A Hybrid System Integrating a Wavelet and TSK Fuzzy Rules for Stock Price Forecasting

Pei-Chann Chang and Chin-Yuan Fan

Abstract—The prediction of future time series values based on past and present information is very useful and necessary for various industrial and financial applications. In this study, a novel approach that integrates the wavelet and Takagi–Sugeno–Kang (TSK)-fuzzy-rule-based systems for stock price prediction is developed. A wavelet transform using the Haar wavelet will be applied to decompose the time series in the Haar basis. From the hierarchical scalewise decomposition provided by the wavelet transform, we will next select a number of interesting representations of the time series for further analysis. Then, the TSK fuzzy-rule-based system is employed to predict the stock price based on a set of selected technical indices. To avoid rule explosion, the k-means algorithm is applied to cluster the data and a fuzzy rule is generated in each cluster. Finally, a K nearest neighbor (KNN) is applied as a sliding window to further fine-tune the forecasted result from the TSK model. The simulation results show that the model has successfully forecasted the price variation for stocks with accuracy up to 99.1% in Taiwan Stock Exchange index. Comparative studies with existing prediction models indicate that the proposed model is very promising and can be implemented in a real-time trading system for stock price prediction.

Index Terms—Fuzzy ruled system, K-mean clustering, multiple regression analysis (MRA), simulated annealing (SA), wavelet preprocessing.

I. INTRODUCTION

MINING stock market tendencies is a challenging task due to its high volatility and noisy environment. Many factors influence the performance of a stock market including political events, general economic conditions, and traders’ expectations. Though stocks and futures traders have relied heavily upon various types of intelligent systems to make trading decisions, the success so far is quite limited [4].

Many attempts have been made to predict the financial markets, ranging from traditional time series approaches to artificial intelligence techniques such as fuzzy systems, and especially, artificial neural network (ANN) methodologies [1]. However, the main drawback with ANNs, and other black-box techniques, is the tremendous difficulty in interpreting the results. They do not provide an insight into the nature of the interactions between the technical indicators and the stock market fluctuations. Thus, there is a need to develop methodologies that facilitate an increased understanding of market processes, in addition to providing temporally accurate predictions [17], [35], [65]. Another issue to be dealt with is that the dimensionality of financial time series data also creates another challenge in ANN approaches.

The development of a timely and accurate trading decision-making tool is a key for stock traders to make profits. Since the stock price series is affected by a mixture of deterministic and random factors [17], new tools and techniques are needed in dealing with noise and nonlinearity in stock price prediction. Data mining, aimed at finding rules hidden in very large amount of data, is a new and efficient approach for time series analysis. Data mining on time series needs to translate the continuous time series into discrete symbol sequences first. In this work, the wavelet transform using the Haar wavelet will be applied to decompose the time series in the Haar basis. From the hierarchical scalewise decomposition provided by the wavelet transform, we will next select a number of interesting representations of the time series for further analysis. In addition, statistical analysis is employed to select important factors that affect the performance of the stock market the most. These factors are chosen as the inputs of the Takagi–Sugeno–Kang (TSK) fuzzy-rule-based system to predict the future stock price. The reason for choosing the TSK fuzzy system is owing to its universal approximation capability [54] and the possibility to gain insights into the data, which is of particular interest for stock price prediction.

The proposed framework combines several soft computing (SC) techniques such as a wavelet transform, TSK fuzzy system, data clustering, simulated annealing (SA), and K nearest neighbor (KNN). In addition to wavelet-based data representation, the k-means clustering algorithm is applied to cluster the data before the TSK fuzzy rules are generated. A fuzzy rule is then generated for each cluster, which enables us to determine the membership functions of the fuzzy subsets, and the optimal number of fuzzy rules as well. Finally, a KNN is applied as a sliding window to further fine tune the forecasted result from the TSK model.

The remainder of the paper is organized as follows. Section II reviews the different methods for stock forecasting using SC techniques such as neural networks (NNs) and fuzzy systems. Section III describes the proposed hybrid approach to stock price prediction by integrating a wavelet with the TSK fuzzy-rule-based system. Section IV presents empirical results of the hybrid approach and compare it with three other approaches. Finally, conclusions and future directions of the research are discussed in Section V.
II. LITERATURE SURVEY

Conventional research addressing the stock forecasting problem has generally relied on time series analysis techniques, i.e., mixed autoregression moving average (ARMA) as well as multiple regression models (MRMs). However, the assumptions of these methods may come out with ineffective results since a number of missing factors such as macroeconomic or political effects may seriously influence stock tendencies.

White [59] was the first to use NNs for market forecasting. He used a feedforward NN (FFNN) to study the IBM daily common stock returns and he found that his training results were overoptimistic, being the result of overfitting or of learning irrelevant features. In general, there are two different methodologies for stock price prediction in using ANN as a research tool [68]. The first methodology is to consider the stock price variations as a time series and predict the future price based on its past values. In this approach, ANNs have been employed as the predictor, see, e.g., [5], [10], [14], [16], [17], [27], [32], [37], [42], [48], [63], and [64]. These prediction models, however, have their limitations owing to the tremendous noise and high dimensionality of stock price data. Therefore, the performances of the existing models are not satisfactory [65].

The second approach takes the technical indices and qualitative factors such as political effects into account in stock market forecasting and trend analysis.

Yao and Poh [63] use technical indicators (%K and %D) along with price information to predict future price values. They achieved good returns, and found that their models performed better using daily data rather than weekly data. Hobbs and Bourbakis [26] predict prices of stocks based on the fluctuations in the rest of the market for the same day. They show consistently high rates of return, although the investment is done in a frictionless environment. Paying commissions on the large number of trades instigated would certainly erode much of the benefit from the trading strategy proposed. Austin and Looney [8] develop an NN that predicts the proper time to move money into and out of the stock market. They used two valuation indicators, two monetary policy indicators, and four technical indicators to predict the four week forward excess return on the dividend adjusted S&P 500 stock index. The results significantly outperformed the buy-and-hold strategy. Backpropagation ANNs are applied to predict future elements in the price time series in the Korea composite stock price index (KOSPI) [30]. López et al. [38] use time delay connections in enhanced NNs (that is, the addition of time-dependant information in each weight) to forecast IBEX-35 (Spanish stock index) index close prices one day ahead. Stochastic NNs is applied for forecast the volatility of index returns in the TUNINDEX (Tunisian stock index), and finds that the out-of-sample NN results are superior to traditional generalized autoregressive conditional heteroskedastic (GARCH) models [52]. Nenortaitė and Simutis [41] present a trading approach based on one-step ahead profit estimates created by combining NNs with particle swarm optimization algorithms. The method is profitable given small commission costs, but does not exceed the S&P500 returns when realistic commissions are introduced. Jaruszewicz and Mandziuk [29] train ANNs using both technical analysis variables and intermarket data, to predict one day changes in the NIKKEI index. They achieve good results using moving average convergence divergence (MACD), Williams, and two averages, along with related market data from the National Association of Securities Dealers Automated Quotation System (NASDAQ) and DAX.

It has been a new tendency that combining the SC technologies of NNs, fuzzy logic (FL), and genetic algorithms (GAs) may significantly improve an analysis [1], [2], [9], [11]–[13], [19], [24], [25], [27], [31], [36], [39], [40], [51], [53], [55]–[57], [61]. In generally, NNs are used for learning and curve fitting, FL is used to deal with imprecision and uncertainty, and GAs are used for search and optimization [13], [28], [39], [62]. Zadeh [65] pointed out that merging these technologies allows for the exploitation of a tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, and low solution cost.

Wavelet analysis is a relatively new field in signal processing [18]. Wavelets are mathematical functions that decompose data into different frequency components, and then, study each component with a resolution matched to its scale—a scale refers to a time horizon [47]. Wavelet filtering is particularly relevant to volatile and time-varying characteristics of real-world time series and is not restrained by the assumption of stationarity [44]. The wavelet transform decomposes a process into different scales, which makes it useful in differentiating seasonalties, revealing structural breaks and volatility clusters, and identifying local and global dynamic properties of a process at these timescales [5], [23], [45]. Wavelet analysis has been shown to be especially productive in analyzing, modeling, and predicting the behavior of financial instruments as diverse as stocks and exchange rates [43], [46].

In this paper, a wavelet filtering is used and that is because of the property of wavelets for economic analysis in data decomposition by time scale. Economic and financial systems, like many other systems, contain variables that operate on a variety of time scales simultaneously so that the relationships between variables may well differ across time scales. A wavelet will decompose the time series into a range of frequency scales [7], [19], [22]. The lower level of the decomposition can capture the long range dependencies with only a few coefficients, while the higher levels capture the usual short-term dependencies. This research, motivated by the effective preprocessing capability of wavelets and the predictive power of fuzzy rule system, presents a hybrid system by integrating the wavelet and a TSK fuzzy rule system for stock price prediction.

III. METHODOLOGY

In tradition, there are two major factors to be considered in stock forecasting, and they are technical analysis and fundamental analysis. Technical analysis concentrates on the study of market action, and fundamental analysis concentrates on the economic forces of supply and demand that cause price movements. In addition, as explained in [68], statistics, technical analysis, fundamental analysis, and linear regression are all used to attempt to predict the market’s direction. However, technical
indexes themselves alone often miss a lot of potential chances in the stock movement before the appropriate trading signal is generated. Thus, although technical analysis may yield insights into the market, its highly subjective nature and inherent time delay does not make it ideal for the fast, dynamic trading markets. That is why a wavelet and TSK fuzzy-rule-based model is developed in this research.

Fundamental analysis must rely on the reasons of price movement, and this process is very complicated since there are so many factors that may affect the price change such as political, psychotically events, etc. Therefore, the basic assumption of this research is that the price movement is closely related to the variation of technical index as widely applied in financial time series researches. A set of technical indexes will be applied as input factors and the output will be the stock price. To study their relationship, a hybrid method that integrates a wavelet and Takagi and Sugeno fuzzy-system-based forecasting model is developed and implemented in this research for Taiwan stock price prediction. The main procedures of the hybrid system are shown in Fig. 1 and inputs and outputs of each block are further explained in the following sections.

The notation of variables applied in the following sections is shown in Table I.

A. Data Preprocessing Using Wavelet Theory

The reason for applying wavelet theory as a data preprocessing method is because that as mentioned by Ramsey [46], the process of representation in wavelet is able to deal with the non-stationarity involved with economic and financial time series. One of the benefits of a wavelet approach is the flexibility in handling very irregular data series, as illustrated in [47]. Economic and financial systems contain variables that operate on a variety of time scale simultaneously so that the relationship between variables may differ across time scale. The most important property of wavelets for economic analysis is decomposition by time scale.

In this research, Haar wavelet is applied as our major wavelet transform tools. Haar wavelet is a wavelet evolved from continuous wavelet transform. According to [3], wavelet not only decompose the data in terms of times and frequency, but also can reduce lots of processing times. For a time series of size \( N \), the wavelet decomposition used here can be determined in \( O(n) \) time. In considering Haar wavelet and Coiflets wavelet [18], Coiflets wavelet considers more aspects than Harr wavelet, especially in combining compact support with various degree of smoothness and numbers of vanishing moments [3], but Haar wavelet still provides easily and quickly process time without losing much in performance than other wavelet systems [48]. Haar wavelet has been widely applied in time series forecasting [6], [48].

Depending on normalization rules, there are two types of Haar wavelets within a given function/family: father and mother wavelets

\[
\begin{align*}
\Phi_{a,b} &= 2^{-1/2} \Phi \left( \frac{t - 2^n b}{2^a} \right) \\
\psi_{a,b} &= 2^{-1/2} \psi \left( \frac{t - 2^n b}{2^a} \right)
\end{align*}
\]

Father wavelets are used for the “lowest frequency” smooth components; those requiring wavelets with the widest support and mother wavelets are used for the “higher frequency” detail components. Father wavelets are used for the “trend components” and mother wavelets are used for all the deviations from trend. While a sequence of mother wavelets is used to represent a function, only one father wavelet is used.

A time series data, i.e., function \( f(t) \), is an input to be represented by a wavelet analysis, and it can be built up as a sequence of projections onto father and mother wavelets indexed by both \( \{b\}, b = \{0, 1, 2, \ldots\} \) and by \( \{s\} = 2^a, \{a = 1, 2, 3, \ldots\} \). In actual data analysis using discretely sampled data, it is necessary to create a lattice over which the calculations will be made.
TABLE I
NOTATION OF VARIABLES APPLIED IN SECTION III

<table>
<thead>
<tr>
<th>Parameter</th>
<th>description</th>
<th>Parameter</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\psi$</td>
<td>Single mother wavelet</td>
<td>$F'_{j}$</td>
<td>Maximum partial $F$ value</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Single father wavelet</td>
<td>$A_\varphi$</td>
<td>$A_\varphi = 1, \ldots, N$ are fuzzy subsets, $N$ is the number of fuzzy rules</td>
</tr>
<tr>
<td>$a$</td>
<td>represents the parameter of observation scale</td>
<td>$\beta_i$</td>
<td>$\beta_i$ parameters for consequence rules</td>
</tr>
<tr>
<td>$A$</td>
<td>represents the maximum observation scale</td>
<td>$w_i$</td>
<td>$w_i$ is the strength of rule $i$</td>
</tr>
<tr>
<td>$b$</td>
<td>represents the parameter of translation scale</td>
<td>$p$</td>
<td>$p$ are data points in the cluster</td>
</tr>
<tr>
<td>$s$</td>
<td>Integral of Single father wavelet</td>
<td>$m_i$</td>
<td>$m_i$ is the center of cluster</td>
</tr>
<tr>
<td>$d$</td>
<td>Integral of Single mother wavelet</td>
<td>$SE$</td>
<td>squared error $\sum_{i=1}^{n} \sum_{j \in C_i}</td>
</tr>
<tr>
<td>$f(t)$</td>
<td>A financial time series data</td>
<td>$a_i$</td>
<td>$a_i$ is the membership functions for each cluster</td>
</tr>
<tr>
<td>$S_i$</td>
<td>an approximation signal</td>
<td>$\sigma_i$</td>
<td>$\sigma_i$ is the standard of membership functions for each cluster</td>
</tr>
<tr>
<td>$D_i$</td>
<td>a detail signal</td>
<td>$K$</td>
<td>$K$ is the total number of clusters</td>
</tr>
<tr>
<td>$\hat{y}$</td>
<td>linear regression model $\hat{y} = f(x)$</td>
<td>$S_i$</td>
<td>the number of data points in cluster $i$</td>
</tr>
<tr>
<td>$X_i$</td>
<td>Input data, i.e., important technical index</td>
<td>$t$</td>
<td>the number of data in now</td>
</tr>
<tr>
<td>$SSE$</td>
<td>A Sum of Squares $SSE = \sum_{i=1}^{n} \left( Y_i - \hat{Y}_i \right)^2$</td>
<td>$L$</td>
<td>Sliding window size</td>
</tr>
<tr>
<td>$SSR$</td>
<td>A Sum of Squared Deviation $SSR = \sum_{i=1}^{n} \left( \hat{Y}_i - \bar{Y} \right)^2$</td>
<td>$j$</td>
<td>$1 \ldots t - s$ period data</td>
</tr>
</tbody>
</table>

Mathematically, it is convenient to use a dyadic expansion as illustrated in (1).

The coefficients in the expansion are given by the projections:

$$
s_{A,b} = \int f(t)\Phi_{A,b}(t)dt
$$

$$
d_{a,b} = \int f(t)\psi_{a,b}(t)dt,
$$
and

$$
a = 1, 2, \ldots, A
$$

(2)

where $A$ is the maximum scale sustainable by the number of data points. The representation of the signal $f(t)$ can now be given by:

$$
f(t) = \sum_{b} s_{A,b}\Phi_{A,b}(t) + \sum_{b} d_{a,b}\psi_{A,b}(t) + \sum_{b} d_{A-1,b}\psi_{A-1,b}(t) + \cdots + \sum_{b} d_{1,b}\psi_{1,b}(t).
$$

(3)

The approximation can be represented:

$$
f(t) = S_A(t) + D_A(t) + D_{A-1}(t) + \cdots + D_1
$$

$$
S_A(t) = \sum_{b} S_{A,b}\phi_{A,b}(t)
$$

$$
D_A(t) = \sum_{b} D_{A,b}\psi_{A,b}(t).
$$

(4)

When $n$, the number of observations is divisible by $2^j$, and then, the number of coefficients of each type is given by the following:

1) at the finest scale, $2^j$: $n/2^j$ coefficients $d_{1,b}$;
2) at the next scale, $2^{j+1}$: $n/2^{j+1}$ coefficients $d_{2,b}$;
3) at the coarsest scale, $2^{j+1}$: $n/2^{j+1}$ coefficients $d_{A,b}$; and
4) at the coarsest scale, $2^{j+1}$: $n/2^{j+1}$ coefficients $s_{A,b}$.
represents an approximation signal, and

Fig. 2. Wavelet transforming process from $f(t)$.

so that

$$ n = \frac{n}{2} + \frac{n}{4} + \cdots + \frac{n}{2A-1} + \frac{n}{2A}. $$

(5)

As shown in Fig. 2, $f(t)$ represents the original data, $S_1$ represents an approximation signal, and $D_1$ represents a detailed signal. We can define the multiresolution decomposition of a signal by specifying:

SA coarsest scale, and

$$ S_{A-1} = S_A + D_A. $$

(6)

In general

$$ S_{n-1} = S_n + D_n $$

(7)

where $\{S_A, S_{A-1}, \ldots, S_1\}$ is a sequence of multiresolution approximations of the function $f(t)$, at ever increasing levels of refinement. The corresponding multiresolution decomposition of $f(t)$ is given by $\{S_A, D_A, D_{A-1}, \ldots, D_2, \ldots, D_1\}$.

The sequence of terms $S_A, D_A, D_{A-1}, \ldots, D_2, \ldots, D_1$ represents a set of orthogonal signal components that provide representations of the signal at resolutions 1 to $A$; each $DA-k$ provides the orthogonal increment to the representation of the function $f(t)$ at the scale, or resolution $2^{A-k}$. When the data pattern is very rough, the wavelet process will be repeatedly applied. In the preprocessing, the target is to minimize the mean absolute percentage error (MAPE) between the signal before and after transformation. In this way, the noise in the original data can be removed.

B. TSK-Fuzzy-System-Based Prediction

1) Input Selection Using Stepwise Regression Analysis: A set of important technical factors, as shown in Table II, which will affect the forecasting result, have been identified by [14]. These important input factors will be further selected through stepwise regression analysis (SRA) model in this research. There are totally six important indexes to be selected from and they are $X_1$ = six day moving average; $X_2$ = six day bias (BIAS); $X_3$ = six day relative strength index (RSI); $X_4$ = nine day stochastic line (KD); $X_5$ = the moving average divergence (MABIAS); and $X_6$ = the 13 days psychological line. The output is $Y$ = stock price.

The SRA is applied to determine the set of independent variables that most closely affect the dependent variable. This is accomplished by the repetition of a variable selection. The step-by-step procedure of the SRA approach is explained in details in the following.

Step 1) Calculate the correlation coefficient ($r$) of each input variable ($X_1, X_2 \cdots X_n$) and output data ($Y$). Then, a correlation matrix is derived.

Step 2) Rank each variable according to its square ($r^2$) from correlation matrix (suppose $X_i$ is the largest one in the current stage), and check the linear regression of this variable to the output data, i.e., derive a regression model as $\hat{Y} = f(X_i)$. $\alpha$ value is applied to consider the significance of each input variable. Repeat this process until all variables are tested. Finally, select those statistically significant variables for further verification and assume that these variables are ($X_1, X_2 \cdots X_k$).

Step 3) Calculate partial $F$ value for those statistically significant variables, as shown in (9), and choose the largest correlation coefficient among these input variables (assume that it is $X_j$). Then, derive another regression model $\hat{Y} = f(X_1, X_2 \cdots X_k)$ again

$$ F_i^* = \frac{\text{MSR}(X_j/X_i)}{\text{MSE}(X_j/X_i)} = \frac{\text{SSR}(X_j/X_i)/(k-2)}{\text{SSE}(X_j/X_i)/(n-k)}, i \in I $$

(8)

$$ F_i^* = \max_{1 \leq j \leq n, j \neq i} (F_j). $$

(9)

Step 4) Calculate the partial $F$ value of the original data for input variable $X_j$. If the value is smaller than a user defined threshold, it is removed from the model because $X_j$ is not statistically significant for the output.

Step 5) Repeat steps 3 to 4. If every input number’s partial $F$ value is greater than the user defined threshold, stop. It means that every input variable should have significant influences on output value. According to [10] and [13], we always set the threshold value as 4. If the $F$ value of a specific variable is greater than the user defined threshold, it is added to the model as a significant factor. When the $F$ value of a specific variable is smaller than a user defined threshold, it is removed from the model. The statistical software Statistical Package for the Social Sciences (SPSS) for Windows 10.0 was used for SRA in this research. The flow diagram of SRA is shown in Fig. 3.

2) TSK Fuzzy Rule Systems: The TSK fuzzy systems is selected as a universal function approximation for the stock prediction problems due to its ability to explain nonlinear relations using a relatively low number of simple rules. The structure of our TSK model is a multiple-input single-output fuzzy system
and its associated fuzzy inference method comprises a set of K IF-THEN rules in the following form:

\[ R_i : \text{If } x_1 \text{ is } A_{i1}, x_2 \text{ is } A_{i2}, \ldots, x_n \text{ is } A_{in}, \]

\[ \text{then } y_i = \beta_{i0} + \beta_{i1} x_1 + \cdots + \beta_{in} x_n, \]

where \( x_j, j = 1, \ldots, n, \) are the inputs of the fuzzy system, \( n \) is the number of inputs, \( A_{ij}, i = 1, \ldots, N, \) are fuzzy subsets, \( N \) is the number of fuzzy rules, \( y_i \) is the output of \( i \)th rule, and \( \beta_{ji} \) are parameters for consequence rules. It is a first-order TSK fuzzy rule system, and in this paper, Gaussian fuzzy membership functions are adopted

\[ A_{ij}(x_j) = \exp \left[ -\frac{(x_j - a_{ij})^2}{\sigma_{ij}^2} \right]. \]  

where \( a_{ij} \) and \( \sigma_{ij} \) are the mean and standard deviation of the Gaussian functions. Given a crisp input pair \( (x_1^0, \ldots, x_n^0) \); the crisp output of the TSK model is described by

\[ y = \frac{\sum_{i=1}^{N} w_i y_i}{\sum_{i=1}^{N} w_i} \]

where \( w_i \) is the strength of rule \( i \) determined by

\[ w_i = \prod_{j=1}^{n} A_{ij}(x_j^0) \]

and

\[ y_i = \beta_{i0} + \beta_{i1} x_1^0 + \cdots + \beta_{in} x_n^0. \]

The main task in TSK fuzzy-rule-based prediction is to determine the parameters in the fuzzy membership functions and in the rule consequences using a learning algorithm, given a set of training data specifying the functional mapping between the inputs and the output.

3) Data Clustering: The purpose of data clustering is to cluster the set of financial time series data into different groups, and data in each group will have a more homogeneous characteristic. However, it is very important to determine how many fuzzy rules should be generated beforehand. If a standard rule structure is used, rule explosion occurs when the number of inputs is high. To resolve this problem, we divide the training data into a number of clusters based on the output data (stock price) and one fuzzy rule is generated for each cluster [32]. By doing this, the number of fuzzy rules can be reduced effectively. Besides, we can determine the fuzzy membership functions using the mean and standard deviation of the data points that belongs to each cluster.

The \( K \)-means clustering algorithm is employed for data clustering. \( K \)-means is a nonhierarchical clustering technique in which the dataset is partitioned into \( K \) clusters. During the clustering, the data points are randomly assigned to the clusters to minimize the following squared error (SE):

\[ SE = \sum_{i=1}^{K} \sum_{p \in C_i} |p - m_i|^2 \]  

where \( p \) are data points in the cluster \( C_i \), \( m_i \) is the center of cluster \( C_i \), and \( K \) is the number of clusters.

Once the training data are clustered, we can calculate the parameters of the membership functions for each cluster as follows:

\[ a_{ij} = \frac{1}{S_i} \sum_{i=1}^{S_i} x_j \]

\[ \sigma_{ij} = \frac{1}{S_i - 1} (x_j - a_{ij})^2 \]

where \( S_i \) is the number of data points in cluster \( i \). In addition, the output of the training data is also normalized using the mean and standard deviation of the data in each cluster.

4) Optimization of the Parameters in Fuzzy Rules Using Simulated Annealing: The purpose of applying the SA is to find a set of best values for the parameters within the fuzzy rules. Traditionally, the parameter settings of TSK’s rules are generated using the gradient method. The generalized gradient algorithm searches for the solution in a multidimensional space along the steepest ascent direction. However, such a search can be extremely slow and ineffective if the equation has many plateaus distributed throughout the landscape. Therefore, this
method may not be able to derive an optimal solution and can be trapped in a local optimum, as shown in Fig. 4. In such cases, statistical search methods may offer better strategy in resolving both problems [60], [66].

One of the most widely used statistical search method is the SA [33], which uses the metropolis algorithm to decide whether to accept or reject a configuration that results in an increased cost during its attempts in searching for the minimum cost. The main characteristics of SA are its simplicity and the rapid convergence. To properly adjust parameters’ weights of TSK fuzzy rules, the SA approach is effective if the chosen energy, or cost function, for the global system is appropriate. In this study, the cost function is defined as the MAPE for the set of testing data, i.e., a series of stock index. The procedure of SA is well known, as described in [33]. First, it is necessary to generate random values of the parameters’ weights, and second, to compute the associated cost of the system. This cost will be minimized when the parameters’ weights achieve a global minimum, the method thus allowing escape from local minima.

The detailed set up of the parameters for SA will be described in the next section. Through a proper setup of the cooling schedule, finally SA can be applied to derive a set of near-optimal parameters for these TSK rules, as shown in Fig. 4.

5) Using K-Nearest-Neighbor as a Sliding Window: The basic idea of KNN [20] is to identify similar patterns of current data trend from the historic data. We use KNN as a sliding window to forecasting the data value for next day and use the current k data as a time window to search within the historic data to see if there are any similar patterns identified. Basically, our approach is categorized as a one-step ahead prediction. The selected data are preprocessed with the wavelet, and then, TSK model is applied to generate a set of fuzzy rules for prediction of stock price. In addition, the KNN sliding window is further applied to reduce the forecasting errors. The set of historic data is divided into training and testing set for cross validation. The KNN is simpler than other SC approaches because there is no model to train on the data series. Instead, the data series is being searched for situations similar to the current, each time a forecast needs to be made.

To describe the KNN process, several terms have to be defined first. Assume the window size is \( L \), which means there are \( L \) data in each window to be considered. The final data points of the data series are the reference data, and the length of the reference is the window size. To forecast the data series’ next data point, the reference is compared to the first group of data points in the historical data series, called a candidate, and an error is computed. Then, the reference is moved one data point forward to the next candidate and another error is computed, and so on. An error is calculated by subtracting the candidate value from the reference value. All errors of the testing data are sorted and stored in an array. Assume that the number of nearest neighbors is \( H \). Then, the smallest \( H \) errors corresponding to these \( H \) candidates will be selected. Finally, the forecasted value will be equal to the average of these \( k \) data points. Then, to forecast the next data point, the process is repeated with the previously forecasted data point appended to the end of the data series. This process can be iteratively repeated until all \( n \) data points are calculated.

Use KNN to calculate the new forecasted value.

**Step 1)** Use original data as a contrast data. Suppose to forecast number \( i \) data from index number \( t + 1 \), i.e., number \( \hat{X}_{i, t+1} \). value.

**Step 2)** Use number \( t \) to \( t-L+1 \) data for contrast base. Using the sliding window method, one by one compare the data from \( 1 \) to \( t-L \), and also calculate the Euclidean distance from every interval \( D_j^{(1)} \), and find the corresponding forecasting value \( F_j^{(1)} \).

\[
D_j^{(1)} = \sqrt{\sum_{l=1}^{L} (X_{i,l-L+l} - \hat{X}_{i,l+j-1})^2}
\]

\[
F_j^{(1)} = X_{i,j+L}.
\] (16)

In (14), \( j = 1 \sim t-L \).

**Step 3)** Consider all \( D_j^{(1)} \) and find the \( k \)th smallest number. It is KNN’s \( K \) option value.

**Step 4)** Use the weighted voting method to find the last forecasting value \( \hat{X}_{i,t+1} \). The equation is

\[
\hat{X}_{i,t+1} = \frac{\sum_{k=1}^{H} F_k/W_k}{\sum_{k=1}^{H} 1/W_k}.
\] (17)

\( W_k \) means the \( k \)th smallest \( D_j^{(1)} \) value, \( F_k \) means the \( F_j^{(1)} \) value corresponding to the \( k \)th smallest \( D_j^{(1)} \) value, \( H = 1-k \). And the parameter set of the sliding windows is \((L, H)\). A simple example for KNN forecasting with window size \( L = 3 \) and \( H = 2 \) is shown in Fig. 5.

C. Different Models to be Compared With

In this research, we use traditional back-propagation neural networks (BPNs), the MRM, and a forecasting method by integrating GA with Wang and Mendal’s algorithm for fuzzy rule generation (GAWM) [13], [58] to compare with our wavelet TSK fuzzy rule forecasting system. These three compared models will be briefly introduced.

BPN [49] is a popular system that has been widely employed in financial forecasting. The most popular training method for BPN is the supervised learning, i.e., learning by samples, which will be selected in this research to train the system. After learning (or training), the trained connection weights can be used for the...
TABLE II
TECHNICAL INDICES USED AS INPUT VARIABLES

<table>
<thead>
<tr>
<th>Technical index</th>
<th>Explanation</th>
<th>Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td>six days Moving Average(MA)</td>
<td>Moving averages are used to emphasize the direction of a trend and smooth out price and volume fluctuations that can confuse interpretation.</td>
<td>[ M \ A (n)<em>t = \frac{\sum</em>{i=k-n}^{k} P_i}{n} ]</td>
</tr>
<tr>
<td>six days bias (BIAS)</td>
<td>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.</td>
<td>[ B I A S (n)_t = \frac{P_t - M A (n)_t}{M A (n)_t} \times 100% ]</td>
</tr>
<tr>
<td>six days relative strength index (RSI)</td>
<td>RSI compares the magnitude of recent gains to recent losses in an attempt to determine overbought and oversold conditions of an asset.</td>
<td>[ R S (n)<em>t = \frac{U P (n)</em>{avg}}{D O W N (n)_{avg}} \times 100% = 100% \left( \frac{1}{1+R S (n)} \right) ]</td>
</tr>
<tr>
<td>nine days Stochastic line (K, D)</td>
<td>The stochastic line K and line D are used to determine the signals of over-purchasing, over-selling, or deviation.</td>
<td>[ R S V (n)_t = \frac{P_t - L (n)_t}{H (n)_t - L (n)_t} \times 100% ]</td>
</tr>
<tr>
<td>Moving Average Convergence and Divergence (MACD)</td>
<td>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. 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.</td>
<td>[ D I F_t = \frac{H_t + L_t + 2 \times C_t}{4}, \quad E M A (n)<em>t = \frac{1}{n} \sum</em>{i=t-n}^{t} D I_t ]</td>
</tr>
<tr>
<td>13 days Psychological Line (PSY)</td>
<td>The output factor (Y) is the stock price and the input factors include six day moving average (X1), six day bias (X2), six day RSI (X3), nine day stochastic line (X4), moving average divergence (X5), and the 13 days psychological line (X6). The multiple regression formula of this problem can be defined as follows:</td>
<td>[ Y = a_1 X_1 + a_2 X_2 + a_3 X_3 + a_4 X_4 + a_5 X_5 + a_6 X_6 + b ]</td>
</tr>
</tbody>
</table>

Authorized licensed use limited to: Hong Kong. Downloaded on February 8, 2009 at 00:06 from IEEE Xplore. Restrictions apply.
D. Performance Measures

There are many measures of prediction accuracy used to compare forecasting methods out of sample [10]. In considering about the dimensionality of the data, in this research, there are six input factors and one output value, and 494 training data. The mean square error (MSE) of each model is very large after training this data set. Since the purpose of the stock prediction is to make profit instead of just predict the future price accurately, we use MAPE as a performance measure instead of MSE. The equation of MAPE is listed later.

1) Mean Absolute Percentage Error: The accuracy of predictions was measured with the following indicator, i.e., MAPE. The average forecast error is measured as a percentage of historical results. The absolute value allows for the effect of adding different signs. It is calculated as follows:

\[
\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{X_t - F_t}{X_t} \right| \times 100
\]  

where \(X_t\) is the true value and \(F_t\) is the predicted value at time \(t\). The MAPE is an average over \(n\) test sets.

IV. SIMULATION RESULTS

The Taiwan Stock Exchange (TSE) began operations since 1962. At the end of January, 2005, Taiwan Stock Exchange Corporation (TSEC) had 699 listed companies with market capital topping NT$13.7 trillion (US $396 billion). Most stock trading goes to the listed IT companies and the trading value of TSE stock market places it in the top ten of stock exchanges in the world.

The data set applied for test in this research is the TSE index, and it has been decomposed into three different sets: the training data, test data, and validation data. The data for TSE index are from July 18, 2003 to December 31, 2005, totally 614 records. During this period, the stock market has gone through a rough up and down period owing to the national political issues. Therefore, these data are very representative and suitable for study and analysis. The first 492 records will be training and cross-validation data and the rest of the data, i.e., 122 records will be for out-of-sample test data. To avoid the interaction among these factors, we will test each factor using SRA 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.

Before training the TSK fuzzy model, a wavelet transformation has been applied to preprocess the data. According to the MAPE, a three-level wavelet preprocessing is thus applied. Through this process, the noise in the original data can be removed. The result of a wavelet-based decomposition process is depicted in Fig. 6. According to the RSA method described in Section III-B, six input variables are finally selected as the
inputs to the TSK model to predict the stock price. They are six day moving average (MA), six day bias (BIAS), six day RSI, nine day stochastic line (KD), the moving average divergence (MABIAS), and the 13 days psychological line (PSY).

In the k-means clustering algorithm, the number of clusters must be predefined. It is a very interesting subject to be further investigated since there is no exact theorem explaining the effect of number of clusters to the forecasting accuracy. To check the sensitivity of the performance of our model on the number of clusters, various numbers of data clusters are investigated. In the experiment, the stock price data were clustered into two to eight clusters. The performance (MAPE) of the algorithms with different number of clusters is shown in Fig. 7.

As can be observed from the figure, MAPE will start to decrease as the number of clusters increase. However, as the number of clusters reaches a certain value, MAPE starts to increase. Part of the reasons is because when the number of clusters is too large, the number of the data in each cluster is too small. These data in each cluster are not representative enough to generate a model to forecast the future stock index. Therefore, in this research, the number of clusters will be three since it provides best performance (the smallest MAPE). This number of clusters is not definite and it has to be decided experimentally for different application purposes.

The parameter setting in three different levels for the SA is provided in Table III. Then, we use statistical software Minitab R14 to run the Taguchi experiments and the results of different factor levels are shown in Table IV. Table V lists the final setting for each factor in SA procedure. The factor response graph of these experimental results is shown in Fig. 8.

The convergence diagram of the learning process of the rule consequence parameters using the SA is shown in Fig. 9. Finally, the MAPE of the forecasting model gradually decreases to 3.8% after the temperature drops to a certain level. It is justified that the proposed SA approach can find near-optimal solutions for the set of parameters of consequence rules.

We compared the proposed hybrid method combining the wavelet and TSK fuzzy rules with three existing methods. To justify the use of SA and KNN sliding window, a set of experimental results are listed in Table VI. In this table, rule number is decided by (14) and (15), and the first number means the window size $L$; the second number means the number of best neighborhood data $H$. According to a series of experiments where $L$ is setup as 2, 4, 6, and 8 and $H$ is setup as 2, 3, 4, and 5, the best forecasting result in terms of the minimum MAPE value from the hybrid model is in (2,5) with an average of 0.792.

The four different algorithms to be compared with are the traditional back-propagation neural networks (BPNs), a standard TSK, the MRM, and a forecasting method by integrating GA with Wang and Mendal’s algorithm for fuzzy rule generation (GAWM) [13], [58]. Tables VII and VIII are the best parameter setting for BPN and GAWM using design of experiments. Fig. 10 also reveals the experimental results that GAWM converged after 100 generations. Table VIII shows the MAPE value for all different methods.

As observed from Table IX, MRM has the largest MAPE value and part of the reasons is because MRM cannot fully explain the nonlinear relationship among the stock price and the technical indexes. BPN also has a large error as compared with other models and that is due to the tremendous noise and complex dimensionality of stock price data. Besides, the quantity of data itself and the input variables may also interfere with each other. In addition, BP learning algorithm is subject to getting stuck in a local optimum, while the TSK is less likely. Therefore, the result may not be that convincing. In addition, BPN methods do not provide an insight into the nature of the interactions between the technical indicators and the stock market fluctuations. As for GAWM, the fuzzy rules generated from the training data are very large when compared with TSK and these rules may interact with each other.

From the experimental tests, we can observe that TSK fuzzy system is more suitable in handling large amount of data. The set of data is clustered according its mean and standard deviation. After this clustering, these set of data will be decomposed into couples of subclusters. These subclusters have more homogeneous characteristics within themselves, and each subcluster will be transformed into a TSK fuzzy rule. Therefore, the fuzzy rules generated from TSK are quite small since each cluster of data only generates one single rule. For our experimental tests, the clusters have been preset into two to eight; therefore, there are two to eight fuzzy rules within each

---

**TABLE III**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Temperature</td>
<td>100</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td>Stop Criteria</td>
<td>0.01</td>
<td>0.05</td>
<td>0.10</td>
</tr>
<tr>
<td>Temperature</td>
<td>100</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td>Total Number of Runs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooling parameter</td>
<td>0.07</td>
<td>0.08</td>
<td>0.09</td>
</tr>
</tbody>
</table>

---

*Fig. 7. MAPEs of the hybrid model for different number of data clusters.*
### TABLE IV
**Taguchi Experimental Result of Different Factor Levels**

<table>
<thead>
<tr>
<th>Trial</th>
<th>Factor Level</th>
<th>Experiments</th>
<th>S/N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

### TABLE V
**Best Parameter Set for SA**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>100</td>
</tr>
<tr>
<td>Temperature (A)</td>
<td>0.01</td>
</tr>
<tr>
<td>Stop Criteria</td>
<td></td>
</tr>
<tr>
<td>Temperature(B)</td>
<td>0.01</td>
</tr>
<tr>
<td>Executive Number</td>
<td>500</td>
</tr>
<tr>
<td>Cooling parameter</td>
<td>0.09</td>
</tr>
</tbody>
</table>

### TABLE VI
**Results of Different Number of Sliding Windows (L) and H Nearest Neighbors Applied in TSE Index Forecasting (in Percent)**

<table>
<thead>
<tr>
<th>Trial</th>
<th>(L,H)</th>
<th>CPU Time(seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(2,2)</td>
<td>5.27</td>
</tr>
<tr>
<td>2</td>
<td>(2,3)</td>
<td>5.16</td>
</tr>
<tr>
<td>3</td>
<td>(2,4)</td>
<td>5.26</td>
</tr>
<tr>
<td>4</td>
<td>(2,5)</td>
<td>5.13</td>
</tr>
<tr>
<td>5</td>
<td>(2,6)</td>
<td>6.02</td>
</tr>
<tr>
<td>6</td>
<td>(2,7)</td>
<td>5.86</td>
</tr>
<tr>
<td>7</td>
<td>(2,8)</td>
<td>5.71</td>
</tr>
<tr>
<td>8</td>
<td>(2,9)</td>
<td>5.62</td>
</tr>
<tr>
<td>9</td>
<td>(2,10)</td>
<td>5.55</td>
</tr>
</tbody>
</table>

### TABLE VII
**BPN Parameter Set**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>In put layer</td>
<td>6 (SRA choose important factors)</td>
</tr>
<tr>
<td>number of neurons in the output layer</td>
<td>1 single hidden layer</td>
</tr>
<tr>
<td>Hidden Layer</td>
<td>3 (According to experimental result)</td>
</tr>
<tr>
<td>network-learning rule</td>
<td>Gradient Method</td>
</tr>
<tr>
<td>transformation function</td>
<td>sigmoid function</td>
</tr>
<tr>
<td>learning rate</td>
<td>0.1</td>
</tr>
<tr>
<td>Executive Number (C)</td>
<td>50000</td>
</tr>
</tbody>
</table>

---

**Fig. 8.** Factor response graph.

**Fig. 9.** SA training astringent diagram.

System. The best forecasting results will be in three clusters. In addition, KNN has been further applied to reduce the forecasting errors. That is why, TSK has a better forecasting accuracy.
preprocessing is new for stock price forecasting. Due to its very promising performance, we are going to apply the system for real-time daily trading.

In the future, a different TSK fuzzy model, such as a nonlinear model using NNs as a consequence, can be further applied in a more complex time series problem. In addition, more advanced pattern matching algorithm can be embedded in the system to retrieve significant patterns from the historical stock data for comparison with the current trend of the data. As a result, intelligent trading signals instead of stock price can be identified.

V. CONCLUSION

This paper proposed a TSK fuzzy model for stock price prediction. To facilitate the prediction, the data are preprocessed using the Haar wavelet. Then, SRA technique is employed to select the most relevant factors for prediction. To avoid rule explosion, the \( k \)-means clustering algorithm is employed to group the data into a number of clusters and one fuzzy rule is generated for each cluster. As an additional benefit, the fuzzy membership function can be determined automatically using the mean and variance of the data in each cluster. The parameters in the consequences of the TSK rules are optimized using the SA. A KNN sliding window is applied to retrieve the similar patterns in the historical data and further adjust the forecasted value from the TSK model.

The proposed model is compared with the BNP, TSK, MRM, and GAWM for stock price prediction. Simulation results show that the TSK model with wavelet-based preprocessing greatly outperforms the other three models. To the best of knowledge, the combination of the TSK fuzzy model with wavelet-based preprocessing is new for stock price forecasting. Due to its very promising performance, we are going to apply the system for real-time daily trading.

In the future, a different TSK fuzzy model, such as a nonlinear model using NNs as a consequence, can be further applied in a more complex time series problem. In addition, more advanced pattern matching algorithm can be embedded in the system to retrieve significant patterns from the historical stock data for comparison with the current trend of the data. As a result, intelligent trading signals instead of stock price can be identified.

REFERENCES


---

**TABLE VIII**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population</td>
<td>100</td>
</tr>
<tr>
<td>Generation</td>
<td>100</td>
</tr>
<tr>
<td>Reproduction/selection</td>
<td>roulette wheel selection</td>
</tr>
<tr>
<td>Crossover</td>
<td>0.8 (two-point crossover method)</td>
</tr>
<tr>
<td>Mutation</td>
<td>0.01 (one-point mutation method)</td>
</tr>
<tr>
<td>The Best Number of Clusters</td>
<td>7</td>
</tr>
</tbody>
</table>

**TABLE IX**

<table>
<thead>
<tr>
<th>Method</th>
<th>Wavelet +TSK</th>
<th>Standard</th>
<th>GAWM</th>
<th>BPN</th>
<th>MRM</th>
<th>Ave MAPE</th>
<th>Best MAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.787</td>
<td>0.875</td>
<td>0.569</td>
<td>4.85</td>
<td>8.03</td>
<td>0.792</td>
<td>0.813</td>
</tr>
<tr>
<td>Ave MAPE</td>
<td>0.792</td>
<td>0.813</td>
<td>1.04</td>
<td>5.24</td>
<td>12.04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 10 Convergence diagram of GAW& M model.


Pei-Chann Chang received the M.S. and Ph.D. degrees from Lehigh University, Bethlehem, PA, in 1985 and 1989. He is currently a Professor at the Department of Information Management, Yuan Ze University, Taoyuan, Taiwan, R.O.C. His current research interests include financial time series forecasting, evolutionary computation, fuzzy neural applications, production scheduling, forecasting, case-based reasoning, and applications of soft computing. He is the author or coauthor of more than 60 papers published in international journals. He is the Senior Editor of the Journal of Chinese Institute of Industrial Engineering (JCIE).

Chin-Yuan Fan received the B.S. degree from the Chinese-Culture University, Taipei, Taiwan, R.O.C., 2001, and the M.S. degree from Da-Yeh University, Changhua, Taiwan, in 2003, both in management. He is currently working toward the Ph.D. degree at the Industrial Engineering and Management Department, Yuan Ze University, Taoyuan, Taiwan.

His current research interests include applications of soft computing, financial time series forecasting, multiobjective optimization problems, and multicriteria decision making.