Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 135–144
Expert Systems with Applications
www.elsevier.com/locate/eswa

A TSK type fuzzy rule based system for stock price prediction

Pei-Chann Chang *, Chen-Hao Liu

Department of Information Management & Department of Industrial Engineering and Management, Yuan-Ze University, 135 Yuan-Tung Road, Chung-Li, 32026, Taiwan, ROC

Abstract

In this paper, a Takagi–Sugeno–Kang (TSK) type Fuzzy Rule Based System is developed for stock price prediction. The TSK fuzzy model applies the technical index as the input variables and the consequent part is a linear combination of the input variables. The fuzzy rule based model is tested on the Taiwan Electronic Shares from the Taiwan Stock Exchange (TSE). Through the intensive experimental tests, the model has successfully forecasted the price variation for stocks from different sectors with accuracy close to 97.6% in TSE index and 98.08% in MediaTek. The results are very encouraging and can be implemented in a real-time trading system for stock price prediction during the trading period.

Keywords: Fuzzy rule based systems; Forecasting; Stock market; Step regression analysis; Forecasting accuracy

1. Introduction

The price variation of stock market is a very dynamic system that has intrigued analysis from a number of disciplines. Two common analytical approaches are fundamental analysis and technical analysis. A fundamental analysis relies on the statistics of the macroeconomics data such as interest rates, money supply, inflationary rates, and foreign exchange rates as well as the basic financial status of the company. After taking all these factors into account, the analyst will then make a decision of selling or buying a stock. A technical analysis is based on the historical financial time-series data. However, financial time series exhibit quite complicated patterns (for example, trends, abrupt changes, and volatility clustering) and such series are often nonstationary, whereby a variable has no clear tendency to move to a fixed value or a linear trend.

Stock price prediction has always been a subject of interest for most investors and professional analysts. Nevertheless, finding out the best time to buy or to sell has remained a very difficult task because there are too many factors that may influence stock prices. Many papers have dealt with input selection when it comes to mapping financial indexes and stocks. Inputs have been divided into two different types of inputs, financial and political (which tend to be qualitative) and these inputs have been discussed in earlier researchers. Quah and Srinivasan (1999) identified 5 key factors that will affect the stock price movement namely yield, liquidity, risk, growth, and momentum. Izumi and Ueda (1999) mentioned that macroeconomic factors such as inflation and short-term interest rate have direct impacts on the stock returns.

As for the measure of system performance, Yao and Poh (1995) showed an example that a model with a low normalized mean square error (NMSE) had a lower return than a model with a higher NMSE. Brownstone (1996) recommended using percentages to measure performance so that the result could be better understood by traders and other people that might need their research and were not experts in the field. Chen, Leung, and Daouk (2003) used a sliding window to predict the next day's price of the index. Everyday the network was retrained with the most recent 68 days of input with the attempt to predict the coming day. Commission (remuneration for services rendered) is commonly overlooked when doing research relating to stock market prediction; however, if any model is actually implemented it is going to incur fees which could greatly affect the profit

* Corresponding author. Tel.: +886 3 4352654; fax: +886 3 4559378.
E-mail address: iepchang@saturn.yzu.edu.tw (P.-C. Chang).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2006.08.020

predicted by the model. Chen and Linkens (2004) considers 3 different levels of commissions and how it would affect the best buying strategy used by Investors.

Taiwan Stock Exchange (TSE) is the stock market in Taiwan whereby stocks may be bought and sold. Like every investment, investing in stocks entails some degree of risk. There are two types of risk; systematic risk and erroneous risk. The erroneous risk can be overcome by a sound investment strategy, called diversification. However by using a better prediction model to forecast the future price variation of a stock, the systematic risk can be minimized if it is not totally eliminated.

2. Literature survey

Prediction of stock price variation is a very difficult task and the price movement behaves more like a random walk and time varying. During the last decade, stocks and future traders have come to rely upon various types of intelligent systems to make trading decisions. Lately, artificial neural networks (ANNS) have been applied to this area (Aiken & Bsat, 1999; Chang, Wang, & Yang, 2004; Chi, Chen, & Cheng, 1999; Kimoto & Asakawa, 1990; Lee, 2001; Yao & Poh, 1995; Yoon & Swales, 1991). These models, however, have their limitations owing to the tremendous noise and complex dimensionality of stock price data and besides, the quantity of data itself and the input variables may also interfere with each other. Therefore, the result may not be that convincing.

Other soft computing methods are also applied in the prediction of stock price and these SC approaches are to use quantitative inputs, like technical indices, and qualitative factors, like political effects, to automate stock market forecasting and trend analysis. Kuo, Chen, and Hwang (2001) uses a genetic algorithm base fuzzy neural network to measure the qualitative effects on the stock price. Variable selection is critical to the success of any network for the financial viability of a company. They applied their system to the Taiwan stock market. Aiken and Bsat (1999) use a FFNN trained by a genetic algorithm (GA) to forecast three-month US Treasury Bill rates. They conclude that an NN can be used to accurately predict these rates. Thammano (1999) used a neuro-fuzzy model to predict future values of Thailand's largest government-owned bank. The inputs of the model were the closing prices for the current and prior three months, and the profitability ratios. The output of the model was the stock prices for the following three months. He concluded that the neuro-fuzzy architecture was able to recognize the general characteristics of the stock market faster and more accurately than the basic back propagation algorithm. Also, it could predict investment opportunities during the economic crisis when statistical approaches did not yield satisfactory results. Tansel et al. (1999) compared the ability of linear optimization, ANNs, and GAs to model time series data using the criteria of modeling accuracy, convenience and computational time. They found that linear optimization methods gave the best estimates, although the GAs could provide the same values if the boundaries of the parameters and the resolution were selected appropriately, but that the NNs resulted in the worst estimations. However, they noted that non-linearity could be accommodated by both the GAs and the NNs and that the latter required minimal theoretical background. Baba, Inoue, and Asakawa (2000) used NNs and GAs to construct an intelligent decision support system (DSS) for analyzing the Tokyo Stock Exchange Prices Indexes (TOPIX). The essential feature of their DSS was that it projected the high and low TOPIX values four weeks into the future and suggested buy and sell decisions based on the average projected value and the then-current value of the TOPIX. Kim and Han (2000) used a NN modified by a GA to predict the stock price index. In this instance, the GA was used to reduce the complexity of the feature space, by optimizing the thresholds for feature discretization, and to optimize the connection weights between layers. They concluded that the GA approach outperformed the conventional models. Abraham, Baikunth, and Mahanti (2001) investigated hybridized SC techniques for automated stock market forecasting and trend analysis. They used principal component analysis to preprocess the input data, a NN for one-day-ahead stock forecasting and a neuro-fuzzy system for analyzing the trend of the predicted stock values. Abraham, Philip, and Saratchandran (2003) investigate how the seemingly chaotic behavior of stock markets could be well represented using several connectionist paradigms and soft computing techniques. To demonstrate the proposed technique, they analyzed the 7 year's Nasdaq-100 main index and 4 year's NIFTY index values. They concluded that all the connectionist paradigms considered could represent the stock indices behavior very accurately.

Recently, neuron-fuzzy networks have been demonstrated to be successful applications in various areas such as in Chang, Liu, and Wang (in press), Chang and Warren Liao (2006), Chang and Wang (2006). Two typical types of neuron-fuzzy networks are Mamdani-type (Wang & Mendel, 1992) and TSK-type models (Takagi & Sugeno, 1985). For Mamdani-type neuron-fuzzy networks, the minimu, fuzzy implication is used in fuzzy reasoning. Meanwhile, for TSK-type neuron-fuzzy networks, the consequence of each rule is a function of various input variables. The general adopted function is a linear combination of input variables plus a constant term. Many researchers have shown that using a TSK-type neuron-fuzzy network achieves superior performance in network size and learning accuracy than that of Mamdani-type neuron-fuzzy networks. In this classic TSk-type neuron-fuzzy network, which is linear polynomial of the input variables, the system output is approximated locally by the rule hyper-planes.

3. A TSK type fuzzy rule based system

The working strategy of this research is to predict the future price of a stock by the technical index input to the

TSK fuzzy model. For a first order Takagi–Sugeno model, a common rule is represented as follows:

If x_1 is A_1, and x_2 is A_2, then $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ where x_1 and x_2 are linguistic variables and A_1 and A_2 are corresponding fuzzy sets and β_0, β_1, β_2 are linear parameters. Usually the least squares algorithm is used to determine the linear parameters and the membership function parameters are fine tuned using a neural network learning method. Initial rules are generated using the grid partitioning method. Usually a mixture of neural learning and global optimization method are used to fine tune the various rule parameters. The simulated annealing (SA) method will be adopted as a global optimization method to search for a set of near-optimal value of these rule parameters.

This research mainly studies the fluctuations of short-term stock prices and tries to develop a forecasting model using TSK type fuzzy rule based approach. The TSE index and MediaTek Inc. are selected for studying purposes. MediaTek Inc. is a professional fables IC company. Since its establishment in 1997, MediaTek has dedicated substantial resources in the research and development of comprehensive digital media integrated chipset solutions. MediaTek has now become one of the world's largest fabless IC companies.

3.1. Input variables

Technical indexes are calculated from the variation of stock price, trading volumes and time following a set of formula to reflect the current tendency of the stock price fluctuations. These indexes can be applied for decision making in evaluating the phenomena of oversold or overbought in the stock market. Basically, the technical index can be classified as index for TSE movement or particular stock price changes, such as KD, RSI, MACD, MA, BIAS. According to Chang et al. (2004), 8 technical indexes are described as shown in Table 3-1.

Table 3-1
Technical indexes used as input variables

Technical index	Explanation
Six days moving average (MA)	Moving averages are used to emphasize the direction of a trend and smooth out price and volume fluctuations that can confuse interpretation
Six days bias (BIAS)	The difference between the closing value and moving average line, which uses the stock price nature of returning back to average price to analyze the stock market
Six days relative strength index (RSI)	RSI compares the magnitude of recent gains to recent losses in an attempt to determine overbought and oversold conditions of an asset
Nine days stochastic line (K, D)	The stochastic line K and line D are used to determine the signals of over-purchasing, over-selling, or deviation
Moving average convergence and divergence (MACD)	MACD shows the difference between a fast and slow exponential moving average (EMA) of closing prices. Fast means a short-period average, and slow means a long period one
13 days psychological line (PSY)	PSY is the ratio of the number of rising periods over the total number of periods. It reflects the buying power in relation to the selling power
Volume	Volume is a basic yet very important element of market timing strategy; Volume provides clues as to the intensity of a given price move

3.2. Performance measure

The mean absolute percentage error (MAPE) is applied in this research as an evaluation basis. The formula is listed as follows:

$$\mathrm{MAPE} = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2, \tag{3.1}$$

where F: forecasted output; A: actual output; n: total number of records.

4. The development of a TSK Type fuzzy system

The research procedure will be introduced in this section and the overall framework is shown as Fig. 4-1.

The factor selection by stepwise regression and K-means clustering will be introduced in 4.1 and 4.2. Section 4.3 introduces the establishment of a simplified Fuzzy

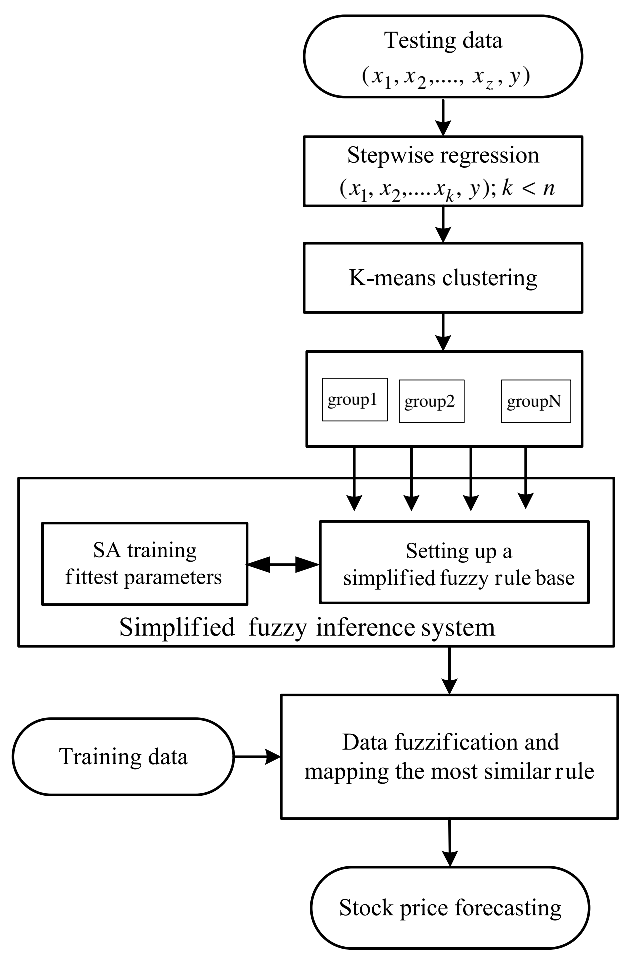

Fig. 4-1. Framework of the TSK fuzzy system.

Inference System, a new model proposed in this research which has two primary parts: the generation of the simplified fuzzy rule base and optimal parameters training of fuzzy rules. The detailed procedure is shown as Fig. 4-2.

4.1. Factor selection by stepwise regression

According to the numbers gathered and indices selected in previous section, 8 technique indices are proposed as the independent variables. To avoid the interferences caused in the training process by indices with low impacts themselves or losing the significance to the model under the influence of mutual function between two indices, which may decrease the explanation ability of such model and increase the error occurrence. Hence, the research adopts stepwise regression to analyze and select variables, and further simplifies it to improve the forecasting accuracy.

Stepwise regression is used to sort variables out and leave more influential ones in the model. In its operation a method by adding variables onward or removing variables backward is used to find the fittest combination for analyzing stock prices. The criterion for adding or removing is decided by F-test statistic value and decreasing the sum of squared error. After the entrance of first variable to the model, the variable number is increased step by step; once it is removed from this model, it will never enter the model again. Before selecting variables, the critical point, level of significant and the values of F_e(F-to-enter) F_r(F-to-remove) have to be determined first. Then the partial F value of each step has to be calculated and compared to F_e and F_r; If $F > F_e$, it is considered to add variables to the model; otherwise, if $F < F_r$, to remove variables from this model is considered. SPSS is used for variable selection and setting up the regression forecasting model.

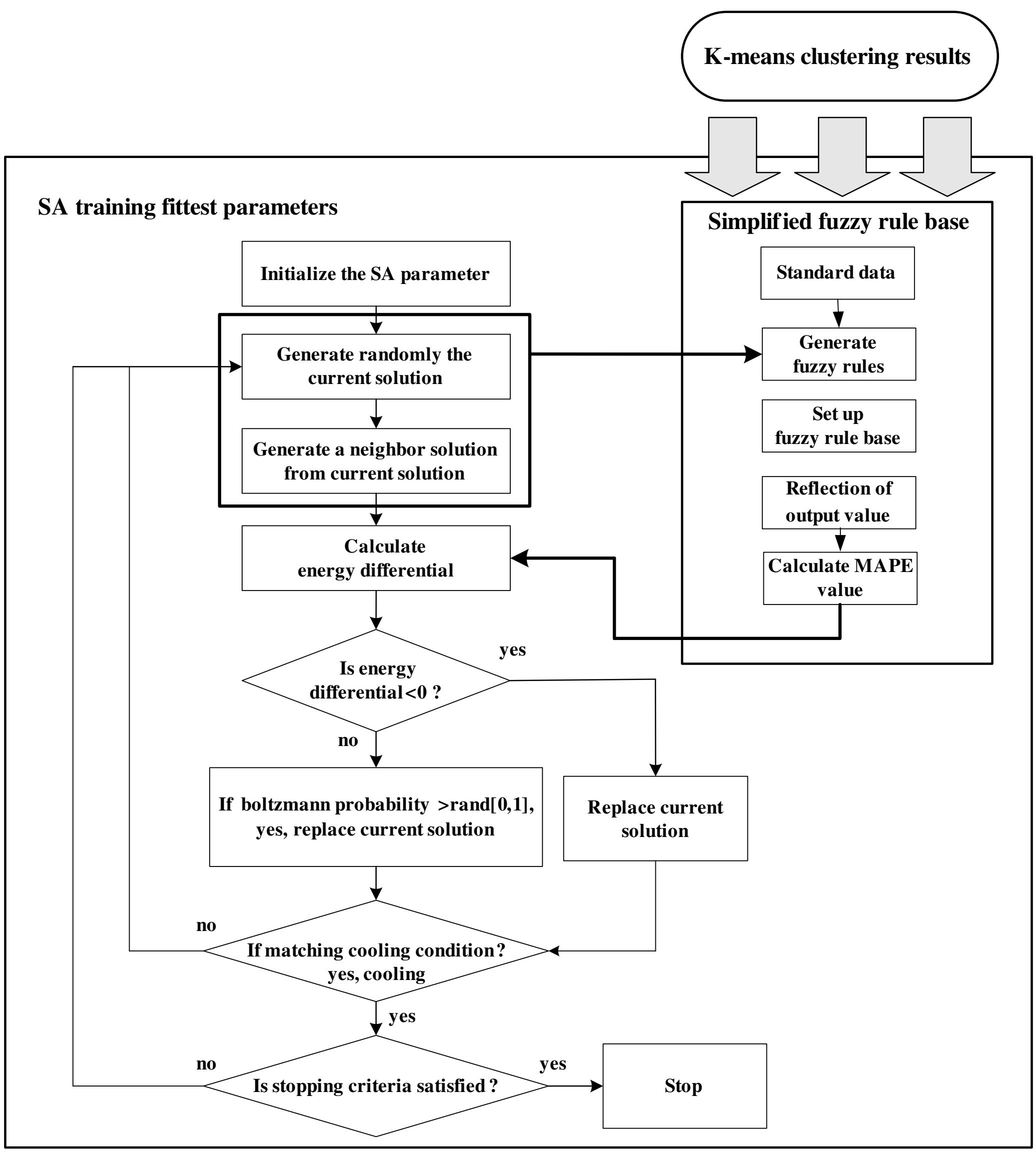

Fig. 4-2. Architecture of the simplified fuzzy inference system.

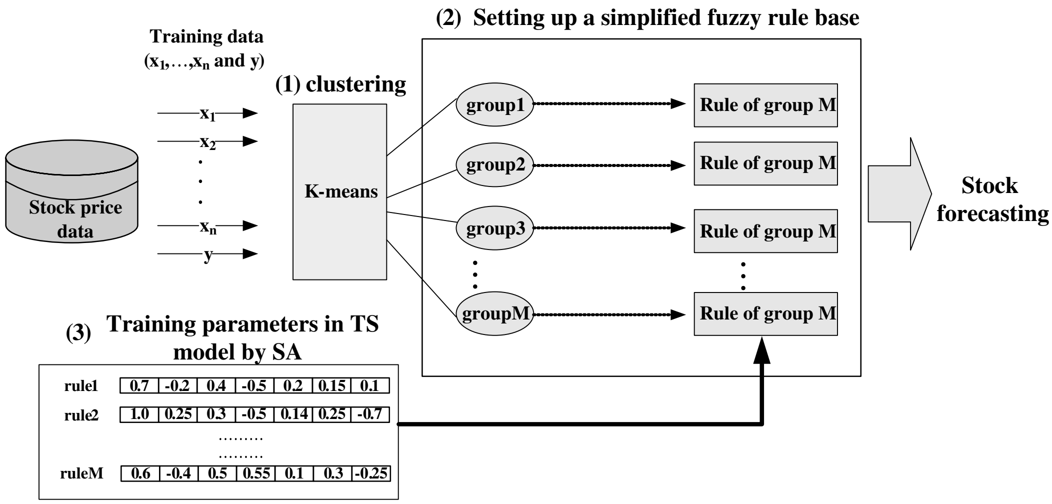

Fig. 4-3. The procedure framework of the simplified fuzzy inference system.

After selecting factors, *K*-means clustering technique and the simplified fuzzy rule base system are integrated for forecasting (as shown in Fig. 4-3). Detailed explanation of such setting process will be given in the following section.

4.2. *K*-means clustering analysis

K-means is a non-hierarchical clustering technique in which the dataset is partitioned into *K* clusters and the data points are randomly assigned to the clusters resulting in clusters that have roughly the same number of data points. The detailed procedures of *K*-means algorithm are described in Fig. 4-4.

K-means algorithm process

Input Value: K: clusters number and n: dataset amount

Output Value: K clusters dataset obtained by SE

Clustering process:

(1) Choose randomly from the dataset a point K as the initial data point.

(2) Calculate the distance from the data point to each cluster. If the data point is closest to its own cluster, leave it where it is. If the data point is not closest to its own cluster, move it into the closest cluster. Then recalculate the data point of each cluster.

(3) Repeat the above step until a complete pass through all the data points results in no data point moving from one cluster to another or until the termination condition is fulfilled. .

Fig. 4-4. The algorithm of *K*-means.

SE (squared-error) is defined as:

$$\mathrm{SE} = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2 \tag{4.1}$$

where p is the data point; C_i is ith cluster; m_i is the mean of all data points in C_i; k is the total clusters number, and its mean value is defined as:

$$m_i = \frac{\sum_{j=1}^{s_i} t_{ij}}{s_i} \tag{4.2}$$

where t_{ij} is the jth data point in cluster i; s_i is the total number of data points.

4.3. Setting up a simplified fuzzy rule inference system

The steps to set up the simplified fuzzy rule base and to train the fittest rule parameters set by simulated annealing will be introduced in this section.

4.3.1. Setting up a simplified fuzzy rule base

Suppose that x_1 and x_2 are input values and y are an output value. The input–output set is:

$$\left(x_1^{(1)}, x_2^{(1)}; y^{(1)}\right), \left(x_1^{(2)}, x_2^{(2)}; y^{(2)}\right), \ldots, \left(x_1^{(k)}, x_2^{(k)}; y^{(k)}\right), \ldots,$$

where the superscript (k) is the kth input–output set, $k = 1 \ldots N$, N is the total number of the input–output set. The fuzzy rules are used for a projecting/reflecting action: $f: (x_1, x_2) \to \hat{y}$

The simplified fuzzy rule base in TSK model can be set up by the following steps:

Training phase:

1. Data normalization

After analyzing the *K*-mean clustering, the mean value and standard deviation corresponding with the input and output variables of each cluster can be obtained and used for the data normalization. The formula is as follows:

(1) Input variable

$$x'_j = \frac{x_j - a_{ij}}{\sigma_{ij}}, \tag{4.3}$$

where a_{ij} and σ_{ij} are the mean value and standard deviation of the ith and jth input variables.

(2) Output variable (Bias)

$$y' = \frac{y - \bar{y}_i}{\sigma_i}, \tag{4.4}$$

where $\bar{y}_i$ and σ_i are the mean value and standard deviation of the input variables in ith cluster.

In the testing phase, the data, i.e., x, are first clustered and then used for normalization, i.e., x', by the mean value and standard deviation derived from the belonged group. The following Table 4-1 explains the normalization procedure of each clustered data:

2. Fuzzy rules generation

After data normalization, each input–output set $(x_1^{(k)}, x_2^{(k)}; y^{(k)})$ will be defined as a ultimate rule, no matter how many pieces of input–output set have been used. Thus for fuzzy if–then rules in this section, we adopt the TS fuzzy model, which is explained as follows:

Take ith as an example:

Rule i:

If x_1 *is* A_{i1}, *and* x_2 *is* A_{i2} , *Then* $1 = \beta_{i0} + 2.2\beta_{i1} + 0.6\beta_{i2}$

If x_1 *is* A_{i1}, *and* x_2 *is* A_{i2} , *Then* $1.75 = \beta_{i0} + 0.2\beta_{i1} - 0.2\beta_{i2}$

If x_1 *is* A_{i1}, *and* x_2 *is* A_{i2} , *Then* $0.5 = \beta_{i0} + 1.8\beta_{i1} - 1.0\beta_{i2}$

•

•

If x_1 *is* A_{i1}, *and* x_2 *is* A_{i2} , *Then* $2.25 = \beta_{i0} - 0.6\beta_{i1} + 0.2\beta_{i2}$

where A_{i1} and A_{i2} are fuzzy terms of x_1 and x_2.

In the process of data training, the main task is to find parameters β_{i0}, β_{i1} and β_{i2} in the fittest TS model by data inside the red square. After finishing training, the ith rule can be induced as:

Rule i: If x_1 is A_{i1}, and x_2 is A_{i2}, *then* $y\prime = \beta_{i0} + \beta_{i1}x'_1 + \beta_{i2}x'_2$.

3. Set up a fuzzy rule base

After finishing the training of all the parameters in each fuzzy rule, the rule base can be set up. Supposed that we have M cluster, then M rules will be generated:

Table 4-1
The normalization procedure of each clustered cata

	(μ,σ)	(450, 250)		(650, 250)		(30, 20)	
	Data	x_1	x'_1	x_2	x'_2	y	y'
ith cluster	1	1000	2.2	800	0.6	50	1.0
	2	500	0.2	600	−0.2	65	1.75
	3	900	1.8	400	−1.0	40	0.5
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	n_i	300	−0.6	700	0.2	75	2.25
	(μ,σ)	(200, 100)		(600, 200)		(20, 15)	
$i+1$th cluster	1	350	1.5	450	−0.75	15	−0.33
	2	100	−1	830	1.15	50	2.00
	3	430	2.3	570	−0.15	47	1.80
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	n_{i+1}	175	−0.25	320	−1.4	5	−1.00

Rule 1: If x_1 is A_{11}, and x_2 is A_{12}, then $y\prime = \beta_{10} + \beta_{11}x'_1 + \beta_{12}x'_2$

Rule 2: If x_1 is A_{21}, and x_2 is A_{22}, then $y\prime = \beta_{20} + \beta_{21}x'_1 + \beta_{22}x'_2$

Rule 3: If x_1 is A_{31}, and x_2 is A_{32}, then $y\prime = \beta_{30} + \beta_{31}x'_1 + \beta_{32}x'_2$

⋮

Rule M: If x_1 is A_{m1}, and x_2 is A_{m2}, then $y\prime = \beta_{m0} + \beta_{m1}x'_1 + \beta_{m2}x'_2$

4. Output values calculation

The purpose of this step is to predict the output values. When given a set of input values $(x_1, x_2, \ldots, x_m)$, the output value $\hat{y}$ can be predicted. The equation used is:

$$\hat{y} = \bar{y}_i + y'\sigma_i, \tag{4-5}$$

where $\bar{y}_i$ is the mean of the output values in ith cluster, and σ_i is the standard deviation of the output values in ith cluster.

4.3.2. Training best rule parameters by simulated annealing

This section explains how to apply the simulated annealing to cooperate with the operation model of the simplified fuzzy inference system. The purpose of applying the simulated annealing is to find a set of best values of the parameters within the fuzzy rule; that is, to find the optimal parameters combination for the calculation of the target value (i.e., mean absolutely percentage error, MAPE). The process is briefly introduced as follows: to find an initial solution first and randomly generate a neighborhood solution based on the initial one; if the generated solution is better than the current one, then it is accepted to replace the current one; otherwise, a probability is given to be determined whether it will be accepted or not. If not, the system will continue solution searching. When the number of solution accepted reaches the estimated number, the temperature has to be lowered and the optimal solution to the previous temperature is used as the initial solution for continued searching. This process will be repeated generation by generation until the termination condition is fulfilled. The randomly generated neighborhood solution is easier to be accepted under high temperature, which is similar to random searching. On the other hand, the randomly generated neighborhood solution is harder to be accepted under low temperature and this is similar to local searching.

The operation process of the simulated annealing approach is depicted as shown is Fig. 4-5:

The step-by-step procedure of the SA approach is explained in details in the following:

Step 1: SA parameter set-up, including the initial temperature, cooling coefficient, searching time of each temperature and the termination condition.

Step 2: A set of current solutions X is randomly generated, which is between -1 and 1; the predicted MAPE value of X can be obtained by the simplified fuzzy inference system.

Step 3: Randomly search for a neighbor solution set X', which equals to X augments and X is a random number between -0.05 and 0.05. All the neighbor solutions are substituted into the simplified fuzzy inference system to have the predicted MAPE value of X', and then the target distance (energy distance) of the current and neighbor solutions will be calculated as follows:

$$\Delta E = \mathrm{MAPE}(X') - \mathrm{MAPE}'(X) \tag{4.6}$$

If $\Delta E \leqslant 0$, then the current solution set will by replaced by the neighbor solution set; otherwise (when $\Delta E > 0$), the winning probability of the neighbor solution set is:

$$F(X') = \exp(-\Delta E/T). \tag{4.7}$$

A random number $u \in (0,1)$ will be afterwards generated and if $u < F(X')$, the neighbor solution set will replace the current solution set; otherwise, go back to step 3 to relocate the neighbor solution set, and add the searching time by 1.

Step 4: Compare X with the optimal solution set X''. If X is better, then it can be used to replace X''.

Step 5: If the maximum searching time of a certain temperature is achieved, then the temperature has to be cooled down, and the optimal solution of the last temperature will be set as the initial solution of the new temperature. If the maximum searching time is not achieved; go back to step 2.

Step 6: Check whether the termination condition is reached. If yes, finish the algorithm; otherwise, go back to carry out step 3 until the termination condition is fulfilled.

5. Experimental setup and results

The data set including TSE index and MediaTek Inc. has been decomposed into three different sets: the training

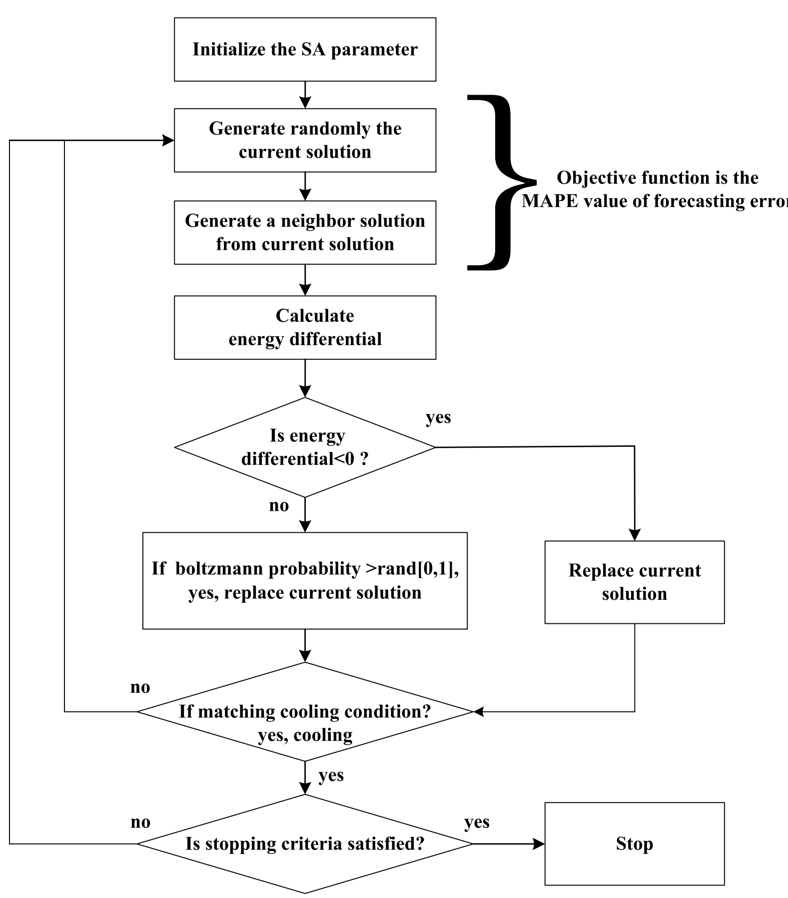

Fig. 4-5. The operation process of the simulated annealing approach.

data, test data and validation data. The data for TSE index are from July 18, 2003 to December 31, 2005 totally 614 records and the first 494 records will be training data and the rest of the data, i.e., 120 records will be test data. The data for MediaTek stock price are from March 3, 2002 to January 4, 2006 totally 710 records the first 590 records of data will be Training data and the rest of data, i.e., 140 records will be Test data. To avoid the interaction between the factors, we will test each factor using stepwise regression analysis and identify the factor that will affect the final forecasted results significantly. The final combination of the factors will be finalized after the analysis. The factors selected finally are MA6 and BIAS6 these two index and the output variables are TSE index and the stock price of MediaTek Inc.

5.1. Forecasted results in TSE index and MediaTek Inc.

The forecasted results of TSE index by different rules are listed in Table 5-1.

According to the MAPE value from different rules, Fig. 5-1 shows clearly that as the number of rules increases, the MAPE value decreases. The best result, is achieved when the number of rules is 9. However, when the number of of rules is over 9, the quality of the forecasted result decreases gradually.

The prediction results for MediaTek using different TSK rules are listed as follows (see Table 5-2).

According to the MAPE value from different rules, Fig. 5-2 shows clearly that as the number of rules increases, the MAPE value decreases. The best result is reached when

Table 5-1
Results of different number of rules applied in TSE index forecasting (%)

	Rule number						
	2	3	4	5	6	7	8
No. of Runs							
1	6.00	6.60	5.40	3.60	3.30	3.00	2.10
2	6.30	7.20	5.10	3.60	2.70	2.70	2.70
3	7.20	6.00	5.10	3.30	3.30	2.40	2.70
4	6.30	6.00	4.50	3.30	2.70	2.70	2.40
5	6.00	6.30	6.00	3.60	2.70	2.40	2.70
CPU time(s)	4.6	4.775	4.912	4.987	5.328	5.356	5.353
Average	6.30	6.30	5.10	3.60	3.00	2.70	2.70
Standard error	0.60	0.60	0.60	0.30	0.30	0.30	0.30

	9	10	11	12	13	14	15
No. of Runs							
1	2.10	2.40	4.20	3.60	3.60	4.80	5.10
2	2.70	2.70	4.20	3.90	3.30	3.00	4.80
3	2.40	2.70	2.70	3.00	4.50	3.30	4.80
4	2.40	2.40	3.30	3.00	2.70	2.10	3.90
5	2.10	2.70	4.20	3.30	3.60	3.90	4.80
CPU time(s)	5.315	5.6	5.965	6.084	6.14	6.393	6.487
Average	2.40	2.40	3.90	3.30	3.60	3.30	4.50
Standard error	0.30	0.30	0.60	0.30	0.60	0.90	0.30

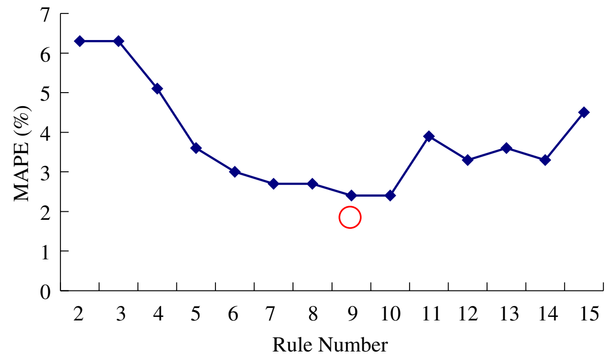

Fig. 5-1. MAPE values of the prediction results for different rules in TSE index.

Table 5-2
Prediction results of different rules for MediaTek (%)

	Rule number						
	2	3	4	5	6	7	8
No. of Runs							
1	8.70	4.14	4.08	2.94	2.91	2.37	4.80
2	11.97	4.32	4.05	2.97	2.76	1.71	2.28
3	11.58	3.93	4.02	2.97	4.08	1.86	6.78
4	7.65	4.17	4.11	4.17	2.67	1.83	4.44
5	7.65	3.96	4.08	2.73	2.73	1.86	1.80
CPU time(s)	5.275	5.55	5.731	5.84	6.006	6.246	6.465
Average	9.51	4.11	4.08	3.15	3.03	1.92	4.02
Standard error	2.13	0.15	0.03	0.57	0.60	0.27	2.04

	9	10	11	12	13	14	15
No. of Runs							
1	3.66	4.08	5.34	4.32	5.94	6.33	4.83
2	1.86	6.42	3.12	3.93	3.81	7.47	6.36
3	3.48	1.59	5.16	3.24	3.87	7.35	12.78
4	2.79	1.74	5.85	7.29	4.50	6.90	6.84
5	2.70	4.68	3.39	3.33	6.33	6.78	5.58
CPU time(s)	6.581	6.871	6.905	7.299	7.394	7.628	7.882
Average	2.91	3.69	4.56	4.41	4.89	6.96	7.29
Standard error	0.72	2.04	1.23	1.68	1.17	0.48	3.18

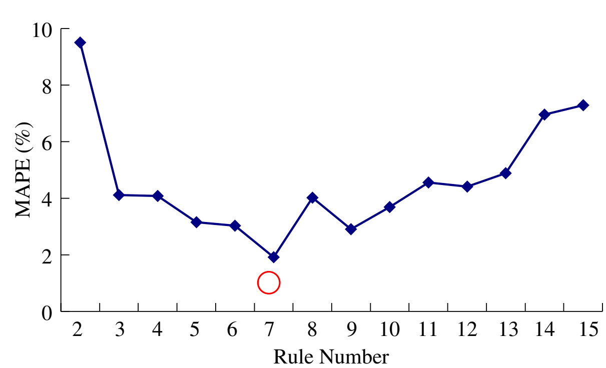

Fig. 5-2. MAPE values of the prediction results for different rules in MediaTek.

the number of rules is 7. However, when the number of rules is over 7, the quality of the forecasted result decreases gradually.

Table 5-3
Comparison of different methods

	Method		
	TSK fuzzy system	BPN	Multiple regression analysis
Data			
MAPE of Taiwan stock exchange index (TSE)	2.4% (rule number is 9)	4.29%	7.08%
MAPE of MediaTek	1.92% (rule number is 7)	3.43%	6.62%

5.2. Comparisons of different forecasting models

The comparisons of different models such as BPN, multiple regression and the TSK fuzzy rule model are listed in Table 5-3. As we can observe here, no matter if it is TSE index or a particular stock such as MediaTek Inc. the forecasted results from TSK fuzzy rule model are much better than those from BPN or multiple regression which justify the TSK fuzzy rule model is the best.

6. Conclusions

A TSK fuzzy based system is presented in the paper by applying a linear combination of the significant technical index as a consequent to predict the stock price. Input variables are effectively selected through the stepwise regression from the set of technical index. In addition, SA is applied to further fine-tune the parameters of the linear combination of the input variables; therefore the forecasting capability of the system is greatly improved. Finally, the system is tested on the TSE and MediaTek Inc. and all the performance results are outperforming other approaches such as BPN, or multiple regression analysis.

The number of rule clusters of the TSK rule based system is subjectively set from 2 to 15 and the empirical results on the two data sets showing that when rule number is 7 has the best forecasting results. Detailed experimental design can be setup for decision of the set of parameters such as number of rules, and different input of the technical index. This research is just a beginning and the long term goal is to predict the trend of the price variation by including various influential factors such as macro economic change, political reasons, fundamental analysis and the technical index..., etc. As a result, the system can be further applied for the daily trading purpose.

References

Abraham, A., Baikunth, N., & Mahanti, P. K. (2001). Hybrid intelligent systems for stock market analysis. *Lecture Notes in Computer Science, 2074*, 337–345.

Abraham, A., Philip, N. S., & Saratchandran, P. (2003). Modeling chaotic behavior of stock indices using intelligent paradigms. *Neural, Parallel and Scientific Computations, 11*, 143–160.

Aiken, M., & Bsat, M. (1999). Forecasting market trends with neural networks. *Information Systems Management, 16*(4), 42–48.

Baba, N., Inoue, N., & Asakawa, H. (2000). Utilization of Neural Networks and GAs for Constructing Reliable Decision Support Systems to Deal Stocks. *IEEE–INNS–ENNS International Joint Conference on Neural Networks (IJCNN'00)* 5, 5111–5116.

Brownstone, D. (1996). Using percentage accuracy to measure neural network predictions in stock market movements. *Neurocomputing, 10*, 237–250.

Chang, P.C., Liu, C.H., & Wang, Y.W. (in press). A hybrid model by clustering and evolving fuzzy rules for sale forecasting in printed circuit board industry. *Decision Support Systems*. On-line available, December 20.

Chang, P. C., & Wang, Y. W. (2006). Fuzzy delphi and back-propagation model for sales forecasting in PCB Industry. *Expert Systems with Applications, 30*(4), 715–726.

Chang, P. C., Wang, Y. W., & Yang, W. N. (2004). An investigation of the hybrid forecasting models for stock price variation in Taiwan. *Journal of the Chinese Institute of Industrial Engineering, 21*(4), 358–368.

Chang, P. C., & Warren Liao, T. (2006). Combing SOM and fuzzy rule base for flow time prediction in semiconductor manufacturing factory. *Applied Soft Computing, 6*(2), 198–206.

Chen, A. S., Leung, M. T., & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan stock index. *Computers and Operations Research, 30*, 901–923.

Chen, M. Y., & Linkens, D. A. (2004). Rule-base self-generation and simplification for data-driven fuzzy models. *Fuzzy Sets and Systems, 142*, 243–265.

Chi, S. C., Chen, H. P., & Cheng, C. H. (1999). A forecasting approach for stock index future using Grey theory and neural networks. *IEEE International Joint Conference on Neural Networks*, 3850–3855.

Izumi, K., & Ueda, K. (1999). Analysis of exchange rate scenarios using an artificial market approach. *Proceedings of the International Conference on Artificial Intelligence, 2*, 360–366.

Kim, K. J., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications, 19*, 125–132.

Kimoto, T., & Asakawa, K. (1990). Stock market prediction system with modular neural network. *IEEE International Joint Conference on Neural Network*, 1–6.

Kuo, R. J., Chen, C. H., & Hwang, Y. C. (2001). An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy Sets and Systems, 118*, 21–24.

Lee, J.W. (2001). Stock Price Prediction Using Reinforcement Learning. *IEEE International Joint Conference on Neural Networks*, 690–695.

Quah, T. S., & Srinivasan, B. (1999). Improving returns on stock investment through neural network selection. *Expert Systems with Applications, 17*, 295–301.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics, 15*, 16–132.

Tansel, I.N., Yang, S.Y., Venkataraman, g., Sasirathsiri, A., Bao, W.Y., & Mahendrakar, N. (1999). Modeling Time Series Data by Using Neural Netwroks and Genetic Algorithms. In Dagli, C.H., A.L. Buczak, J. Ghosh, M.J. Embrechts, & O. Ersoy (Eds.) Smart Engineering System Design: neural networks, fuzzy logic, evolutionary programming, data mining, and complex systems. In *Proceedings of the Artificial Neural Networks in Engineering Conference (ANNIE '99)*. New York: ASME Press, 1055–1060.

Thammano, A. (1999). Neuro-fuzzy Model for Stock Market Prediction. In Dagli, C.H., A.L. Buczak, J. Ghosh, M.J. Embrechts, & O. Ersoy (Eds.) Smart Engineering System Design: neural networks, fuzzy logic, evolutionary programming, data mining, and complex systems. In *Proceedings of the Artificial Neural Networks in Engineering Conference (ANNIE '99)*. New York: ASME Press, 587–591.

Wang, L. X., & Mendel, J. M. (1992). Generating fuzzy rules by learning form examples. *IEEE Transaction on Systems, Man, and Cybernetics, 22*, 1414–1427.

Yao, J., & Poh, H. L. (1995). Forecasting the KLSE index using neural networks. *IEEE International Conference on Neural Networks., 2*, 1012–1017.

Yoon, Y., & Swales, J. (1991). Prediction stock price performance: a neural network approach. *Proceeding of Twenty-Fourth Annual Hawaii International Conference on System Science*, 156–162.