Combination of Artificial Neural-Network Forecasters for Prediction of Natural Gas Consumption

Alireza Khotanzad, *Member, IEEE*, Hassan Elragal, *Member, IEEE*, and Tsun-Liang Lu, *Member, IEEE*

***Abstract*—The focus of this paper is on combination of artificial neural-network (ANN) forecasters with application to the prediction of daily natural gas consumption needed by gas utilities. ANN forecasters can model the complex relationship between weather parameters and previous gas consumption with the future consumption. A two-stage system is proposed with the first stage containing two ANN forecasters, a multilayer feedforward ANN and a functional link ANN. These forecasters are initially trained with the error backpropagation algorithm, but an adaptive strategy is employed to adjust their weights during on-line forecasting. The second stage consists of a combination module to mix the two individual forecasts produced in the first stage. Eight different combination algorithms are examined, they are based on: averaging, recursive least squares, fuzzy logic, feedforward ANN, functional link ANN, temperature space approach, Karmarkar's linear programming algorithm and adaptive mixture of local experts (modular neural networks). The performance is tested on real data from six different gas utilities. The results indicate that combination strategies based on a single ANN outperform the other approaches.**

***Index Terms*—Artificial neural-network forecasters, combination of forecasts, feedforward artificial neural networks, functional link artificial neural networks, fuzzy logic, gas load forecasting, linear programming, mixture of experts, multistage forecaster, prediction of gas consumption.**

I. Introduction

A GAS UTILITY, also known as a local distribution company (LDC), buys its gas from a pipeline company on a daily basis. The LDC needs to provide the pipeline company with the prediction of its need for the future days so the pipeline company can plan for the required gas. If the forecast error is large, the pipeline company assesses a financial penalty against the LDC. Thus, producing accurate gas consumption forecasts is important for a LDC from an operational point of view.

Daily gas consumption referred to as *daily gas sendout*, is influenced by many factors. Since natural gas is mostly used for heating, the most important factor is temperature. Another important factor is wind speed, in that buildings lose more heat on a windy day than on a calm day. Heat loss is also a dynamic process, hence climatic conditions of previous days are also contributing factors to gas consumption. Many industrial customers and some commercial customers shut down over weekends; thus the day-of-week is considered a factor as well. Another relevant parameter is recent consumption trend, which represents a general profile for the demand.

The relationship between these parameters and the future sendout is complex and nonlinear. We use the function approximation ability of the artificial neural networks (ANN's) to model this relationship.

There have been relatively few attempts in the past to develop gas-forecasting systems. The linear regression-based systems make up the majority of the existing computer-based forecasters in industry. In these systems, gas demand is predicted as a linear function of the recent demand values and weather data. However, as pointed out, due to nonlinear nature of the problem such methods cannot capture this relationship adequately. There has been one other attempt to build an ANN-based predictor for gas forecasting problem [2]. This system uses a nonadaptive ANN.

Traditional forecasting techniques use a single forecaster to carry out the task [24]. However, this single forecaster may not be the best for all situations and/or databases. For instance, an ANN forecaster with a particular topology might work well for some data bases, whereas another ANN topology would perform better for others. If these different forecasters are complementary, the combination of their decisions will result in improved accuracy. In this paper, we focus on decision combination of two topologically different ANN forecasters, a feedforward multilayer perceptron and a functional-link ANN; both modified to become adaptive during on-line operation.

The idea of forecast combination is pursued in several previous studies [1], [3], [4], [6]–[10], [16]–[18], [24] and is shown to improve the accuracy. Most of these methods focus on minimizing the variance of the forecast errors. In this work, we investigate several linear and nonlinear combination approaches including ANN-based ones. The nonlinear ANN-based combiners proposed here outperform the traditional linear techniques.

In the work reported here, only two forecasts are combined. In most other decision combination approaches a larger number of forecasts are used. We did investigate utilizing additional forecasters such as, similar day-type forecasters, ANN forecasters trained with other than backpropagation algorithm, ANN forecasters focusing on different regions of data, etc. The results indicated little improvement over using just the two forecasters considered here. It should be noted that the discussed combination schemes could be applied to a larger number of forecasts if so desired.

II. Adaptive ANN Forecasters

The proposed gas forecasting system consists of two stages. In the first stage, two adaptive ANN forecasters run in parallel and produce independent forecasts of the daily gas sendout. These forecasts are then input to the second stage, which includes a forecast combination module.

Manuscript received August 27, 1998; revised August 26, 1999 and December 20, 1999.

The authors are with the Electrical Engineering Department, Southern Methodist University, Dallas, TX 75275 USA.

Publisher Item Identifier S 1045-9227(00)03003-4.

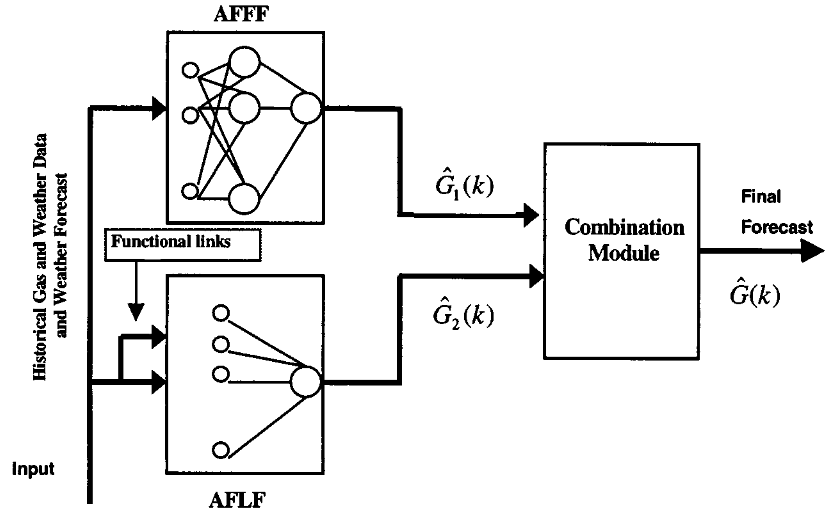

Fig. 1. Schematic of the forecasting system.

The details of the two ANN forecasters are presented in Sections II-A and II-B.

A. Adaptive Feedforward Forecaster (AFFF)

The first ANN forecaster is an adaptive feedforward forecaster (AFFF). AFFF is a three-layer (input, hidden, output) feedforward perceptron ANN with sigmoid nodes and trained with the error backpropagation (BP) learning rule [20]. The input vector of the AFFF consists of the following parameters:

- gas sendout of the past two days, $G(k-1)$, $G(k-2)$;
- average daily temperature of the past two days, $T(k-1)$, $T(k-2)$;
- average daily wind speed of the past day, $W(k-1)$;
- forecast average daily temperature of the next day, $\hat{T}(k)$;
- forecast average daily wind speed of the next day, $\hat{W}(k)$;
- day of week indicators.

All inputs are scaled to lie in [0, 1] range using a reasonable estimate of the maximum and minimum values for each input type, G, T, and W.

The output of AFFF consists of a single node whose output is the forecast sendout of the next day denoted by $\hat{G}_1(k)$.

Unlike conventional feedforward ANN's, AFFF is made to be an adaptive forecaster in the sense that its weights change and adapt on a daily basis during the on-line forecasting phase [14], [15]. Initially AFFF is trained with the historical data using BP algorithm in the traditional manner. However, after this initial training, the weights are adaptively updated at the end of each day when actual data for that day becomes available. This task is accomplished by running AFFF in a backcast mode for the previous day using actual weather data. Since the actual sendout for that day is also available, a small-scale BP operation is performed involving the data of this day and a few days before. This mini-training with the most recent data results in a slight adjustment of the weights and biases them toward the recent trend in data. This adaptive adjustment is quite helpful in tracking the nonstationary behavior of the sendout and also accounting for gradual load growth (loss) that an LDC could experience from one year to the next.

B. Adaptive Functional Link Forecaster (AFLF)

The second ANN forecaster is an adaptive functional link forecaster (AFLF). AFLF is a two layer feedforward ANN with no hidden layer and a sigmoid output node trained with BP learning rule. The basic idea behind a functional link ANN is the use of nonlinear transformation of some components of the input vector before it is fed to the input layer of the network.

The inputs selected for AFLF consist of the same inputs used for AFFF plus the following additional functional links of these variables:

- functions of previous day's sendout: $G^2(k-1)$, $\cos[\pi.G(k-1)]$;
- functions of previous day's temperature: $T^2(k-1)$, $\cos[\pi.T(k-1)]$;
- functions of next day's forecast temperature: $\hat{T}^2(k)$, $\cos[\pi.\hat{T}(k)]$.

Again, note that the G and T values are normalized to [0, 1] range.

The single output of AFLF is the forecast sendout of the next day denoted by $\hat{G}_2(k)$. A similar adaptive strategy to AFFF is employed here to update AFLF weights during the on-line forecasting phase.

III. Forecast Combination

The second stage of the proposed system consists of a forecast combination module that mixes the AFFF and AFLF forecasts in either a linear or nonlinear fashion as illustrated in Fig. 1. In this work we investigate eight different approaches for this combination task. These methods are based on the following:

1) average of forecasts;
2) recursive least squares;
3) fuzzy logic;
4) adaptive feedforward ANN;
5) adaptive functional link ANN;
6) temperature space;
7) Karmarkar's linear programming algorithm;

8) adaptive mixture of local experts (modular neural network).

A. Combination Using Averaging

A simple average of the two forecasts is the first considered combination approach. The final forecast denoted by $\hat{G}_A(k)$ is

$$\hat{G}_A(k) = \frac{\hat{G}_1(k) + \hat{G}_2(k)}{2}. \tag{1}$$

B. Combination Using Recursive Least Squares

The recursive least square (RLS) [11] approach is an adaptive linear combination strategy according to

$$\hat{G}_{\mathrm{RLS}}(k) = \alpha_1 \hat{G}_1(k) + \alpha_2 \hat{G}_2(k). \tag{2}$$

The α_i parameters are computed in such a way to minimize the sum of square of differences between past actual sendout and the corresponding forecasts

$$J = \sum_{j=1}^{k-1} \beta^{(k-1)-j} \left[G(j) - \hat{G}_{\mathrm{RLS}}(j)\right]^2 \tag{3}$$

where j is an index starting from the beginning of the past historical data extending to the previous day, β is a forgetting factor in the range of $0 < \beta < 1$, whose function is to de-emphasize old data.

The initial values of α_i are computed based on the training data. These values change every day when actual data of previous day becomes available. A positiveness constraint is also imposed on α_i. If at any iteration the value of α_i becomes negative, the average of the forecasts is used in place of the RLS combination for that iteration.

C. Combination Using Fuzzy Logic

Fuzzy logic has many applications in different areas such as decision making, classification, pattern recognition, and control [5], [19], [22], [23]. This methodology is adapted for combining the outputs of the two forecasters in the following manner.

1) Compute $\hat{G}_1$ and $\hat{G}_2$ for all the samples in the training set.
2) Cluster these forecasts in the two-dimensional space of $[\hat{G}_1, \hat{G}_2]$ into C clusters using subtractive clustering algorithm [5].
3) Consider cluster centers $[\hat{G}^*_{1i}, \hat{G}^*_{2i}]$, $i = 1, 2, \cdots, C$ as a fuzzy representation of the data.
4) During on-line forecasting phase when the next day's forecasts, $[\hat{G}_1, \hat{G}_2]$, are generated, compute a cluster membership measure for each forecast as

$$A^i_j(\hat{G}_j) = \exp\left[-\alpha \left\|\hat{G}_j - \hat{G}^*_{ji}\right\|^2\right]$$
$$j = 1, 2 \quad i = 1, 2, \cdots, C \tag{4}$$

where α is a positive constant and $A^i_j(\hat{G}_j)$ denotes $\hat{G}_j$'s membership to cluster i.

5) The combined forecast, $\hat{G}_{FZ}$, is computed using fuzzy implication given by Takagi–Sugeno fuzzy inference model [23] with fuzzy rules defined as

R^i: if $\hat{G}_1$ is A^i_1 and $\hat{G}_2$ is A^i_2 then
$\hat{G}^i = p^i_0 + p^i_1 * \hat{G}_1 + p^i_2 * \hat{G}_2, \quad i = 1, 2, \cdots, C.$

The final output, i.e., combined forecast, is given by

$$\hat{G}_{FZ} = \frac{\sum_{i=1}^{C} \left\{\left(A^i_1(\hat{G}_1) \diamondsuit A^i_2(\hat{G}_2)\right)\left(p^i_0 + p^i_1 * \hat{G}_1 + p^i_2 * \hat{G}_2\right)\right\}}{\sum_{i=1}^{C}\left(A^i_1(\hat{G}_1) \diamondsuit A^i_2(\hat{G}_2)\right)} \tag{5}$$

where $\diamondsuit$ stands for the minimum operation.

In simple terms, for each cluster, the combined forecast associated with each input vector, $[\hat{G}_1, \hat{G}_2]$, is considered to be a linear combination of the inputs. The minimum membership of a given input vector to each cluster is taken as a measure of how much of the respective linear combination should be used for the final combination.

The p^i_j coefficients are computed from the training set in the following manner.

Let

$$\beta_i = \frac{\left(A^i_1(\hat{G}_1) \diamondsuit A^i_2(\hat{G}_2)\right)}{\sum_{i=1}^{C}\left(A^i_1(\hat{G}_1) \diamondsuit A^i_2(\hat{G}_2)\right)}$$
$$\therefore \hat{G}_{FZ} = \sum_{i=1}^{C} \beta_i \left(p^i_0 + p^i_1 * \hat{G}_1 + p^i_2 * \hat{G}_2\right)$$
$$\therefore \hat{G}_{FZ} = \sum_{i=1}^{C} \left(p^i_0 \beta_i + p^i_1 \beta_i * \hat{G}_1 + p^i_2 \beta_i * \hat{G}_2\right). \tag{6}$$

Assume the training data set consists of k input–output pairs $[\hat{G}^m_1, \hat{G}^m_2] \Rightarrow G^m$, $m = 1, 2, \cdots, k$, where G^m is the actual sendout corresponding to $\hat{G}^m_1$ and $\hat{G}^m_2$. Equation (6) can be written as

$$\boldsymbol{Y} = \boldsymbol{X}\,\boldsymbol{P} \tag{7}$$

where $\boldsymbol{Y}$, $\boldsymbol{X}$, and $\boldsymbol{P}$ are as shown in (7a) at the bottom of the following page, with

$$\beta_{ij} = \frac{\left(A^i_1(\hat{G}^j_1) \diamondsuit A^i_2(\hat{G}^j_2)\right)}{\sum_{i=1}^{C}\left(A^i_1(\hat{G}^j_1) \diamondsuit A^i_2(\hat{G}^j_2)\right)}.$$

The unknown vector $\boldsymbol{P}$ could be solved for using the pseudoinverse operation.

The number of clusters C in the described approach is computed using cross-validation method on the training set.

D. Combination Using Adaptive Feedforward Perceptron ANN

In this combination scheme, $\hat{G}_1$ and $\hat{G}_2$ are combined in a nonlinear fashion using an adaptive feedforward (AFF) perceptron ANN whose architecture and operation is similar to AFFF. The use of an ANN combiner offers the potential of learning the underlying relationship that exist between the final forecast and $\hat{G}_1$ and $\hat{G}_2$.

In this approach it was found that the use of the forecast average temperature for the next day as an input to the AFF combiner could improve the accuracy. As a result, the input vector to the AFF consists of $\hat{G}_1(k)$, $\hat{G}_2(k)$, and $\hat{T}(k)$. In other words, the combined forecast denoted by $\hat{G}_{\mathrm{AFF}}(k)$ is

$$\hat{G}_{\mathrm{AFF}}(k) = f\left(\hat{G}_1(k),\, \hat{G}_2(k),\, \hat{T}(k)\right) \tag{8}$$

where f represents the overall nonlinear function of the AFF combiner.

E. Combination Using Adaptive Functional Link ANN

This nonlinear combination strategy uses an adaptive functional link (AFL) ANN to perform the task. Again, the architecture and operation is similar to AFLF described before.

The input to the AFL combiner is the same as those used for the AFF combiner plus the additional six functional links of

$$\hat{G}_1^2(k),\, \cos\left[\pi.\hat{G}_1(k)\right],\, \hat{G}_2^2(k),\, \cos\left[\pi.\hat{G}_2(k)\right]$$
$$\hat{T}^2(k), \cos\left[\pi.\hat{T}(k)\right].$$

The combined forecast denoted by $\hat{G}_{\mathrm{AFL}}(k)$ is

$$\begin{aligned}\hat{G}_{\mathrm{AFL}}(k) = g\Big(&\hat{G}_1(k),\, fl(\hat{G}_1(k)),\, \hat{G}_2(k),\, fl(\hat{G}_2(k)),\\ &\hat{T}(k),\, fl(\hat{T}(k))\Big)\end{aligned} \tag{9}$$

where g is the overall nonlinear function of the AFL and fl represents the functional links of the inputs.

F. Combination Using Temperature Space

This combination method is based on the assumption that the behavior of the considered forecasters could be categorized according to temperature. Based on this assumption, a two-dimensional temperature space is constructed for the average daily temperature of two consecutive days, i.e., one dimension corresponding to $T(k-1)$ (previous day's temperature) and the other to $\hat{T}(k)$ (next day's temperature forecast). Each dimension is then quantized into Q levels ($Q = 5$ used in this study) resulting in the division of the temperature space into Q^2 cells. Each sample in the training set is then allocated to a cell in this space according to its respective $T(k)$ and $T(k-1)$. The samples falling in each cell are grouped into one set and the parameters for the best linear combination of them are computed using a least squares approach. Thus, forecast combination for each cell p denoted by $\hat{G}_{TS}(k)$ is

$$\hat{G}_{TS_p}(k) = \alpha_{1p}\hat{G}_1(k) + \alpha_{2p}\hat{G}_2(k) \qquad p = 1,\, 2,\, \cdots,\, Q^2. \tag{10}$$

The α_{ip} parameters are computed by minimizing

$$J = \frac{1}{k}\sum_{j=1}^{k}\left[G(j) - \hat{G}_{TS_p}(j)\right]^2 \tag{11}$$

where k is the number of samples falling in the pth cell, and j represents sample index.

During on-line forecasting, each $[\hat{G}_1(k),\, \hat{G}_2(k)]$ sample is assigned to a cell according to $\hat{T}(k)$ and $T(k-1)$ and the α_{ip} parameters corresponding to the assigned cell is used to calculate $\hat{G}_{TS}(k)$. Upon adding an additional sample to a cell, the α_{ip} parameters are updated accordingly when the actual sendout for the respective sample becomes available. Also, if a sample falls into either an empty cell or a cell with only one sample, then the average of $\hat{G}_1(k)$ and $\hat{G}_2(k)$ is used as $\hat{G}_{TS}(k)$. If any case results in a negative α_{ip} value, the average is used instead.

G. Combination Using Karmarkar's Linear Programming Algorithm

The next combination method is based on Karmarkar's linear programming algorithm, which is an alternative to the simplex method. It is shown that Karmarkar's algorithm outperforms the simplex method for many practical linear programming problems [13].

Ruzinsky and Olsen [21], present an iterative algorithm for least absolute deviation (LAD) minimization based on a variant of Karmarkar's linear programming algorithm. This method is used here to generate the final forecast denoted by $\hat{G}_{LP}(k)$ which is a linear combination of $\hat{G}_1(k)$ and $\hat{G}_2(k)$ according to

$$\hat{G}_{LP}(k) = \alpha_1\hat{G}_1(k) + \alpha_2\hat{G}_2(k). \tag{12}$$

$$\begin{aligned}
\boldsymbol{Y} &= [G^1,\, G^2,\, \cdots,\, G^k]^T\\
\boldsymbol{P} &= [p_0^1\, p_0^2 \cdots p_0^C \quad p_1^1\, p_1^2 \cdots p_1^C \quad p_2^1\, p_2^2 \cdots p_2^C\,]^T,\\
\boldsymbol{X} &= \begin{bmatrix}
\beta_{11}\, \beta_{21} \cdots \beta_{c1} & \beta_{11}\hat{G}_1^1\;\; \beta_{21}\hat{G}_1^1 \cdots \beta_{c1}\hat{G}_1^1 & \beta_{11}\hat{G}_2^{1^1}\;\; \beta_{21}\hat{G}_2^1 \cdots \beta_{c1}\hat{G}_2^1\\
\beta_{12}\, \beta_{22} \cdots \beta_{c2} & \beta_{12}\hat{G}_1^2\;\; \beta_{21}\hat{G}_1^2 \cdots \beta_{c1}\hat{G}_1^2 & \beta_{11}\hat{G}_2^2\;\; \beta_{21}\hat{G}_2^2 \cdots \beta_{c1}\hat{G}_2^2\\
\cdot & & \\
\beta_{1k}\, \beta_{2k} \cdots \beta_{ck} & \beta_{1k}\hat{G}_1^k\;\; \beta_{2k}\hat{G}_1^k \cdots \beta_{ck}\hat{G}_1^k & \beta_{1k}\hat{G}_2^k\;\; \beta_{2k}\hat{G}_2^k \cdots \beta_{ck}\hat{G}_2^k
\end{bmatrix}
\end{aligned} \tag{7a}$$

The algorithm computes α_1 and α_2 such that

$$J = \sum_{j=1}^{k-1} \left| G(j) - \hat{G}_{LP}(j) \right| \tag{13}$$

is minimized. The steps of this iterative algorithm are as follows:

$$\text{Let } \boldsymbol{X} = \begin{bmatrix} \hat{G}_1(1) & \hat{G}_2(1) \\ \vdots & \\ \hat{G}_1(k-1) & \hat{G}_2(k-1) \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} G(1) \\ \vdots \\ G(k-1) \end{bmatrix}$$
$$\boldsymbol{a}^T = [\alpha_1, \ \alpha_2].$$

Initialize $\varepsilon =$ Some small positive number.

1) Initialize $\boldsymbol{U} = [u_1, u_2, \cdots, u_{k-1}]^T$, to $u_i = 0.5, i = 1, 2, \cdots, k-1$.
2) Compute $\boldsymbol{V} = [v_1, v_2, \cdots, v_{k-1}]^T = [1, 1, \cdots, 1]^T - [u_1, u_2, \cdots, u_{k-1}]^T$.
3) A diagonal weighting matrix $\boldsymbol{W}$ is found with
$$w_{ii} = \left[\frac{u_i^2 v_i^2}{u_i^2 + v_i^2} \right], \qquad i = 1, 2, \cdots, k-1.$$
4) Compute $\boldsymbol{a}^T = (\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{W} \boldsymbol{Y}$.
5) Compute error, $\boldsymbol{e} = \boldsymbol{Y} - \boldsymbol{X} \boldsymbol{a}^T$.
6) Compute vector $\boldsymbol{r}$ with
$$r_i = \left[\frac{u_i - v_i}{u_i^2 + v_i^2} \, e_i \right], \qquad i = 1, 2, \cdots, k-1.$$
7) If $(1 - \mathbf{1}^T \boldsymbol{r})/||\mathbf{e}|| > \varepsilon$, where $||\boldsymbol{e}||$ is norm of vector $\boldsymbol{e}$, stop. Otherwise go to next step.
8) Compute vector δ as, $\delta = \boldsymbol{W} \boldsymbol{e}$.
9) Update $\boldsymbol{U}$ as, $\boldsymbol{U} = \boldsymbol{U} + \beta\delta$ with
$$\beta = \frac{0.97}{\max \left[\max_i(\delta_i/u_i), \ \max_i(\delta_i/v_i) \right]}.$$
10) Return to Step 2).

Note that $\boldsymbol{a}^T$ parameters change on a daily basis similar to RLS algorithm.

H. Combination Using Adaptive Mixture of Local Experts

The last considered combination method is known as adaptive mixture of local experts, which is also referred as modular neural network (MNN) [12]. This scheme involves a number of "local experts" with each combining the two forecasts. A so-called gating network is used to learn to assign different regions of the two-dimensional data space to different local experts.

Training of the gating network and the local experts occurs simultaneously using the BP algorithm. The local experts utilized here are three-layer feedforward perceptron ANN's with linear nodes. The gating network is a two-layer ANN (no hidden node) with linear output nodes. The number of local experts and hidden nodes within each local expert are optimized for each case using the training set. These numbers range from two to ten experts and two to five hidden nodes per expert.

In this method, the two-dimensional space of $[\hat{G}_1(k), \hat{G}_2(k)]$ is effectively partitioned into different regions with each local expert (i.e., ANN combiner) taking responsibility for a different region. The outputs of the gating network are the weights assigned to the outputs of local experts. These weights tend to choose a single local expert depending on what region of the space the considered data point $[\hat{G}_1(k), \hat{G}_2(k)]$ falls in.

The combined forecast referred to as $\hat{G}_{\mathrm{MNN}}(k)$ is therefore a linear combination of $\hat{G}_1(k)$ and $\hat{G}_2(k)$ since linear nodes are used in the architecture. We investigated the use of sigmoid nodes but the performance was inferior to using linear nodes in all cases.

We also tried another variation of this combination strategy. We took the AFFF and AFLF as the "local experts." The gating network is used to combine these two forecasters in a way that each of the two forecasts is assigned to different regions of the original input space. In other word, the mixture of experts network operate on the original input space. The results were worse in all tested cases compared to operating on the two-dimensional forecast space.

I. Relative Implementation Complexity

The relative implementation difficulty of these eight combination methods is discussed in this section. In general, the main cost associated with the more complex methods is in their initial setup. During the on-line forecasting phase, there is basically no significant difference in execution speed of any of the methods.

A comparative ranking of the implementation difficulty of these methods is provided in the following with the averaging (no. 1) being the least difficult and the modular neural networks (no. 8) being the most involved.

1) averaging;
2) recursive least squares;
3) temperature space;
4) fuzzy logic;
5) linear programming using Karmarkar's algorithm;
6) adaptive functional link ANN;
7) adaptive feedforward ANN;
8) adaptive mixture of local experts.

IV. Experimental Study

The performance of the proposed system and the eight discussed combination schemes is tested using real data collected from six different LDC's. Two sets of studies are performed. In the first set, actual weather data is used in place of weather forecast so as to remove the impact of weather forecast error on the overall performance. In the second set, weather forecasts are used for those databases that include this information. In both cases, the training is performed with actual weather and testing is performed in a blind fashion, i.e., no information from testing set is used during training.

The performance index is the mean absolute percentage error (MAPE) of the forecasts. This is the measure used in industry for evaluating the performance and is computed as

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \frac{\left| \left(G(i) - \hat{G}(i) \right) \right|}{G(i)} * 100 \tag{14}$$

TABLE I
LDC INFORMATION FOR THE PERFORMANCE STUDY

LDC	TRAINING SET PERIOD-WINTER OF	TESTING SET PERIOD	HISTORY OF WEATHER FORECAST AVAILABLE ?
1	94, 95, 96	Nov. 2 -96 to March 31 -97 (150 days)	Yes
2	94, 95, 96	Nov. 2 -96 to May 31 -97 (211 days)	Yes
3	94, 95, 96	Nov. 3 -96 to March 30 -97 (148 days)	Yes
4	94, 95, 96	Nov. 3 -96 to May 31 -97 (210 days)	No
5	95,96	Nov. 3 -96 to May 31 -97 (210 days)	No
6	94, 95	Nov. 2 -95 to May 31 -96 (212 days)	No

TABLE II
MAPE'S FOR ONE-DAY-AHEAD FORECASTING USING ACTUAL WEATHER DATA. BEST RESULTS FOR EACH CASE SHOWN IN BOLD AND UNDERLINED

Method	LDC1	LDC2	LDC3	LDC4	LDC5	LDC6	Average MAPE	Average Std. Dev.
AFFF($\hat{G}_1$)	2.97	5.07	3.41	3.56	3.74	5.33	4.01	4.02
AFLF($\hat{G}_2$)	2.92	5.19	3.42	3.77	3.62	5.05	4.00	4.06
$\hat{G}_A$	2.79	4.92	3.35	3.58	3.60	5.13	3.90	3.99
$\hat{G}_{RLS}$	2.80	4.95	3.36	3.55	3.56	5.07	3.88	4.00
$\hat{G}_{FZ}$	2.70	4.87	3.31	3.58	3.53	5.06	3.84	4.00
$\hat{G}_{AFF}$	2.75	**<u>4.66</u>**	3.37	3.54	3.60	5.00	3.82	3.87
$\hat{G}_{AFL}$	**<u>2.66</u>**	4.71	3.31	3.54	3.59	**<u>4.90</u>**	**<u>3.78</u>**	**<u>3.83</u>**
$\hat{G}_{TS}$	2.75	4.86	**<u>3.28</u>**	**<u>3.49</u>**	**<u>3.46</u>**	5.01	3.81	3.88
$\hat{G}_{Lp}$	2.69	4.90	3.33	3.57	3.57	5.06	3.85	3.90
$\hat{G}_{MNN}$	2.72	5.03	3.32	3.58	3.54	5.06	3.87	4.00

where

- N total number of the test samples (i.e., total number of days);
- $G(i)$ actual sendout of the ith sample;
- $\hat{G}(i)$ corresponding sendout forecast.

Note that the proposed system performs one-day-ahead forecasting. It is possible to extend this horizon by using the system in a recursive manner. For instance, to obtain a two-day-ahead forecast $\hat{G}(k+1)$, the one-day-ahead forecast, $\hat{G}(k)$, is computed first and used in place of $G(k)$ which is a required input for generating $\hat{G}(k+1)$.

The details of the training and testing sets for all the LDC's are reported in Table I. Since the sendout prediction is more critical during the heating season (in most cases November–April), the winter period is considered in all studies.

In the first set of experiments, the performance is tested using actual weather instead of weather forecast. This means that all the required weather inputs are actual observed values. This approach is commonly used in load forecasting studies to analyze the modeling performance of the system without mixing in the effect of weather forecast errors. Table II lists the resulting MAPE's for one day ahead forecasts. The first two rows show

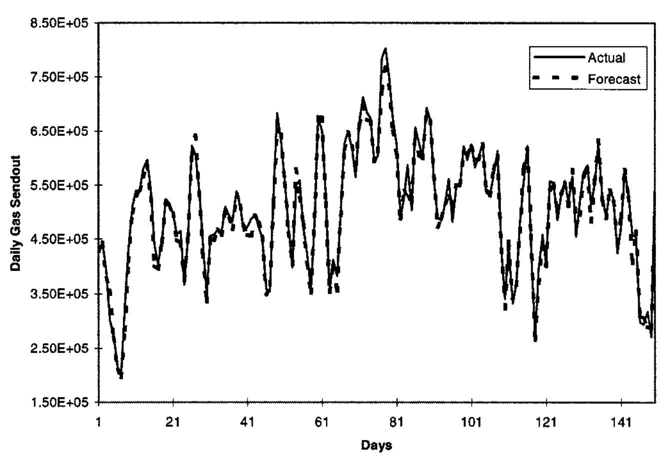

Fig. 2. Actual and one-day-ahead $\hat{G}_{\mathrm{AFL}}$ forecasts for LDC1 using actual weather data.

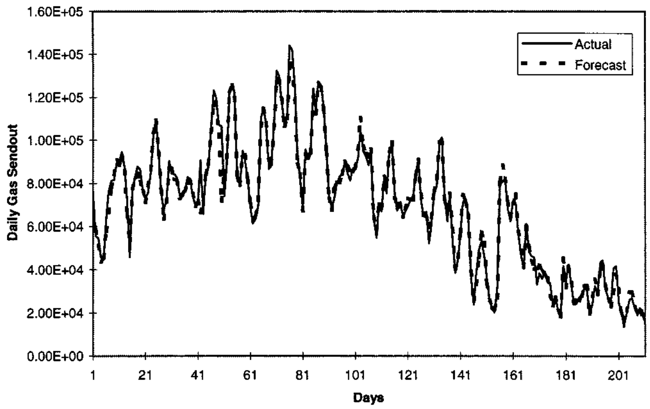

Fig. 3. Actual and one-day-ahead $\hat{G}_{\mathrm{AFF}}$ forecasts for LDC2 using actual weather data.

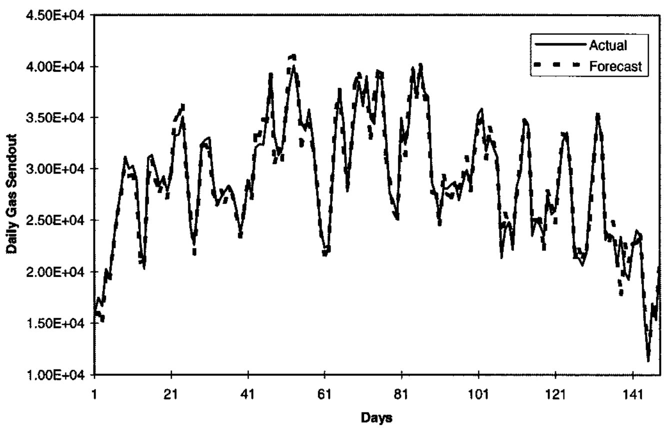

Fig. 4. Actual and one-day-ahead $\hat{G}_{TS}$ forecasts for LDC3 using actual weather data.

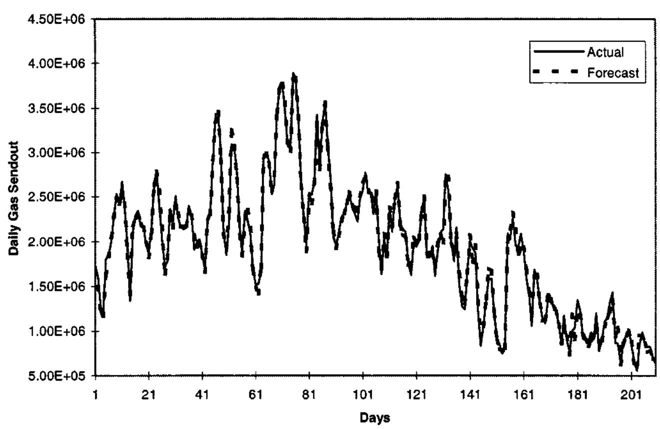

Fig. 5. Actual and one-day-ahead $\hat{G}_{TS}$ forecasts for LDC4 using actual weather data.

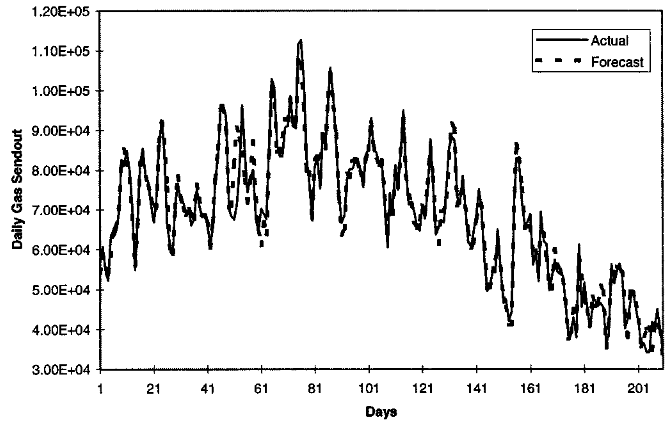

Fig. 6. Actual and one-day-ahead $\hat{G}_{TS}$ forecasts for LDC5 using actual weather data.

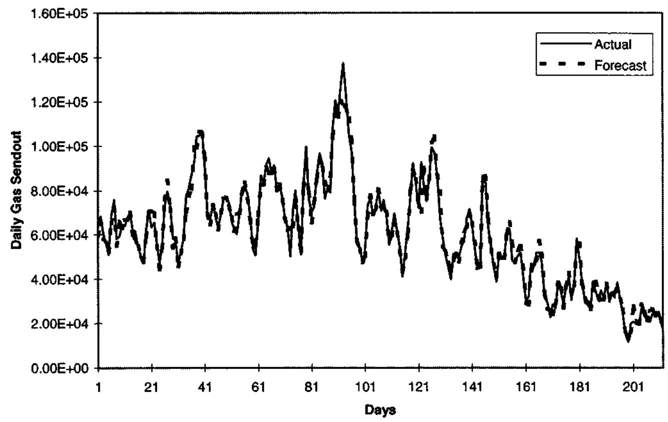

Fig. 7. Actual and one-day-ahead $\hat{G}_{\mathrm{AFL}}$ forecasts for LDC6 using actual weather data.

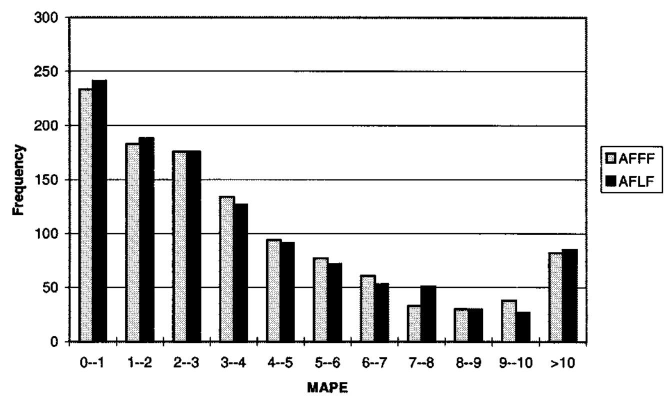

Fig. 8. The aggregated distribution of MAPE's for AFFF and AFLF using actual weather data.

the individual performance of the two forecasters. The next eight rows contain the results of the eight discussed combination approaches. The best performance for each LDC is highlighted in bold and underlined. The average MAPE for all six LDC's along with the average standard deviation of absolute error (ASDAE) are reported in the last two columns.

The best results for each LDC are shown in Figs. 2–7. In these figures the actual daily sendout data is plotted with solid line and is the corresponding forecast and is superimposed on it with dotted line. The distributions of MAPE's for the two individual forecasters as well as the AFL and AFF combiners which produce the best average results are shown in Figs. 8 and 9. The shown statistics are the MAPE's of all six LDC's quantized into 1% steps. Notice that the errors are well behaved and fall off gradually.

In the second set of studies, the performance is evaluated under real-world conditions when forecast weather is used. The

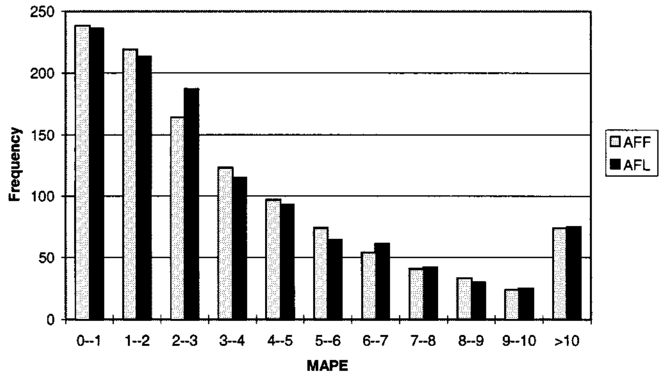

Fig. 9. The aggregated distribution of MAPE's for AFF and AFL combination using actual weather data.

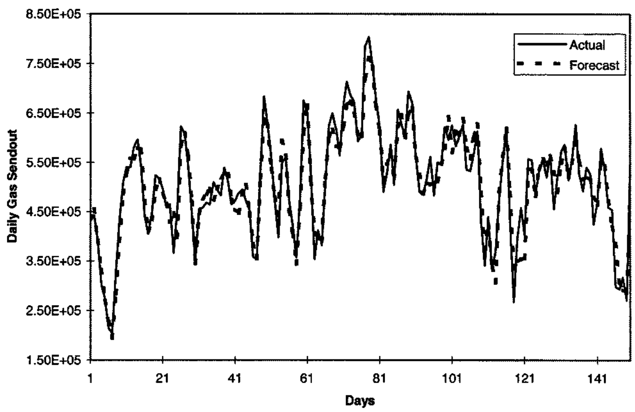

Fig. 10. Actual and one-day-ahead $\hat{G}_{\mathrm{AFL}}$ forecasts for LDC1 using forecast weather data.

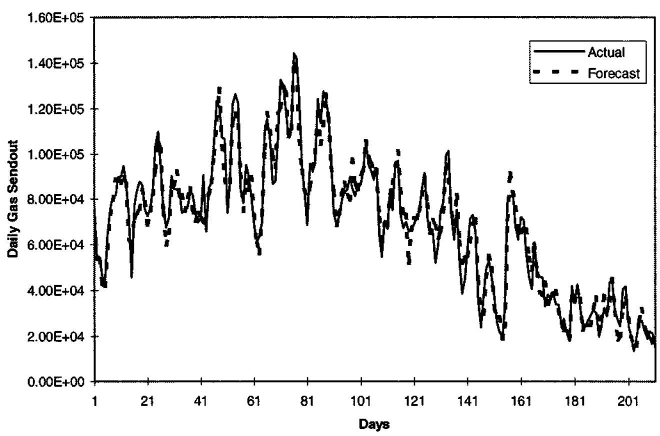

Fig. 11. Actual and one-day-ahead $\hat{G}_{\mathrm{AFL}}$ forecasts for LDC2 using forecast weather data.

forecast weather information for next day is only available for LDC1 to LDC3. The forecasts results are tabulated in Table III. Notice that the MAPE's increase due to errors in weather forecast. The weather forecast errors are reported in Table IV in terms of mean absolute error (MAE). The forecast results are also shown in graphical form in Figs. 10–12. The MAPE distributions are illustrated in Figs. 13 and 14.

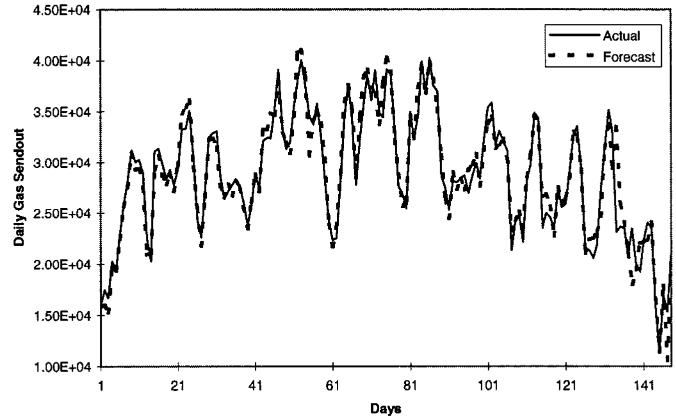

Fig. 12. Actual and one-day-ahead $\hat{G}_{\mathrm{AFL}}$ forecasts for LDC3 using forecast weather data.

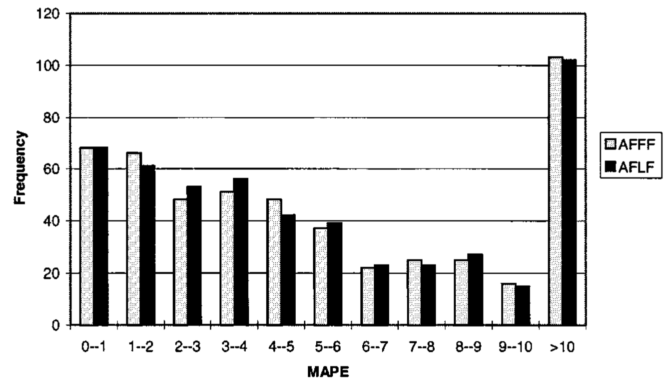

Fig. 13. The aggregated distribution of MAPE's for AFFF and AFLF using forecast weather data.

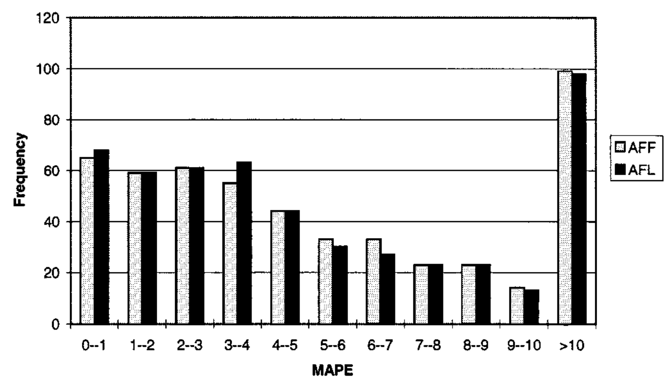

Fig. 14. The aggregated distribution of MAPE's for AFF and AFL combination using forecast weather data.

V. Analysis of Results

It can be seen that in both sets of experiments, the forecasts track the actual data quite well. The average one-day-ahead MAPE's of 3.78 and 5.89% with actual and forecast weather, respectively, are well within the acceptable range from an operational point of view. In general, every one-degree error in temperature forecast causes approximately 0.85% additional error in the sendout forecast.

Although each of the two ANN forecasters performs well, the combination of their forecasts does indeed improve the result. For the experiments with the actual weather data (Table II), the

TABLE III
MAPE'S FOR ONE-DAY-AHEAD FORECASTING USING FORECAST WEATHER DATA. BEST RESULTS FOR EACH CASE SHOWN IN BOLD AND UNDERLINED

Method	LDC1	LDC2	LDC3	Average MAPE	Average Std. Dev.
AFFF($\hat{G}_1$)	5.58	8.57	4.29	6.15	5.97
AFLF($\hat{G}_2$)	5.23	8.73	4.25	6.07	5.80
$\hat{G}_A$	5.33	8.61	4.18	6.04	5.83
$\hat{G}_{RLS}$	5.25	8.59	4.19	6.01	5.80
$\hat{G}_{FZ}$	5.24	8.43	4.18	5.95	5.87
$\hat{G}_{AFF}$	5.27	8.42	**4.15**	5.95	**5.69**
$\hat{G}_{AFL}$	**5.17**	**8.35**	**4.15**	**5.89**	5.76
$\hat{G}_{TS}$	5.31	8.51	4.19	6.00	5.94
$\hat{G}_{Lp}$	5.19	**8.35**	**4.15**	5.90	5.78
$\hat{G}_{MNN}$	5.29	8.48	4.18	6.02	5.78

AFL combiner performs the best if average MAPE criterion is considered. This combination strategy results in 6% improvement in one-day-ahead MAPE reducing it from around 4.02% for the best single forecaster to 3.78% for the AFL combination. The temperature space and AFF combiner performances are also close to the AFL combiner with the AFF producing the best ASDAE in two cases. On average, the use of a combination strategy results in about 4% improvement of ASDAE over a single forecaster. The simple averaging scheme produces the least amount of accuracy improvement.

In Figs. 8 and 9, it can be seen that the forecasting error is above 10% for about 6% of time. Such large errors are caused either by corrupted data due to problems in telemetry or the onset of sudden weather changes. The discussed adaptive scheme can usually compensate for weather-related changes if the unusual weather condition lasts beyond one day.

The AFL combiner also produces the best average MAPE when forecast weather data is used (Table III) resulting in about 3% improvement over the best single forecaster. The AFF combiner gives the best ASDAE, improving this measure by about 2%. Again simple averaging yields the least accurate combined forecast.

The overall conclusion is that the nonlinear AFL and AFF combiners perform better than other linear and/or nonlinear combination approaches. The AFL yields the best average MAPE and the AFF produces the best ASDAE.

TABLE IV
MAE'S FOR TEMPERATURE AND WIND SPEED FORECASTS

LDC	Temperature Error in Degrees F	Wind Speed Error in Miles/Hour
1	2.24	3.36
2	2.87	2.55
3	2.03	1.48

VI. CONCLUSIONS

A two stage forecasting system for prediction of daily gas sendout is developed in this paper. The nonlinear and complex relationship between temperature and past sendout with future sendout is modeled using two ANN forecasters with different topologies. The first one has a multilayer feedforward architecture whereas the second forecaster is a functional link ANN with some inputs being nonlinear functions of the considered parameters. An adaptive scheme is employed that adjust the weights of the ANN forecasters during on-line forecasting making them adaptive. Both forecasters are initially trained with the error BP algorithm.

In the second stage, the two individual forecasts are mixed together to arrive at the final forecast. Eight different combination strategies are considered, they include both linear and nonlinear approaches based on: averaging, recursive least squares, fuzzy logic, feedforward ANN, functional link ANN, a temperature space approach, linear programming, and MNN.

The system is tested on real data from six different gas utilities. Two sets of experiments with actual and forecast weather data are carried out. Based on the results it can be concluded that:

1) Each of the two ANN forecasters in the first stage performs well for gas forecasting.
2) The combination of the forecasts does indeed result in improvement of the forecast accuracy.
3) The combination schemes based on a single ANN outperform others. The adaptive functional link ANN combiner

produces the best average MAPE whereas the adaptive feedforward ANN combiner results in the most reduction in ASDAE.

Finally, it should be noted that in any implementation of the forecast combination, the extra complexity and maintenance cost should be weighted against the extra accuracy gained. This tradeoff should be significant enough for the considered application to merit the use of the respective combination approach.

REFERENCES

[1] J. M. Bates and C. W. Granger, "Combination of forecasts," *Operational Res. Quart.*, vol. 20, no. 4, pp. 451–468, 1969.

[2] R. H. Brown, L. Matin, P. Kharouf, and L. P. Piessens, "Development of artificial neural-network models to predict daily gas consumption," *A.G.A. Forecasting Rev.*, vol. 5, pp. 1–22, 1996.

[3] D. W. Bunn, "A Bayesian approach to the linear combination of forecasts," *Operational Res. Quart.*, vol. 26, pp. 325–329, 1975.

[4] D. W. Bunn and R. M. Oliver, "Combination of forecasts: A model building perspective," *Operational Res. Lett.*, vol. 8, pp. 179–184, August 1989.

[5] S. L. Chiu, "Fuzzy model identification based on cluster estimation," *J. Intell. Fuzzy Syst.*, pp. 267–278, 1994.

[6] R.T. Clemen, "Combining forecasts, A review and annotated bibliography," *Int. J. Forecasting*, vol. 5, pp. 559–583, 1989.

[7] J. P. Dickinson, "Some statistical results on the combination of forecasts," *Operational Res. Quart.*, vol. 24, pp. 253–260, 1973.

[8] J. P. Dickinson, "Some comments on the combination of forecasts," *Operational Res. Quart.*, vol. 26, pp. 205–210, 1975.

[9] F. X. Diebold and P. Pauly, "Structure change and the combination of forecasts," *J. Forecasting*, vol. 6, pp. 21–40, 1987.

[10] R. G. Donaldson and M. Kamstra, "Forecast combining with neural networks," *J. Forecasting*, vol. 15, pp. 49–61, 1996.

[11] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[12] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E Hinton, "Adaptive mixtures of local experts," *Neural Comput.*, vol. 3, pp. 79–87, 1991.

[13] N. Karmarkar, "A new polynomial time algorithm for linear programming," *Combinatorica*, vol. 4, pp. 373–395, 1984.

[14] A. Khotanzad, R. A. Rohani, T. L. Lu, A. Abaye, M. Davis, and D. J. Maratukulam, "ANNSTLF—A neural-network-based electric load forecasting system," *IEEE Trans. Neural Networks*, vol. 8, pp. 835–845, July 1997.

[15] A. Khotanzad, R. A. Rohani, and D. J. Maratukulam, "ANNSTLF—Artificial neural-network short-term load forecaster—Generation three," *IEEE Trans. Power Syst.*, vol. 13, pp. 1413–1423, 1998.

[16] S. Makridakis and R. L. Winkler, "Averages of forecasts: Some empirical results," *Management Sci.*, vol. 29, pp. 987–996, 1983.

[17] M. Mostaghimi, "Combining ranked mean value forecasts," *Eur. J. Operational Res.*, vol. 94, pp. 505–516, 1996.

[18] P. Newbold and C. W. Granger, "Experience with forecasting univariate time series and the combination of forecasts," *J. Royal Statist. Soc.*, vol. 137, pp. 131–165, 1974.

[19] T. J. Ross, *Fuzzy Logic with Engineering Applications*. New York: McGraw-Hill, 1995.

[20] MAD. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed*. Cambridge: MIT Press, 1986.

[21] S. A. Ruzinsky and E. L. Olsen, "L_1 and L_∞ minimization via a variant of Karmarkar's algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, Feb. 1989.

[22] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 1–31, February 1993.

[23] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 116–132, January 1985.

[24] R. L. Winkler and S. Makridakis, "The combination of forecasts," *J. Royal Statist. Soc.*, vol. 146, pp. 150–158, 1983.

[25] G. Zhang, B. E. Patuwo, and Y. Hu, "Forecasting with artificial neural network: The state of the art," *Int. J. Forecasting*, vol. 14, pp. 25–62, 1998.

Alireza Khotanzad (S'79–M'80) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN in 1978, 1980, and 1983, respectively.

He joined the faculty of the Department of Electrical Engineering at Southern Methodist University (SMU), Dallas, TX in 1984, where he is currently a Professor. His research interests include artificial neural networks and their applications to various forecasting problems, pattern recognition, intelligent systems, fuzzy logic, and signal and image processing.

Dr. Khotanzad serves as an Associate Editor of the *Pattern Recognition* journal. He was also an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS from 1995 to 1997. He served as the Chair of the IEEE Dallas Section in 1995 to 1996. He received the SMU School of Engineering and Applied Science Outstanding Graduate Professor Award in 1993 and 1996. He also was the recipient of the SMU Sigma Xi Outstanding Faculty Research Award in 1995. He was elected the IEEE Dallas Section Engineer of the Year in 1998.

Hassan Elragal (M'99) received the B.Sc. and M.Sc. degrees in electrical engineering from Alexandria University, Alexandria, Egypt, in 1991 and 1995, respectively. He is currently pursuing the Ph.D. degree in electrical engineering at Southern Methodist University, Dallas, TX.

He was an Instructor and an Assistant Teacher at Alexandria University from 1991 to 1995. His research interests include signal processing, fuzzy logic, and neural networks and their applications in modeling and forecasting.

Tsun-Liang Lu (S'95–M'98) was born in Yunlin Hsien, Taiwan, R.O.C., in 1963. He received the B.S. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, R.O.C. in 1986 and the M.S. and Ph.D. degrees in electrical engineering form Southern Methodist University, Dallas, TX, in 1998.

He is presently an R&D Engineer with Pattern Recognition Technologies, Inc., Dallas, TX, working on various forecasting systems. His professional interests include neural networks, genetic algorithms, and their application to forecasting and prediction.