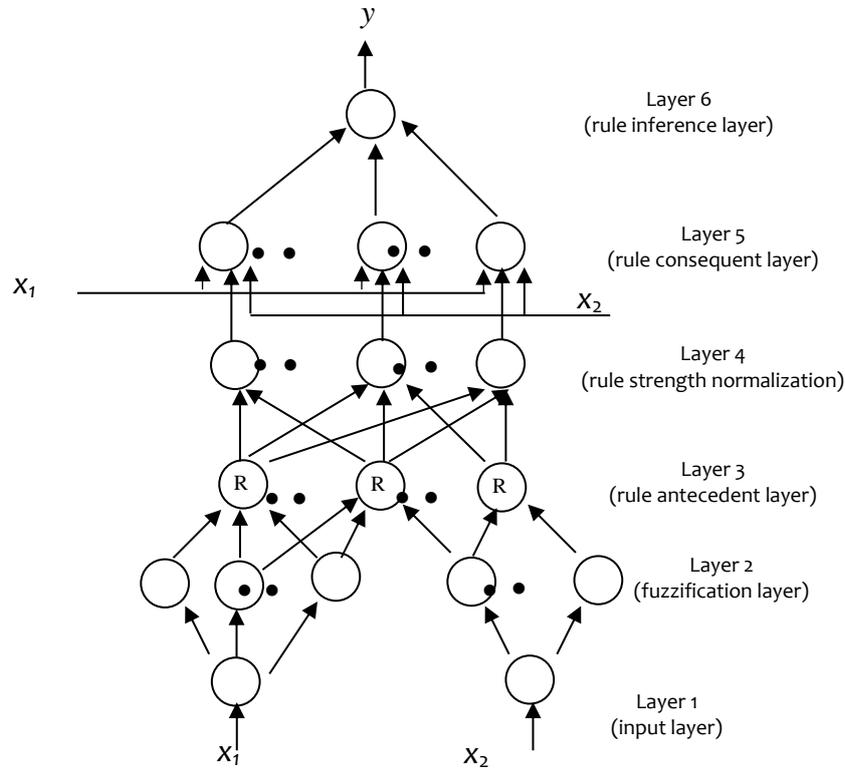


Fig. 3 TSK neuro fuzzy system

H. Performance evaluation

The performance measures used for benchmarking are Mean squared error (MSE), root mean-square error (RMSE) and coefficient of determination R^2 . MSE and RMSE provide a good measure of accuracy. R^2 accounts for the proportion of variability in a dataset of the forecasting system. In other words, it is a statistic that gives some information about the goodness of fit of a model. It is a measure of the linearity of the predicted and actual values. A correlation of +1 indicates a perfect positive correlation, which means that both the predicted and actual values move in the same direction together. Other than the above two measures, the number of rules generated by each model during the prediction process is used as an interpretability measure.

1) Mackey-Glass chaotic time series

Time series prediction is an important component in a diverse field of applications ranging from business planning to signal processing. The chaotic Mackey-Glass time series (Mackey, & Glass, 1977) is a commonly used as benchmarking in machine learning. The time series is generated using the following formula:

$$x_t = \frac{0.2x_{t-\tau}}{1+x_{t-\tau}^n} - 0.1x_t \quad (16)$$

where x_t is the value at time t and τ is a defined constant. The problem can be formulated as given values $\{x_{t-m+1}, x_{t-m+2}, \dots, x_t\}$ to determine x_{t+n} where m and n are fixed positive integers.

In this performance evaluation, three different time series are generated using $n=1, 2$ and 4 , with the following parameters $m=6$ and $\tau=17$. Each series consists of 1000 ordered pairs of data values (x_p, y_p) , where input data values are $x_p = \{x_{t-5}, x_{t-4}, \dots, x_t\}$ and the target value is $y_p = x_{t+n}$ for the p -th instance of data value. The 1000 data input-output pairs are partitioned into two sets: one set consisting of the first 700 data instances is used for training and one set containing the subsequent 300 data instances is used for testing. Thus the forecasted outputs for the testing dataset would be used for benchmarking the various models.

The proposed system in this paper, referred to as T2FS, is benchmarked against Mamdani type rough set neuro-fuzzy System (RNFS) (Ang, 2009), Mamdani type evolving fuzzy neural network (EFuNN) (Kasabov & Song, 2002), TSK type adaptive neurofuzzy inference system (ANFIS) (Jang, 1993) and dynamic evolving neuro-fuzzy inference system (DENFIS) (Kasabov & Song, 2002). Results are shown in Tables 1 and 2.

2) Discussion

Thus looking over the entire prediction process of T2FS, all the three chaotic series consist of 6 inputs and 1 output. For all the three series, 6 fuzzy clusters are identified for all the input variables by the self-organizing clustering technique of T2FS.

TABLE 1
RESULTS OF MACKEY GLASS TIME SERIES.

| System | 1-step | | | 2-step | | | 4-step | | |
|--------|---------|----------------|-----------------|--------|----------------|-----------------|--------|----------------|-----------------|
| | RMSE | R ² | Number of Rules | RMSE | R ² | Number of Rules | RMSE | R ² | Number of Rules |
| T2FS | 0.03025 | 0.992 | 85 | 0.0461 | 0.980 | 85 | 0.0957 | 0.907 | 85 |
| RNFS | 0.0519 | 0.956 | 53 | 0.0744 | 0.905 | 67 | 0.103 | 0.815 | 68 |
| EFuNN | 0.0291 | 0.987 | 113 | 0.0295 | 0.984 | 169 | 0.0478 | 0.975 | 235 |

However, for the output variable, 6 clusters are identified for 1-step series while 7 clusters for 2-step and 4-step series. Due to the approach adopted by the clustering method, the clusters identified are highly ordered and allow for meaningful label to be assigned to the clusters.

Analyzing the results among the Mamdani systems, based on R², T2FS outperformed all models for 1-step. EFuNN had the best results for the other two series, while T2FS came in a close second. Similar results can be observed when judging in terms of the RMSE measure (see Table 1). Across all 3 experiments, T2FS has an average of 29% less error than RNFS.

However, in terms of the number of rules generated and therefore interpretability, T2FS clearly outperforms the remaining models considering its high accuracy and yet fewer rules being generated. This result indicates the excellent use of Gaussian type-2 fuzzy set. T2FS generates 85 rules on an average as opposed to an average of 172 by EFuNN. Although, RNFS generates fewer rules than T2FS, results of T2FS exhibit higher accuracy.

On comparing T2FS with TSK models ANFIS and DENFIS, it is observed that the latter two outperforms T2FS (see Table 2). This can be attributed to the TSK nature of the ANFIS and DENFIS, which focuses on greater accuracy than Mamdani models. However, it is also worthwhile noting that due to the TSK nature of ANFIS and DENFIS, they are inherently less interpretable systems than T2FS.

The performance of T2FS is still comparable and accurate enough to be considered as one of the likely systems to model the Mackey-Glass dataset. Overall among Mamdani systems, T2FS came in either first or a close second in terms of the accuracy. However, along with such high accuracy, T2FS outperforms other models in

terms of interpretability with the least number of rules generated (excluding RNFS) as well as, identifying ordered semantic clusters from the raw data fed into the system.

TABLE 2
RESULTS OF MACKEY GLASS TIME SERIES AGAINST TSK SYSTEMS

| System | 1-step | | | 2-step | | | 4-step | | |
|--------|---------|----------------|-----------------|---------|----------------|-----------------|--------|----------------|-----------------|
| | RMSE | R ² | Number of Rules | RMSE | R ² | Number of Rules | RMSE | R ² | Number of Rules |
| T2FS | 0.03025 | 0.992 | 85 | 0.0461 | 0.980 | 85 | 0.0957 | 0.907 | 85 |
| ANFIS | 0.00479 | 1 | 13 | 0.00162 | 1 | 15 | 0.0581 | 0.999 | 17 |
| DENFIS | 0.00250 | 1 | 13 | 0.00135 | 1 | 13 | 0.0831 | 0.999 | 13 |

All Mamdani systems performs poorer than ANFIS and DENFIS, however TSK systems do not have semantic if-then fuzzy rules and are intrinsically less interpretable than Mamdani systems.

Fig. 4 shows the plot of the T2FS predicted values and the actual target values under the Mackey-Glass series testing dataset. It can be seen that the predicted values exhibit a very high correlation R² with the actual values. The increase in error as step size increases is due to more uncertainties in the data set.

IV. EXPERIMENTAL RESULTS

The major issue of the decision support system is the type of information, which should be included as the input data. Conventionally there are 2 methods used in implementing the input data for stock prediction purposes:

1. Use the closing price of previous n days, to predict the closing price of today transaction. This option would enable the input of several days of closing price information in the system (Lendasse, de Bodt, Wertz & Verleysen, 2000). This method also called the time series analysis.
2. Use the main information of stock trading which consist of opening price, highest price of the day, lowest price of the day and volume of trade. All the data were taken from previous trading day in order to predict the closing price of today transaction. This approach has the advantage in predicting the short term trends such as "profit taking" (Ritter, 2003).

In order to be able to provide a more prudent input data, the input should not consist of only the closing price data. However it is known that the time-series approach is a good way in providing a pathway for pattern recognition based on the financial market general fact where decision making is influenced by previous trading situation.

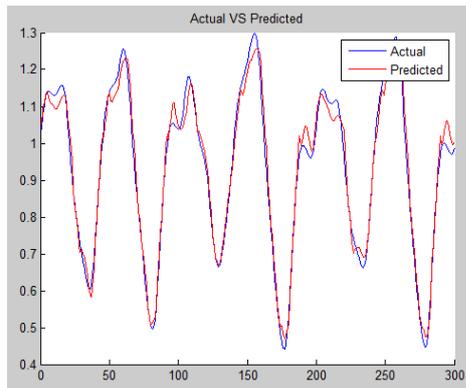


Fig. 4 (a): Predicted vs. actual values of T2FS: Mackey-Glass 1-step series.

Fig. 5(a) shows the setup of time series analysis data. The first 4 columns relate to the stock’s closing price of previous 4 days. The last column refers to the closing price of the current day of trade.

Fig. 5(b) shows the setup of stock characteristic analysis. The first 4 columns relates to opening price, lowest price, highest price, and volume of the trade. Usually this information was taken from the previous trading day and the last column refers to the closing price of the current day of trade.

| | 1 | 2 | 3 | 4 | 5 |
|----|--------|--------|--------|--------|--------|
| 1 | 26.86 | 26.895 | 27.385 | 27.9 | 27.12 |
| 2 | 26.895 | 27.385 | 27.9 | 27.12 | 27.905 |
| 3 | 27.385 | 27.9 | 27.12 | 27.905 | 27.96 |
| 4 | 27.9 | 27.12 | 27.905 | 27.96 | 28.195 |
| 5 | 27.12 | 27.905 | 27.96 | 28.195 | 28.485 |
| 6 | 27.905 | 27.96 | 28.195 | 28.485 | 28.135 |
| 7 | 27.96 | 28.195 | 28.485 | 28.135 | 27.675 |
| 8 | 28.195 | 28.485 | 28.135 | 27.675 | 25.73 |
| 9 | 28.485 | 28.135 | 27.675 | 25.73 | 25.665 |
| 10 | 28.135 | 27.675 | 25.73 | 25.665 | 25.5 |
| 11 | 27.675 | 25.73 | 25.665 | 25.5 | 26.14 |
| 12 | 25.73 | 25.665 | 25.5 | 26.14 | 24.925 |
| 13 | 25.665 | 25.5 | 26.14 | 24.925 | 24.585 |
| 14 | 25.5 | 26.14 | 24.925 | 24.585 | 24.41 |
| 15 | 26.14 | 24.925 | 24.585 | 24.41 | 24.955 |
| 16 | 24.925 | 24.585 | 24.41 | 24.955 | 24.12 |
| 17 | 24.585 | 24.41 | 24.955 | 24.12 | 23.73 |
| 18 | 24.41 | 24.955 | 24.12 | 23.73 | 24.28 |
| 19 | 24.955 | 24.12 | 23.73 | 24.28 | 23.66 |
| 20 | 24.12 | 23.73 | 24.28 | 23.66 | 23.48 |

Fig. 5 (a): Input data. Time series data.

The mean square error of the predicted output was used as a parameter to grade the performance of each selected technique. Table 3 shows the comparison of the result obtained from both methods.

The main reason why stock characteristic performs badly is that the system does not have adequate information to recognize the pattern of the stock’s price movement. In other words, for input to a fuzzy neural network, sufficient information must be provided in order for the system to be able to create a model to predict the pattern.

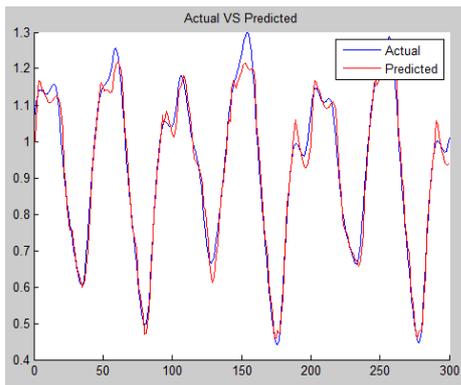


Fig. 4 (b): Predicted vs. actual values of T2FS: Mackey-Glass 2-step series.

| | 1 | 2 | 3 | 4 | 5 |
|----|--------|--------|--------|-------------|--------|
| 1 | 26.15 | 26.875 | 25.855 | 6.719e+007 | 26.895 |
| 2 | 26.795 | 26.9 | 26.44 | 5.5744e+... | 27.385 |
| 3 | 27.01 | 27.615 | 26.9 | 5.9628e+... | 27.9 |
| 4 | 27.46 | 28.005 | 27.34 | 8.0723e+... | 27.12 |
| 5 | 27.685 | 27.775 | 27.055 | 6.4742e+... | 27.905 |
| 6 | 27.36 | 27.96 | 27.265 | 6.2019e+... | 27.96 |
| 7 | 27.55 | 28.15 | 27.45 | 6.7973e+... | 28.195 |
| 8 | 28.262 | 28.375 | 27.889 | 6.1031e+... | 28.485 |
| 9 | 28.13 | 28.5 | 28.095 | 5.4401e+... | 28.135 |
| 10 | 28.5 | 28.66 | 28.095 | 5.9981e+... | 27.675 |
| 11 | 28.16 | 28.325 | 27.555 | 8.1786e+... | 25.73 |
| 12 | 26.47 | 26.5 | 25.655 | 1.5619e+... | 25.665 |
| 13 | 25.935 | 26.075 | 25.645 | 8.6403e+... | 25.5 |
| 14 | 25.795 | 26.2 | 25.455 | 8.7212e+... | 26.14 |
| 15 | 25.975 | 26.27 | 25.73 | 7.1616e+... | 24.925 |
| 16 | 26.015 | 26.025 | 24.851 | 1.0341e+... | 24.585 |
| 17 | 24.66 | 25.3 | 24.205 | 1.1517e+... | 24.41 |
| 18 | 24.845 | 24.85 | 24.281 | 8.6409e+... | 24.955 |
| 19 | 24.365 | 25.025 | 23.965 | 1.0722e+... | 24.12 |
| 20 | 25.08 | 25.085 | 24.095 | 8.8164e+... | 23.73 |

Fig. 5 (b): Input data. Stock characteristic data.

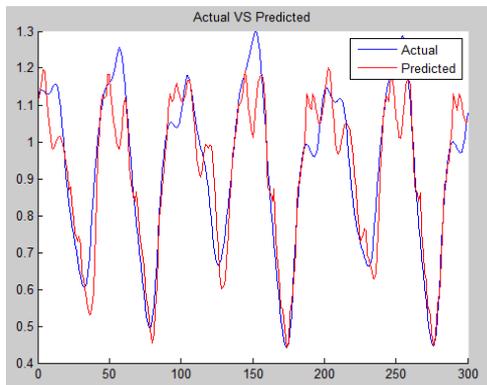


Fig. 4 (c): Predicted vs. actual values of T2FS: Mackey-Glass 4-step series.

TABLE 3
COMPARISON RESULT OF MEAN SQUARE ERROR.

| Month | Time-Series | Stock Characteristic |
|----------|-------------|----------------------|
| January | 0.2164 | 3.8759 |
| February | 0.1872 | 1.6489 |
| March | 0.1615 | 0.1536 |

From the results, it does not mean that stock characteristic cannot work in fuzzy logic system. With small changes, it is still a powerful method that worth considering. For example, both methods were collaborated by using the closing price of 3 previous days and the volume of previous day trading. Trade volume was chosen as the incorporated data because it has always been seen as one of the major indicator in determining the

trend of stock's movement. Fig. 6 shows the setup of the data.

| | 1 | 2 | 3 | 4 | 5 |
|----|--------|--------|--------|-------------|--------|
| 1 | 26.86 | 26.695 | 27.385 | 5.9628e+... | 27.9 |
| 2 | 26.695 | 27.385 | 27.9 | 8.0723e+... | 27.12 |
| 3 | 27.385 | 27.9 | 27.12 | 6.4742e+... | 27.905 |
| 4 | 27.9 | 27.12 | 27.905 | 6.2019e+... | 27.96 |
| 5 | 27.12 | 27.905 | 27.96 | 6.7973e+... | 28.195 |
| 6 | 27.905 | 27.96 | 28.195 | 6.1031e+... | 28.485 |
| 7 | 27.96 | 28.195 | 28.485 | 5.4401e+... | 28.135 |
| 8 | 28.195 | 28.485 | 28.135 | 5.9981e+... | 27.675 |
| 9 | 28.485 | 28.135 | 27.675 | 8.1786e+... | 25.73 |
| 10 | 28.135 | 27.675 | 25.73 | 1.5619e+... | 25.665 |
| 11 | 27.675 | 25.73 | 25.665 | 8.6403e+... | 25.5 |
| 12 | 25.73 | 25.665 | 25.5 | 8.7212e+... | 26.14 |
| 13 | 25.665 | 25.5 | 26.14 | 7.1616e+... | 24.925 |
| 14 | 25.5 | 26.14 | 24.925 | 1.0341e+... | 24.585 |
| 15 | 26.14 | 24.925 | 24.585 | 1.1517e+... | 24.41 |
| 16 | 24.925 | 24.585 | 24.41 | 8.6409e+... | 24.955 |
| 17 | 24.585 | 24.41 | 24.955 | 1.0722e+... | 24.12 |
| 18 | 24.41 | 24.955 | 24.12 | 8.8164e+... | 23.73 |
| 19 | 24.955 | 24.12 | 23.73 | 1.0716e+... | 24.28 |
| 20 | 24.12 | 23.73 | 24.28 | 8.078e+007 | 23.66 |

Fig. 6 Combined method data.

This kind of arrangement is just one of many possible arrangements between stock characteristic and time-series analysis method. Major disadvantage of this combined method is that the computational could be very expensive if best result is desired.

TABLE 4
COMPARISON RESULT OF MEAN SQUARE ERROR.

| Month | Time-Series | Combined Method |
|----------|-------------|-----------------|
| January | 0.2164 | 0.1859 |
| February | 0.1872 | 0.1441 |
| March | 0.1615 | 0.1582 |

For example, incorporating 4 days of complete characteristic data would means 16 variables of input. The model of this combined method was then used to predict the same data used previously in time-series and stock characteristic analysis. Table 4 shows the result of the prediction.

Hence it is proposed that the input for this decision support system should compromise both methods. It should also be arranged that for each vector, consists of more than 1 day of trading data as well as other selection of variables (in this case volume is taken as the extra measure). As for the volume, general practice only considered the last day of the trading since most trends that affected volume will be reflected on the last day of trading. This is based on the consideration that for each input vector, the time-series structure is short (2-5 trading days).

A. Microsoft Stock

For this decision support system, the historical stock information of Microsoft Corporation was chosen as the input data. The main reason for this decision was due to the high volume of trading and significance of price fluctuation. This condition would help in testing whether the system able to predict the trend of the stock's price movement.

B. Model selection: number of rules

Consider a training data with 3 linguistic variables and 1 output. There can be many possibilities of rules in term of linguistic approach shown in Table 5.

TABLE 5
FUZZY RULES.

| LINGUISTIC VARIABLE 1 | LINGUISTIC VARIABLE 2 | LINGUISTIC VARIABLE 3 | OUTPUT |
|-----------------------|-----------------------|-----------------------|--------|
| ¬high | high | medium | low |
| Low | ¬low | low | low |
| - | low | low | medium |
| - | medium | medium | medium |
| low | - | high | medium |
| ¬high | low | medium | medium |
| ¬low | medium | low | medium |
| medium | ¬low | high | medium |
| High | high | ¬high | medium |
| high | low | medium | high |
| high | medium | high | high |
| ¬low | low | high | high |

The main purpose is to check the optimal number of rules to maximize the performance of the fuzzy neural system. In fuzzy neural, the number of rules is actually the number of fuzzy clusters of the input data. This number will be used to perform the fuzzy clustering for the creation of the fuzzy model.

For actual data selection, the time-series was based on 6 days of stock's information with last day volume characteristic data. It resulted in 7 variables of input data and 1 variable of output data. The whole data consist of 504 of trading days. Fig. 7 shows the data selection.

Due to the nature of neural network, the training with so many input data took a significant amount of time (Lippmann, 1987). Fig. 8 shows the number of rules needed.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----|--------|--------|--------|--------|--------|--------|-------------|--------|
| 1 | 33.52 | 34.615 | 34.45 | 34.28 | 34.69 | 34.355 | 7.2992e+... | 34.64 |
| 2 | 34.615 | 34.45 | 34.28 | 34.69 | 34.355 | 34.64 | 5.091e+007 | 34.305 |
| 3 | 34.45 | 34.28 | 34.69 | 34.355 | 34.64 | 34.305 | 6.0714e+... | 34.235 |
| 4 | 34.28 | 34.69 | 34.355 | 34.64 | 34.305 | 34.235 | 6.1947e+... | 34.775 |
| 5 | 34.69 | 34.355 | 34.64 | 34.305 | 34.235 | 34.775 | 7.8223e+... | 33.935 |
| 6 | 34.355 | 34.64 | 34.305 | 34.235 | 34.775 | 33.935 | 1.0714e+... | 34.93 |
| 7 | 34.64 | 34.305 | 34.235 | 34.775 | 33.935 | 34.93 | 7.6828e+... | 33.05 |
| 8 | 34.305 | 34.235 | 34.775 | 33.935 | 34.93 | 33.05 | 8.0344e+... | 32.23 |
| 9 | 34.235 | 34.775 | 33.935 | 34.93 | 33.05 | 32.23 | 8.3882e+... | 31.87 |
| 10 | 34.775 | 33.935 | 34.93 | 33.05 | 32.23 | 31.87 | 4.898e+007 | 32.3 |
| 11 | 33.935 | 34.93 | 33.05 | 32.23 | 31.87 | 32.3 | 4.162e+007 | 31.9 |
| 12 | 34.93 | 33.05 | 32.23 | 31.87 | 32.3 | 31.9 | 5.8665e+... | 31.91 |
| 13 | 33.05 | 32.23 | 31.87 | 32.3 | 31.9 | 31.91 | 6.6096e+... | 31.16 |
| 14 | 32.23 | 31.87 | 32.3 | 31.9 | 31.91 | 31.16 | 6.9565e+... | 31.425 |
| 15 | 31.87 | 32.3 | 31.9 | 31.91 | 31.16 | 31.425 | 6.3461e+... | 31.855 |
| 16 | 32.3 | 31.9 | 31.91 | 31.16 | 31.425 | 31.855 | 7.0307e+... | 31.33 |
| 17 | 31.9 | 31.91 | 31.16 | 31.425 | 31.855 | 31.33 | 6.6769e+... | 30.56 |
| 18 | 31.91 | 31.16 | 31.425 | 31.855 | 31.33 | 30.56 | 6.5969e+... | 30.575 |
| 19 | 31.16 | 31.425 | 31.855 | 31.33 | 30.56 | 30.575 | 6.3314e+... | 30.2 |
| 20 | 31.425 | 31.855 | 31.33 | 30.56 | 30.575 | 30.2 | 6.128e+007 | 29.9 |

Fig. 7 Input Data Selection.

C. Model Training

After the number of rules is chosen, the next step is to train the input data based on the desired initialization settings. In model training, the system clustered the input data based on number of rules specified and then trained the data to obtain the basic model for data prediction purposes. The model was described by 3 basic types of

information produced at the output, which are centers, bases, and parameters value. The centers and bases describe the membership function of the antecedents or the points of each Centroid for each clusters of the input data, whereas parameters describe the consequent model. Using this model, the system would be able to make a prediction if it was given a set of testing data.

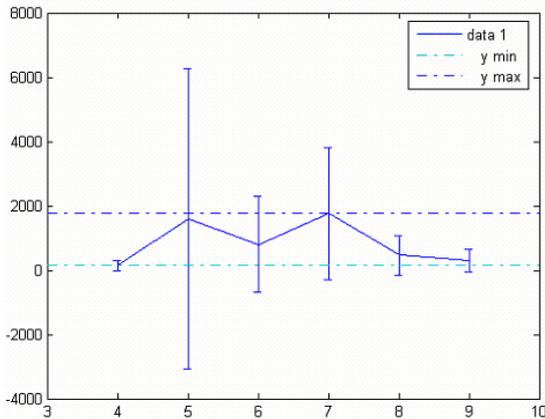


Fig. 8 Number of rules vs error value.

A. Model Prediction

The main input into this model is the test data. Test data only compiled of input data without any output information. The testing data was set on 1-year data of 2004. The testing data was setup in the same order as the input data, which contain 7 variables of input.

After the system finished calculating the test data, it generated a set of output data, which was the prediction of the stock's closing price. The final result of the entire system was the comparison chart between the outputs of test data against the actual stock's closing price data with the value of mean square error of 1.508.

DISCUSSION

Overall, the model was able to predict the movement of the price, even though there is some minor miscalculation. Fig. 9 shows the final output of the prediction with 3 highlighted area which shows significant miscalculation of the prediction process.

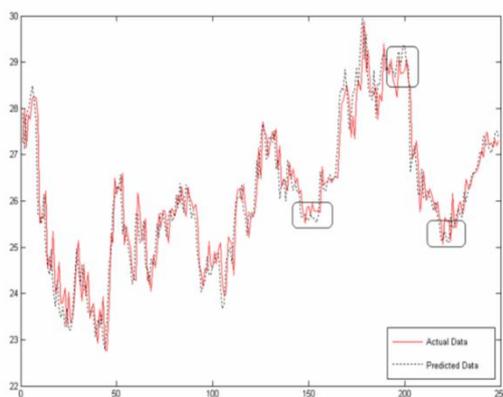


Fig. 9 Predicted output vs actual closing price.

This may occur due to the high dimension of uncertainty of the real world situation during the period that could not be eliminated by the fuzzy neural system.

CONCLUSION

The needs of accurate financial decision support system come from the strong demand of the financial society in overcoming potential market risks and the beneficial return in trading by indexes. Although the implementation of financial decision support system in trading industry has been significant, but there is no assurance of its stability in handling uncertainty in the market. For this work, Gaussian type-2 fuzzy set was identified as an efficient method in tackling uncertainty issue. It has also been discussed and identified the major importance of integrating the fuzzy system with neural network architecture. Other important issues that relate directly to the system's performance have also been brought up mainly on fuzzy clustering technique and structural parameters. Several experiments were conducted in order to clarify the issues by comparing the results with other existing models. Note that, the proposed architecture may not be the best technique to approach the financial decision support problem, but what this system wants to show is the possible steps and ways to deal with that particular issue: uncertainty issue can be approached using type-2 fuzzy set theory; and trend of the market can approached by using the simpler model such as neural network.

REFERENCES

- Ang, K. K. (2009). Rough set-based Neuro-Fuzzy System: Towards increasing interpretability without compromising accuracy. VDM Verlag.
- Bezdek, J. C. (1981). Models for pattern recognition. In Pattern recognition with fuzzy objective function algorithms (pp. 1-13). Springer US.
- Bolloju, N. (1996). Formulation of qualitative models using fuzzy logic. *Decision support systems*, 17(4), 275-298.
- Fama, E. F. (1963). Mandelbrot and the stable Paretian hypothesis. *Journal of Business*, 420-429.
- Hisdal, E. (1981). The IF THEN ELSE statement and interval-valued fuzzy sets of higher type. *International Journal of Man-Machine Studies*, 15(4), 385-455.
- Hoppner, F. (2015, July 17). Fuzzy clustering. Retrieved from <http://public.rz.fh-wolfenbuettel.de/~hoeppnef/clustering.html>.
- Jang, J. S. R. (1993). ANFIS: adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics, IEEE Transactions on*, 23(3), 665-685.
- Johnson, N. L., Kotz, S., & Balakrishnan, N. (1994). *Continuous univariate distributions*, vol. 1-2.
- Kaneko, T. (1996). Building a financial diagnosis system based on fuzzy logic production system. *Computers & industrial engineering*, 31(3), 743-746.
- Karnik, N. N., Liang, Q., & Mendel, J. M. (2001). Type-2 fuzzy logic software. Kasabov, N. (2001). Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 31(6), 902-918.
- Kasabov, N. K., & Song, Q. (2002). DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems, IEEE Transactions on*, 10(2), 144-154.
- Lendasse, A., de Bodt, E., Wertz, V., & Verleysen, M. (2000). Non-linear financial time series forecasting-Application to the Bel 20 stock market index. *European Journal of Economic and Social Systems*, 14(1), 81-91.

- Lippmann, R. P. (1987). An introduction to computing with neural nets. *ASSP Magazine, IEEE*, 4(2), 4-22.
- Mackey, M. C., & Glass, L. (1977). Oscillation and chaos in physiological control systems. *Science*, 197(4300), 287-289.
- Ng, G. S., Wahab, A., & Shi, D. (2003). Entropy learning and relevance criteria for neural network pruning. *International journal of neural systems*, 13(05), 291-305.
- Padmakumari, K., Mohandas, K. P., & Thiruvengadam, S. (1999). Long term distribution demand forecasting using neuro fuzzy computations. *International Journal of Electrical Power & Energy Systems*, 21(5), 315-322.
- Pham, D. L. (2001). Spatial models for fuzzy clustering. *Computer vision and image understanding*, 84(2), 285-297.
- Ritter, J. R. (2003). Behavioral finance. *Pacific-Basin Finance Journal*, 11(4), 429-437.
- Shaout, A., & Al-Shammari, M. (1998). Fuzzy logic modeling for performance appraisal systems: a framework for empirical evaluation. *Expert systems with Applications*, 14(3), 323-328.
- Sharaf, A. M., & Lie, T. T. (1995). A novel neuro-fuzzy based self-correcting online electric load forecasting model. *Electric power systems research*, 34(2), 121-125.
- Studer, L., & Masulli, F. (1997). Building a neuro-fuzzy system to efficiently forecast chaotic time series. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 389(1), 264-267.
- Sugeno, M., & Kang, G. T. (1988). Structure identification of fuzzy model. *Fuzzy sets and systems*, 28(1), 15-33.
- Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, (1), 116-132.
- Tan, W. W. (n.d.). A type-2 fuzzy logic controllers for liquid-level process. Retrieved from http://www.eng.nus.edu.sg/EResnews/0410/rd/rd_7.html.
- Tan, T. Z., Quek, C., & Ng, G. S. (2005). Brain-inspired genetic complementary learning for stock market prediction. In *Evolutionary Computation, 2005. The 2005 IEEE Congress on* (Vol. 3, pp. 2653-2660). IEEE.
- Tan, T. Z., Quek, C., & Ng, G. S. (2007). Biological brain inspired genetic complementary learning for stock market and bank failure prediction. *Computational intelligence*, 23(2), 236-261.
- Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning—I. *Information sciences*, 8(3), 199-249.