LINEAR AND NONLINEAR MODELING AND FORECASTING OF ELECTRIC POWER LOADS

By

YONGQING ZHANG

Bachelor of Science
Nankai University
Tianjin, China
1982

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1987

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 1992

LINEAR AND NONLINEAR MODELING AND
FORECASTING OF ELECTRIC
POWER LOADS

Thesis Approved:

Martin T Hagan
Thesis Adviser

Kevin A. Teague

Marvin L. Keener

George M. Scheets

Thomas C. Collins
Dean of the Graduate College

PREFACE

The objective of the research reported in this thesis was to investigate several linear and nonlinear algorithms for short-term load forecasting (one to twenty-four hour), which is an integral part of electric power system operations and is required for economic generation scheduling, power system security assessing and maintenance scheduling. The difficulties encountered in the study are mainly due to the complex nature of power load which is characterized as a multi-period (e.g., daily, weekly) and multi-variable (e.g., weather factor, holidays, strikes) process.

During the course of this research, a temperature enhanced Hammerstein nonlinear model, which has the ability to model both multiperiodicity and nonlinearity, was found to be an extremely efficient algorithm applied to this particular problem. Another nonlinear model-the Bilinear time series model- is reported in this thesis and has been shown to have potential applicability to the short-term load forecasting problem and, remarkably, may provide a very useful model class for general applications.

At this time I would like to take this opportunity to express my sincere gratitude to many people who were a great help to me in this project and during my coursework at Oklahoma State University. In particular, I wish to thank my major adviser, Dr. Martin Hagan, for his intelligent guidance, inspiration, and invaluable aid. I cannot

imagine how this research could have been accomplished without him. I am also grateful to the other committee members, Dr. Keith Teague, Dr. George Scheets and Dr. Marvin Keener for their advisement and support. In addition, I want to extend thanks to Dr. James Baker and Dr. Carl Latino for their support during my years at this school.

Most of all, I would like to thank my family and my wife, Yisun, for their love and constant encouragement.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER I

INTRODUCTION

Power load forecasting has in the past two decades evolved into one of the most interesting research areas. This is because it relates directly to economical considerations in power systems, and also because it is such an interesting problem that involves many techniques and approaches to be investigated and developed.

The economical considerations come from the fact that load forecasting plays a key role in the planning and operation of electric utilities. Specifically, the power load forecasts are needed for

(i) economic generation scheduling. This determines the most economic commitment of generation sources and economic energy interchange with other utilities.

(ii) power system security assessing. This information permits the dispatchers to detect many vulnerable situations in advance and, therefore, operate the power system securely.

(iii) maintenance scheduling and fuel allocation.

It is obvious that the accuracy of load forecasts has significant effects on operating and production costs. For instance, forecast error due to underprediction of load results in the shortage of the necessary reserves which, in turn, raise costs because of the use of the expensive peaking units. Forecast error due to overprediction of load, on the other hand, produces higher costs also because of an

unnecessary increase in reserves by the start-up of too many units.

For the purpose of economically efficient operation, load forecasting can be further classified according to its forecasting range. Very short term forecasts with lead times from a few minutes to an hour are needed in the on line automatic generation control (AGC) and economic dispatch functions. Short term forecasts with lead times from an hour to a few weeks are needed for economic generation scheduling, system security assessing and short term maintenance scheduling. Intermediate/long term forecasts with lead times longer than a month are needed for fuel, hydro and maintenance scheduling, new utility constrution consideration and the determination of prices. This paper is concerned with the short term load forecast with lead times from one hour to one day.

A number of different forecasting algorithms have been applied to the STLF problem. The existing load forecasting techniques include multiple linear regression [1], general exponential smoothing [1], spectral expansion [2], stochastic time series [3], state space [4], Hammerstein nonlinear [5], artificial neural network (ANN) [6] and expert system based [7] algorithms. The multiple linear regression approach finds functional relationships between weather variables and the system load and expresses the relation by a linear or piecewise linear function. The general exponential smoothing method uses a fitting function and takes the form $Z_t=\Sigma\ \beta_i f_i(t)+a_t$, where Z_t is the power load at time t, which is computed by the sum of a finite number of fitting functions $f_i(t)$, usually sinusoids with a period of 24 hours. β_i and a_t represent slowly time-varying constants and a white noise, respectively. These two algorithms are computationally simple,

but may not give sufficiently accurate results because they do not accurately represent the intrinsic characteristics of power load. In fact, they produce an averaged result. The spectral expansion model has basically the same form as that for the general exponential smoothing except that the time functions $f_i(.)$ here represent the eigenfunctions corresponding to the autocorrelation function of the load sequences (after removal of trends and periodicities). It is reported [8] that the spectral expansion model has the advantage over the general exponential smoothing model in that the time functions chosen can more closely approximate its autocorrelation function and, therefore, the representation of loads can be obtained with greater precision than with arbitrarily selected time functions. One problem with the spectral expansion model is that its practical advantage is not as sound as its theoretical one and, as a result, only a few utilities seem to use such a method. Reference [8] has categorized the above three models into the class of time-of-day models.

Another class of models is categorized as dynamic models, which include two basic types, stochastic time series models and state space models. In dynamic models, the load is considered to be not only a function of the time of day, but also of its most recent history as well as the weather and noise components. The stochastic time series approaches, normally using autoregressive-integrated-moving average (ARIMA) and transfer function (TF) models discussed by Box and Jenkins [9], have been shown [3] to be well suited to the load forecasting application because of their ability to incorporate dynamic, weather and random effects. The identification

process of the stochastic time series usually needs more intensive computation than those of the time-of-day models. However, considering that the frequency of parameter identification is low (once a day) whenever the model structure is set up, this computational burden is not a real problem.

The state space model can actually include the time series model and vice versa [10]. One of the most attractive features of the state space method is that the on-line load prediction can be done by employing the recursive property of the Kalman filter. The limitation of this approach is that the model has to be known prior to using the Kalman filter. Unfortunately, the identification of the state space model is usually difficult.

So far the models we discussed are limited to linear types. Generally speaking, linear models are used for their simplicity and their ability to approximate most real problems. In order to be able to better understand the STLF problem, the new direction is, certainly, the study of nonlinear models and, hopefully, these models will provide us with even better performance.

Recently several researchers have developed some STLF nonlinear models. Hagan [3] proposed a nonlinear model to the STLF by a simple nonlinear extension to the transfer function model. Later, Lu et al. [5] applied the Hammerstein nonlinear model structure [11] to the STLF with lead times of one hour. Another type of nonlinear model, using an artificial neural network (ANN), was presented by Park et al. [6]. In this paper, another nonlinear model, called the bilinear model, is proposed for the STLF application. In addition, the Hammerstein nonlinear model is further investigated and extended

for 24-hour-ahead forecasts.

Since temperature is very important to the STLF, a temperature enhanced model, which modifies the Hammerstein model by adjusting parameters through the k^{th} step ahead forecasting errors or the sum of 1 to k step ahead forecasting errors instead of residuals, which are one step ahead forecasting errors, is presented with the goal to make full use of temperature information and, thus, further improve forecasting performance.

Following this introduction, the paper is organized as follows: the linear time series models, which include the ARIMA and TF models, are reviewed in chapers II-III; the general nonlinear model of the Volterra expansion is dicussed in chapter IV; a test for linearity is outlined in chapter IV also; some special nonlinear models, which include the bilinear time series model, Hammerstein nonlinear model and temperature enhanced model, are presented in chapters V-VII; and, finally, the conclusions are given in chapter VIII.

CHAPTER II

LINEAR ARMA MODELS

In this chapter, some basic concepts, which will be used throughout the paper, are presented first. Then each of the model building steps in the general system identification scheme is discussed and applied to the power load process. These distinct steps include:

1. Select a class of models which will be most useful for our load forecasting purpose;

2. Identify the order of the model selected by step 1 (i.e., preliminary identification);

3. Estimate the parameters of the system and, thus, set up the tentatively identified model and, finally

4. Check the tentative model according to some criterion to decide its goodness of fit. If good, then it is ready to use for forecasting. Otherwise, the process is repeated from step 2.

Some Basic Concepts

Let us denote the power load series as Z_t, Z_{t-1}, Z_{t-2}, . . ., where t, t-1, t-2, . . . represent integer values in time, say hours. Then it is obvious that $\{Z_t\}$ can be viewed as a time series and, thus, all the time series analysis and modeling techniques can be equally applied

to the load series $\{Z_t\}$. In the following discussions, we treat $\{Z_t\}$ simply as a general real valued time series.

Autocovariance and Autocorrelation Functions

The autocovariance of a time series $\{Z_t\}$ is defined as

$$\mathrm{Cov}(Z_t, Z_{t+k}) = E[(Z_t-\mu)(Z_{t+k}-\mu)] = R_t(k), \quad k = 0,\pm1,\pm2, \qquad (II.1)$$

where E[.] denotes the expectation, $\mu=E[Z_t]$ is the mean, and $R_t(k)$ represents the covariance between observations k lags apart. If the mean is constant and the covariance is only a function of lag k, then the series is said to be stationary, otherwise it is nonstationary. Now, let us assume that $\{Z_t\}$ is a stationary process, then the autocorrelation function of lag k is defined by

$$\rho(k) = R(k)/R(0) \qquad (II.2)$$

where $R(0)=\mathrm{Cov}(Z_t,Z_t)=E[(Z_t-\mu)^2]=\sigma^2$ is called variance of $\{Z_t\}$. The physical meanings of the mean and covariance are obvious. The mean is, roughly speaking, the average value of the series, the variance is a measurement of the data spread about the mean and the covariance function describes inherent correlation between observations k lags apart. The natural estimates of μ and R(k) are obtained by

$$\hat{\mu} = \frac{1}{N}\sum_{t=1}^{N} Z_t, \qquad \hat{R}(k) = \frac{1}{N}\sum_{t=1}^{N-|k|}(Z_t - \hat{\mu})(Z_{t+|k|} - \hat{\mu})$$
$$k = 0,\pm1,\pm2, \ldots, \pm(N-1) \qquad (II.3)$$

where N is the number of observations.

Gaussian Processes

$\{Z_t\}$ is called a stationary, white Gaussian process if it satisfies

$$P(Z_1,Z_2,\ldots,Z_N)=\frac{1}{(2\pi)^{N/2}\sigma^N}\exp\left(-\sum_{j=1}^{N}(Z_j-\mu)^2/2\sigma^2\right) \tag{II.4}$$

In other words, the set of random variables $\{Z_1, Z_2, \ldots, Z_N\}$ has a multivariate normal distribution. From (II.4), the stationary, white Gaussian processes are completely specified by two parameters, the mean μ and the variance σ^2. We will see later that the Gaussian processes play an important role in the model parameter identification.

White Noise Processes

A process $\{a_t, t=0,1,2, \ldots\}$ is called a white noise process if $\{a_t\}$ consists merely of a sequence of independent random variables and satisfies

$$E[a_t]=\mu_a,$$

$$\mathrm{Cov}(a_t,a_{t+k})=\begin{cases}\sigma_a^2, & k=0\\ 0, & k\neq 0\end{cases} \tag{II.5}$$

or equivalently,

$$\rho_a(k)=\begin{cases}1, & k=0\\ 0, & k\neq 0\end{cases} \tag{II.6}$$

Figure 1 illustrates this property with lags up to 6.

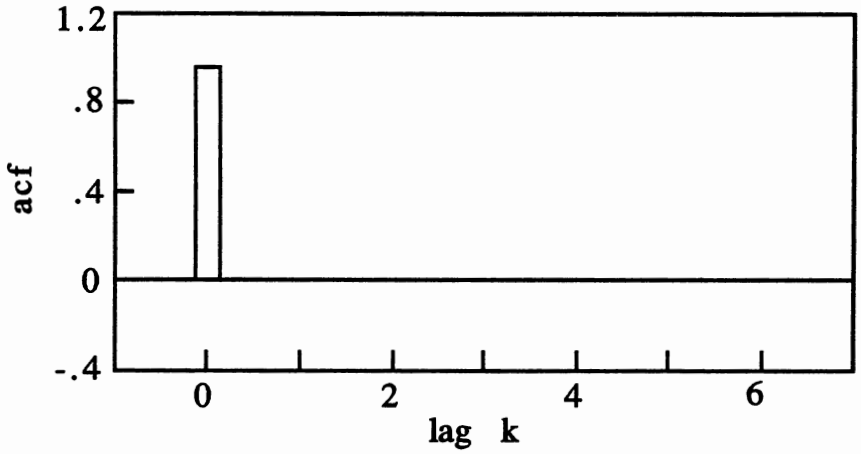

Figure 1. Autocorrelation Function of a White Noise Process

Model Building and Forecasting

The model building problem is solved, basically, through two routes. One of them is through mathematical modeling based on physical laws. For example, the trajectories of the planets may be modeled by Newton's laws. The other route is through system identification, which is based on observations from the system. This paper is concerned with the later case, and various models are built based on system identification techniques. The forecasting problem accordingly is stated as follows: we are given (N+1) past observations $\{Z_t, Z_{t-1}, Z_{t-2}, \ldots, Z_{t-N}\}$, and wish to use the information contained in the data set to predict m steps ahead. This predictor is denoted $Z_t(m)$. Mathematically, we try to compute $Z_t(m)$ by

$$Z_t(m) = \theta(Z_t, Z_{t-1}, Z_{t-2}, \ldots, Z_{t-N}) \qquad \text{(II.7)}$$

where θ is a function of the given observations. It is shown [12] that the "best" predictor $Z_t(m)$ of Z_{t+m}, under the "criterion" of the minimum "mean square error" (i.e., min $E[\{Z_t(m)-Z_{t+m}\}^2]$), is the conditional mean

$$Z_t(m) = E[Z_{t+m}|Z_t, Z_{t-1}, \ldots, Z_{t-N}] \tag{II.8}$$

AR, MA and ARMA Models

In time series analysis, a model for $\{Z_t, t=0,\pm1,\pm2, \ldots\}$ means a functional relationship described by

$$f(\ldots Z_{t-2}, Z_{t-1}, Z_t, Z_{t+1}, Z_{t+2}, \ldots) = a_t \tag{II.9}$$

where f(.) is the function to be found and a_t is white noise. In general, it is impossible to determine a specific function, knowing only a finite set of data on $\{Z_t\}$. However, if we consider the class of linear models, (II.9) can be written as

$$\sum_{i=-\infty}^{\infty} \phi_i Z_{t-i} = a_t \tag{II.10}$$

where $\{\phi_i\}$ is a sequence of constants. In the load forecasts, since Z_t depends only on past values, (II.10) is simplified by setting $\phi_i=0$ for $i<0$, and rewritten as

$$\sum_{i=0}^{\infty} \phi_i Z_{t-i} = a_t \tag{II.11}$$

(II.11) is called the general linear process. Note that the assumption, that f(.) is a linear function, is actually applied to the whole of

standard time series.

In practice, we cannot hope to estimate an infinite number of parameters. Instead, a finite order polynomial is used. In this case, $\phi_i=0$ for $i>p$, thus (II.11) becomes

$$\sum_{i=0}^{p}\phi_i Z_{t-i}=a_t \tag{II.12}$$

where ϕ_0 is assumed to be 1 and p is the order and determined by the identification techniques discussed in next section. By introducing the following operators

$$BZ_t = Z_{t-1},\ B^mZ_t = Z_{t-m}$$
$$\phi(B) = 1 - \phi_1B - \phi_2B^2 - \ldots - \phi_pB^p \tag{II.13}$$

(II.12) can be conveniently written in the form

$$\phi(B)Z_t = a_t \tag{II.14}$$

(II.14) is called the autoregressive model of order p (i.e., the AR(p) model). The autoregressive model forms an important class of the fundamental time series models.

Another fundamental class of time series models is called the moving average model. In the moving average model of order q (i.e. the MA(q) model), the current value of the process is expressed linearly in terms of the current and q previous random shocks.

$$Z_t = a_t - \theta_1a_{t-1} - \ldots - \theta_qa_{t-q} \tag{II.15}$$

where $\theta_1, \theta_2, \ldots, \theta_q$ are constants and $\{a_t\}$ is a white noise process. Equivalently, we may write (II.15) as

$$Z_t = \theta(B)a_t \tag{II.16}$$

where

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q \tag{II.17}$$

Since either the AR(p) or the MA(q) model can be used to describe a given stochastic process, it is natural to combine them together to form a more general model: the mixed autoregressive-moving average model of order (p,q) (i.e., the ARMA(p,q) model).

$$\phi(B)Z_t = \theta(B)a_t \tag{II.18}$$

where $\phi(B)$ and $\theta(B)$ are given by (II.13) and (II.17), respectively. Note that the ARMA(p,q) model includes both AR(p) and MA(q) models as special cases and possesses a remarkably wide range of applicability.

The time series $\{Z_t\}$ described by an AR(p) , MA(q) or ARMA(p,q) process will be a stationary process if the roots of $\phi(B)$ lie outside the unit circle. For some non-stationary processes, a stationary process W_t can be achieved by differencing the original process Z_t,

$$W_t = \nabla^d Z_t \tag{II.19}$$

where the operator ∇ is defined as follows

$$\begin{aligned} \nabla Z_t &= (1-B)Z_t \\ \nabla_k^d Z_t &= (1-B^k)^d Z_t \end{aligned} \tag{II.20}$$

Since the resulting process W_t is stationary, it can be modeled by an ARMA(p,q) process

$$\phi(B)W_t = \theta(B)a_t \qquad \text{(II.21)}$$

Substituting W_t by (II.19), we obtain the autoregressive-integrated-moving average model ARIMA(p,d,q)

$$\phi(B)\nabla^d Z_t = \theta(B)a_t \qquad \text{(II.22)}$$

Finally, it is noted that many time series present some periodicities or "seasonal" behaviors. This is especially true for our power load because, as we will see later, the load curve contains a daily periodicity (e.g. the load at 10 A.M. Tuesday is related to the load at 10 A.M. Monday) as well as weekly periodicity (Sundays are not like Mondays). Combining all these factors together, a general time series model of the power load can be expressed as

$$\phi_p(B)\phi'_{p'}(B^{24})\phi''_{p''}(B^{168})\nabla^d\nabla_{24}^D\nabla_{168}^{D'}Z_t = \theta_q(B)\theta'_{q'}(B^{24})\theta''_{q''}(B^{168})a_t \qquad \text{(II.23)}$$

(II.23) forms one of our fundamental models, namely "seasonal" ARIMA model, for the STLF application. Since the "seasonal" ARIMA model is the basic model used by Box and Jenkins (1970) in their work on forecasting, it is often called the Box and Jenkins time series model.

Identification and Estimation of ARMA Models

In the last section, we chose the "seasonal" ARIMA model as one of our fundamental classes of models for the STLF purpose. Now

we proceed discussing the techniques used to identify the order and estimate the parameters of the model.

Preliminary Identification

In preliminary identification, the techniques used for the order determination rely on the analysis of the autocorrelation function (acf) and partial autocorrelation function (pacf) of the given process. For the purpose of illustrating how these methods are being systematically applied to the problem of preliminary identification, we investigate the specific properties of the acf and the pacf associated with each particular model (e.g., AR, MA).

For an AR(p) process, the autocorrelation function is found by (II.14), (II.1) and (II.2), i.e.

$$\begin{aligned} R(k) &= E[Z_t Z_{t-k}] \\ &= \phi_1 E[Z_{t-1} Z_{t-k}] + \phi_2 E[Z_{t-2} Z_{t-k}] + \ldots + \phi_p E[Z_{t-p} Z_{t-k}] + E[a_t Z_{t-k}] \\ &= \phi_1 R(k-1) + \phi_2 R(k-2) + \ldots + \phi_p R(k-p) \end{aligned} \qquad \text{(II.24)}$$

where Z_t is assumed to be zero mean, otherwise Z_t should be replaced by $Z_t-\mu$. The expectation $E[a_t Z_{t-k}]$ varnishes because Z_{t-k} is uncorrelated with a_t for $k > 0$. Dividing both sides of (II.24) by R(0), we obtain the autocorrelation function

$$\rho(k) = \phi_1 \rho(k-1) + \phi_2 \rho(k-2) + \ldots + \phi_p \rho(k-p) \qquad \text{(II.25)}$$

As is seen from (II.25), the autocorrelation function follows a certain pattern in nature. Taking AR(1) as an example, (II.25) becomes

$$\rho(k) = \phi_1 \rho(k-1) \qquad |\phi_1| < 1, k > 0 \qquad \text{(II.26)}$$

It is easy to show that the solution of (II.26) is

$$\rho(k) = \phi_1^k \qquad |\phi_1| < 1, k > 0 \qquad \text{(II.27)}$$

Obviously, $\rho(k)$ decreases exponentially as the lag k increases as shown in Figure 2. For general AR(p) processes, $\rho(k)$ is a combination of damped sinusoids and exponentials.

The partial autocorrelation function is another important tool in preliminary identification. It is defined with the use of the AR model. Consider trying to fit a sequence of AR(p=1,2,3,. . .) models to a stochastic process, which is generated by AR(p'). Let ϕ_{kj} denote the j^{th} coefficient in one of the fitted AR models of order k and, then, ϕ_{kk} is the last coefficient. Clearly, ϕ_{kk} will be nonzero for k less than or equal to p' and zero for k greater than p'. Figure 2 shows the characteristics of the pacf for an AR(1) process.

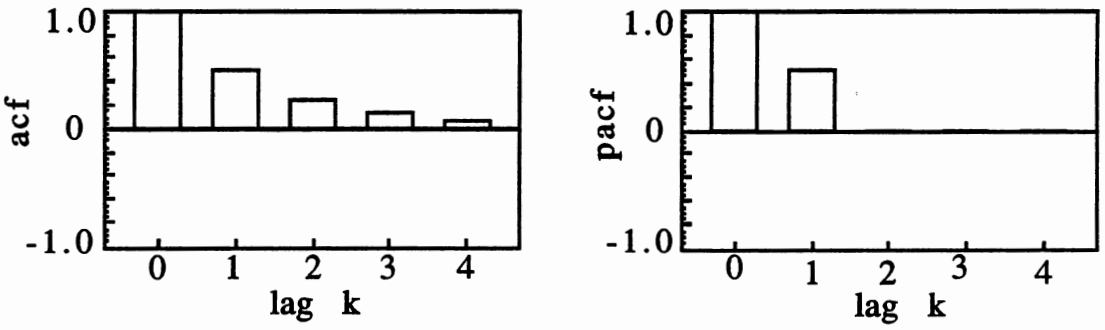

Figure 2. ACF and PACF of $Z_t = 0.5Z_{t-1} + a_t$

For an MA(q) process, the autocorrelation function, by using a derivation procedure similar to that above, is found to be identically zero for lags k greater than q. In other words, the acf of an MA(q) process will be cut off after lag q. The partial autocorrelation function of the MA(q) process, unlike the AR(p) process, consists of a combination of damped sinusoids and exponential. This result may be explained by the fact that an MA(q) process can be equally written as

$$\theta^{-1}(B)Z_t = a_t \qquad \text{(II.28)}$$

where $\theta(B)$ is assumed to be invertible, with its inverse denoted by $\theta^{-1}(q)$. In general, $\theta^{-1}(B)$ is a polynomial with infinite order, and, thus, (II.28) represents an AR process with an infinite order. Following the discussions of the pacf for the AR process, the pacf of an MA process has infinite components. Figure 3 illustrates the acf and the pacf graphs for an MA(1) process.

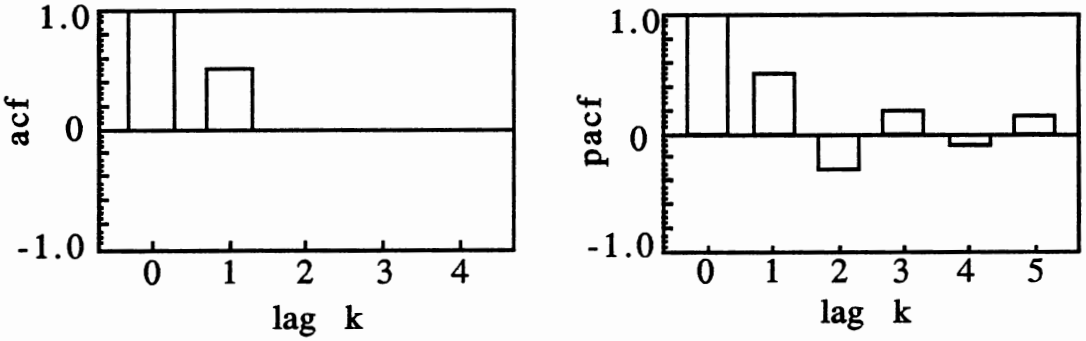

Figure 3. ACF and PACF of $Z_t = a_t + 0.8a_{t-1}$

From previous discussions, we have seen that the AR process and the MA process follow different patterns of the acf and pacf, and thus can be distinguished easily from one another. In addition, the cut-off property shown from either the acf or the pacf may be used to decide the order of the model.

For an ARMA(p,q) process, and its variants the ARIMA and "seasonal" ARIMA processes, the identification procedures may include (1) differencing the original sequence with an appropriate order until a stationary process is obtained. This means that no heavy periodic component/slow decaying in the acf remain; and (2) applying the identification results from the analysis of the AR and MA models to the acf and pacf obtained in (1), and then determining the orders p and q. The reader is referred to [9] for further details about using the acf and pacf techniques for order identification of an ARMA process.

Parameter Estimation

Once the "order" of the model is determined, the next step is to estimate each parameter in the model. There exist many algorithms for parameter estimation. Investigation of these algorithms is beyond our scope. The method we use in this research is the maximum likelihood algorithm. In this algorithm, the noise series, a_t, is assumed to follow a jointly normal distribution with zero-mean and variance σ^2, and the following log-likelihood function is maximized

$$l(\phi,\theta,\sigma) = f(\phi,\theta) - n\ln\sigma - \frac{S(\phi,\theta)}{2\sigma^2}$$

where $$S(\phi,\theta) = \sum_{j=1}^{n} a_j^2 \qquad \text{(II.29)}$$

The term $f(\phi,\theta)$ is a function of coefficients ϕ and θ, and is small in comparison with the sum of squares function $S(\phi,\theta)$ when the effective number of observations, n, is large. Thus, the parameters which minimize $S(\phi,\theta)$ are usually used as close approximations to maximum likelihood estimates. A nonlinear optimization routine, which minimizes $S(\phi,\theta)$, is outlined in [9].

Diagnostic Checks and Forecasts

Now that the model order has been determined and the parameters have been estimated, the last step in the identification process is to check if the identified model is accurate. Two common ways to determine the goodness of fit of the model are presented in this section. These are Bartlett's test and the portmanteau lack of fit test.

Recall that in earlier sections what we tried to do for a given set of observations $\{Z_t\}$ was to find a function which accurately describes the relationship between the values, and whose errors are white noise (see (II.9)). Thus, checking the whiteness of the residuals a_t is commonly used as the criterion to evaluate the goodness of fit of the model. If a_t is white, the model is considered as adequate. Otherwise, we need to reevaluate the orders and reestimate the parameters. One way of checking the whiteness of a_t is by checking the autocorrelation function for a_t. If there is only one spike at t=0 with magnitude 1 and all the other autocorrelations equal to zero,

then a_t is white noise. Since, in practice, only the approximations $r_k(a)$ of the acf for a_t are available, the boundary defined by Bartlett's approximation

$$\mathrm{Var}[r_k] \cong \frac{1}{n}\{1 + 2 \sum_{v=1}^{q} r_v^2\} \qquad k > q \tag{II.30}$$

is used to make the decision, where n is the effective number of observations after differencing on Z_t. If quite a few of the autocorrelations lie outside the range of $\pm 2\sqrt{\mathrm{Var}[r_k]}$, then it probably means that the model is not adequate.

Another way of checking a_t's whiteness is through a portmanteau lack of fit test. This test checks the Q values computed by

$$Q = n \sum_{k=1}^{K} r_k^2(\hat{a}) \tag{II.31}$$

which, under the null hypothesis that a_t is white noise process, is approximately distributed as χ^2(K-p-q) on (K-p-q) degrees of freedom, where p, q represent the orders of the ARMA(p,q) model. If the average value of Q is high compared with a table of percentage points (say, 5% point) of the χ^2(K-p-q) distribution, then the fitted model is poor.

Up to this point, we have completed the identification process. In other words, we have obtained a fitted model whose residual sequence $\{a_t\}$ is white noise and, therefore, we are ready to use the model for forecasts.

As noted earlier, the "best" m step ahead predictor, ($Z_t(m)$), under the "criterion" of the minimum mean square error, is

$$Z_t(m) = E[Z_{t+m}|Z_t, Z_{t-1}, \ldots, Z_{t-N}] \qquad \text{(II.32)}$$

which is the conditional expectation of Z_{t+m} given all the history of values of Z's up to the current time t. It can be shown that the one step ahead forecasting error is the white noise process a_t.

As an example of using (II.32) for forecasts, consider an ARIMA(1,1,1) model

$$(1-0.7B)(1-B)Z_t = (1-0.4B)a_t \qquad \text{(II.33)}$$

$$(1-1.7B+0.7B^2)Z_t = (1-0.4B)a_t$$

$$Z_t = 1.7Z_{t-1}-0.7Z_{t-2}+a_t-0.4a_{t-1} \qquad \text{(II.34)}$$

For one step ahead forecast, replace t by t+1 and then take the conditional expectation on both sides

$$Z_t(1) = 1.7Z_t - 0.7Z_{t-1} + a_t(1) - 0.4a_t$$
$$= 1.7Z_t - 0.7Z_{t-1} - 0.4a_t$$

where $a_t(1) = E[a_{t+1}|a_t, a_{t-1}, \ldots] = E[a_{t+1}] = 0$ since a_t is an independently identically distributed white noise. Similarly, $a_t(j)=0$ for $j>0$.

For the two step ahead forecast, replace t by t+2 in (II.34) and then take the conditional expectation on both sides

$$Z_t(2) = 1.7Z_t(1) - 0.7Z_t + a_t(2) - 0.4a_t(1)$$
$$= 1.7Z_t(1) - 0.7Z_t$$

It is easy to show that m step ahead forecast is

$$Z_t(m) = 1.7Z_t(m-1) - 0.7Z_t(m-2) \quad \text{for } m > 2$$

In general, for a given model we may find the m step ahead forecasts by the following procedure:

(1) expand the given model until an explicit expression for Z_t is obtained (e.g. (II.34));

(2) replace t by t+m and then take conditional expectation;

(3) apply the following properties to the equation obtained by (2)

$$\begin{aligned} &E[Z_{t+j}|Z_t, Z_{t-1}, \ldots] = Z_t(j) && j = 1, 2, \ldots \\ &E[Z_{t-j}| Z_t, Z_{t-1}, \ldots] = Z_{t-j} && j = 0,1,2, \ldots \\ &E[a_{t+j}| a_t, a_{t-1}, \ldots] = 0 && j = 1,2, \ldots \\ &E[a_{t-j} | a_t, a_{t-1}, \ldots] = a_{t-j} && j = 0,1,2, \ldots \end{aligned} \qquad \text{(II.35)}$$

Simulation Results

In this section, the preceding ARIMA model (II.23) is fitted for the electric power load series, which is a collection of hourly load readings for 1983 from a moderately sized system in the Tulsa area. Figure 4 and Figure 5 show the plots of the typical power load series for a period of two weeks during summer and spring seasons, respectively.

As observed from these figures, each season has a different pattern of behavior. Therefore, four separate ARIMA models, corresponding to the summer, fall, winter and spring seasons, are considered. In addition, the plots also clearly indicate the daily and weekly periodicities of the power loads. In the following presentation, we discuss utilizing those identification procedures outlined previously to set up appropriate models for power loads.

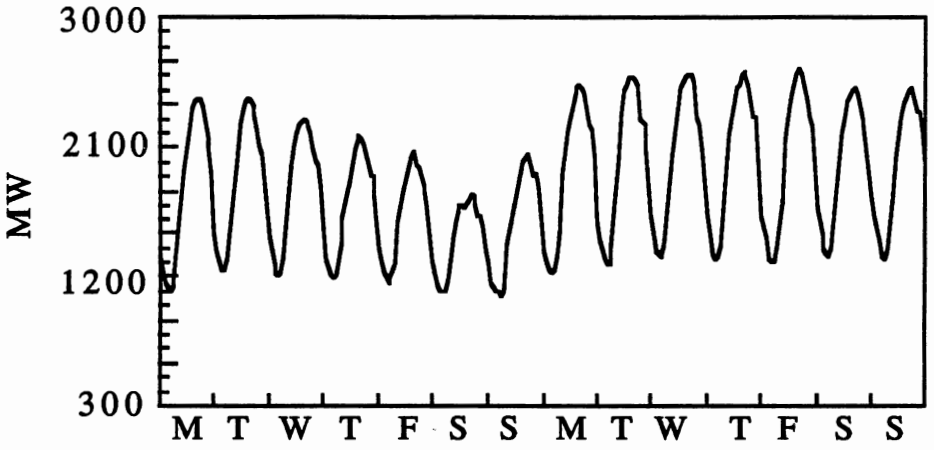

Figure 4. Summer Power Load: July 11 - July 24, 1983

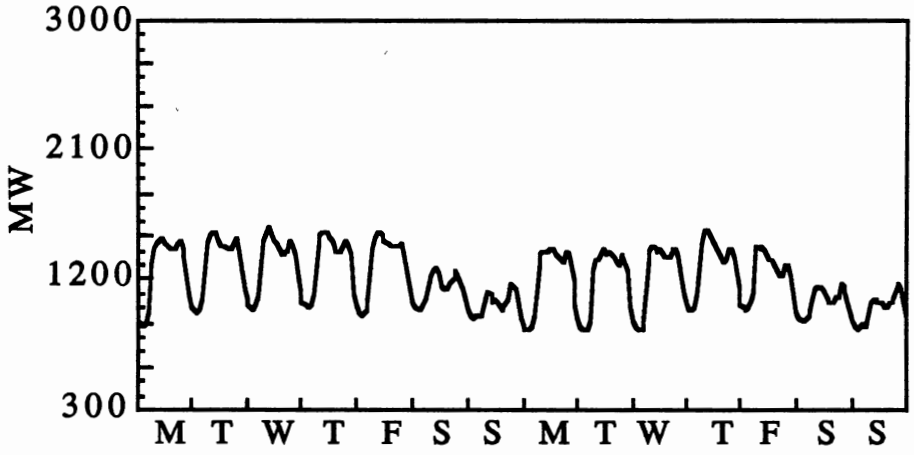

Figure 5. Spring Power Load: April 4 - April 17, 1983

Consider the summer data for a period of four weeks (July 11 - August 7, 1983), we first examine its acf to determine if the original process is stationary. As shown in Figure 6, the oscillatory/slow-decaying pattern indicates that the given process is nonstationary. In order to obtain a stationary process and, thus, be able to fit the process by an ARMA model, several differencing schemes are

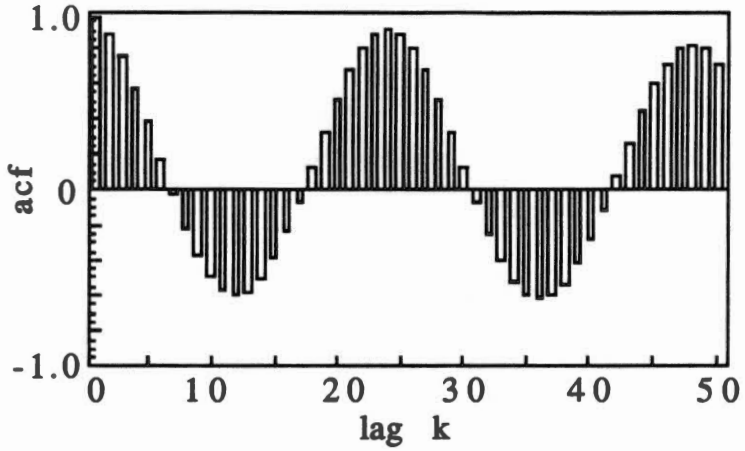

Figure 6. ACF of the Summer Load Series

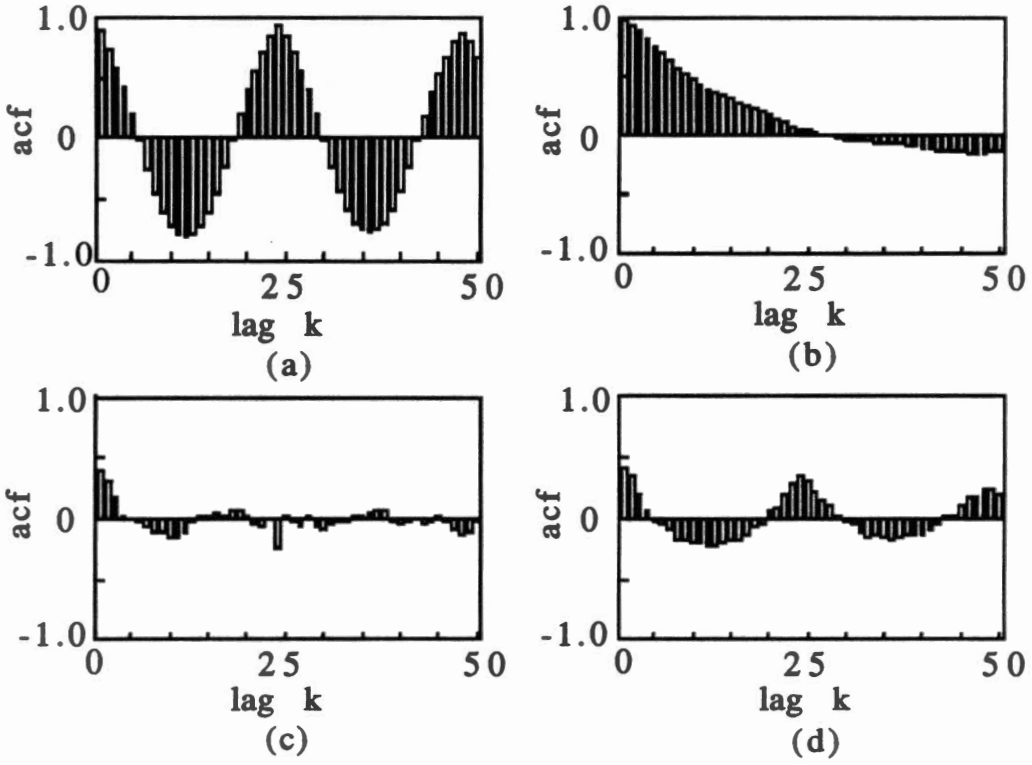

Figure 7. ACF of the Summer Load Series Differenced w.r.t.
a) hours only b) days only
c) hours and days d) hours and weeks

considered (see Figure 7). As suggested from Figure 7, the stationary process may be obtained by either a differencing scheme of $\nabla_1\nabla_{24}$ or $\nabla_1\nabla_{168}$. In the following discussions, we use $W_t=\nabla_1\nabla_{24} Z_t$ as the resulting stationary process.

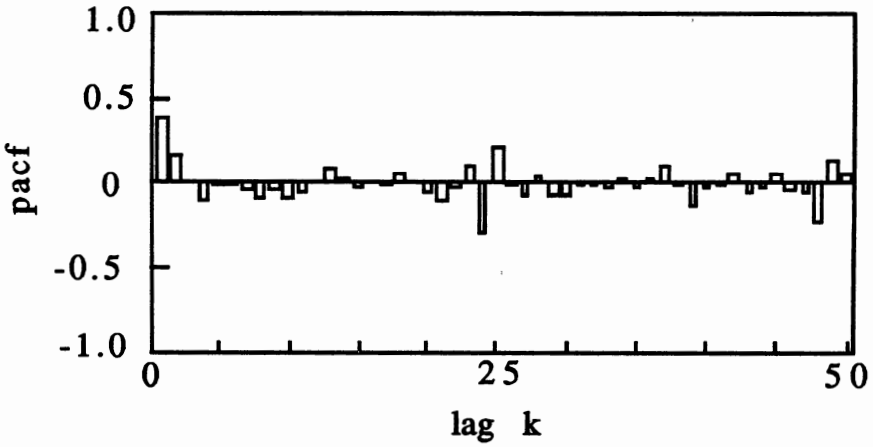

Figure 8. PACF of the Summer Load Series Differenced $\nabla_1\nabla_{24}$

The next step in developing the model is to identify the model orders (p,q), (p',q') and (p",q"). These values may be estimated by examining the acf and pacf of W_t. As shown in Figure 7c and Figure 8, the exponential decaying pattern in the acf and the large spikes at lags 1 and 2 in the pacf suggest p=2 and q=0 (i.e., an hourly component is AR(2)).

Now in order to find the daily order (p',q'), those acf and pacf values at lags of multiples of 24 are pulled out and reploted in

Figure 9. This time the exponential decaying pattern shows up in the pacf while the large spikes at lags 24 and 48 in the acf. This leads to p'=0 and q'=2 (i.e., a daily component is MA(2)). We might proceed further for the weekly component since the large spike at lag 168 is likely related to it. At this moment, however, we simply assume that the tentative summer model has the form

$$(1-\phi_1 B-\phi_2 B^2)\nabla_1\nabla_{24}Z_t=(1-\theta_1 B^{24}-\theta_2 B^{48})a_t \qquad \text{(II.36)}$$

or conveniently denoted by $(2,1,0)_1 x(0,1,2)_{24}$.

The parameters in (II.36) were estimated using the maximum likelihood routine [18], and then the model's residuals, a_t, were checked for goodness of fit. Figure 10 illustrates the plots of the acf of a_t. To see the details of its pattern, the spikes at lag of 0, whose magnitude is 1, were omitted.

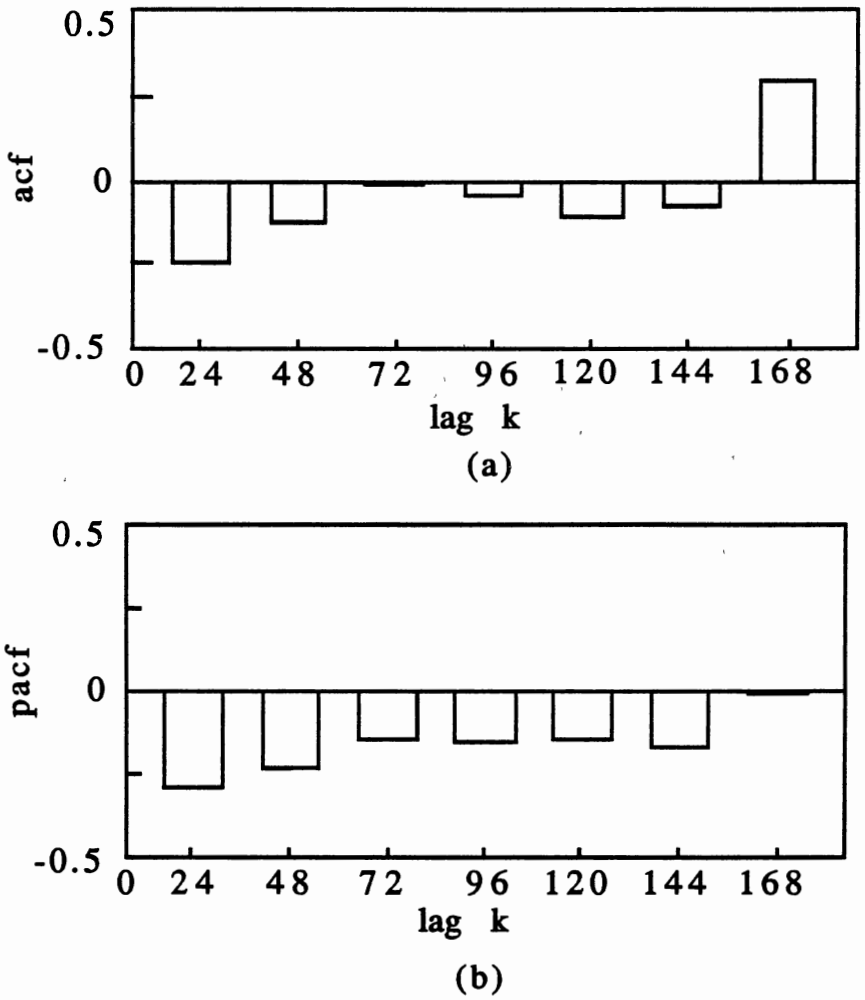

Figure 9. (a) Daily Components of Figure 7c
(b) Daily Components of Figure 8

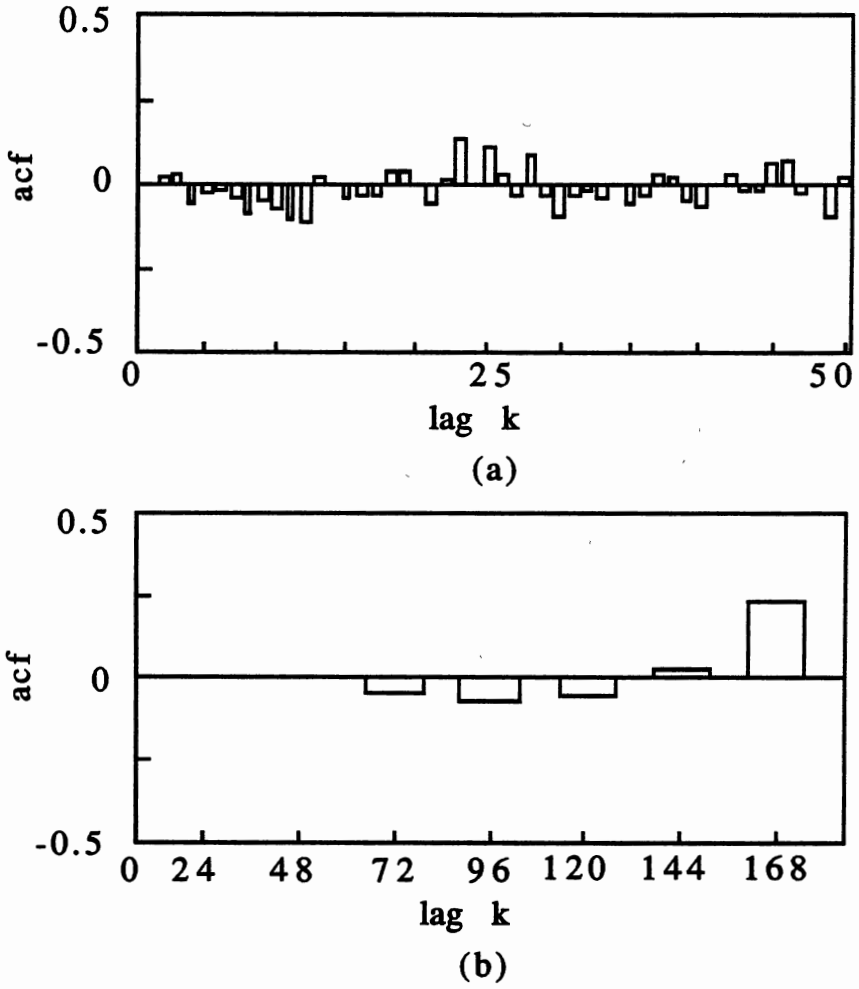

Figure 10. ACF of the Residuals in Summer Model (II.36)

The large spike at lag 168 suggests that the model (II.36) may need to have a weekly component added to it. This matches our earlier suspicion. By adding a weekly AR(1) component to (II.36) and recomputing the parameters, we obtain the following model of the form $(2,1,0)_1 x (0,1,2)_{24} x (1,0,0)_{168}$:

$$(1-.26B-.18B^2)(1-.39B^{168})\nabla_1\nabla_{24}Z_t = (1-.73B^{24}-.09B^{48})a_t \quad \text{(II.37)}$$

A whiteness test for a_t from (II.37) gives no indication of model inadequacy and this completes the identification procedure.

Finally, we are ready to use (II.37) for our STLF. Using [18], 1 to 24 hour ahead load forecasts were made for a three week period (July 29 - August 17, 1983). Table I shows a typical daily forecasts. Figure 11 illustrates the average percent errors for each day for the same time span. The total average percent error is found to be 4.49. Figure 12 shows the graph of forecasts against the actual loads. We see from this graph that a poor forecast happened on the second sunday. Many sources may result in such a situation (e.g., a sudden temperature change, shut down of nearby industrial plants, holidays, etc.). It seems that the performance would be improved if temperature could be explicitly involved in the model, since temperature changes are inevitable.

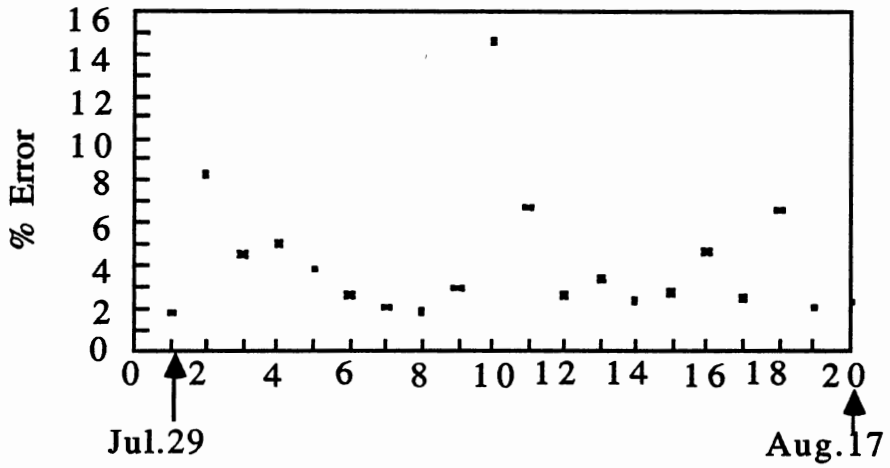

Figure 11. Average Percent Forecasting Errors Using (II.37)

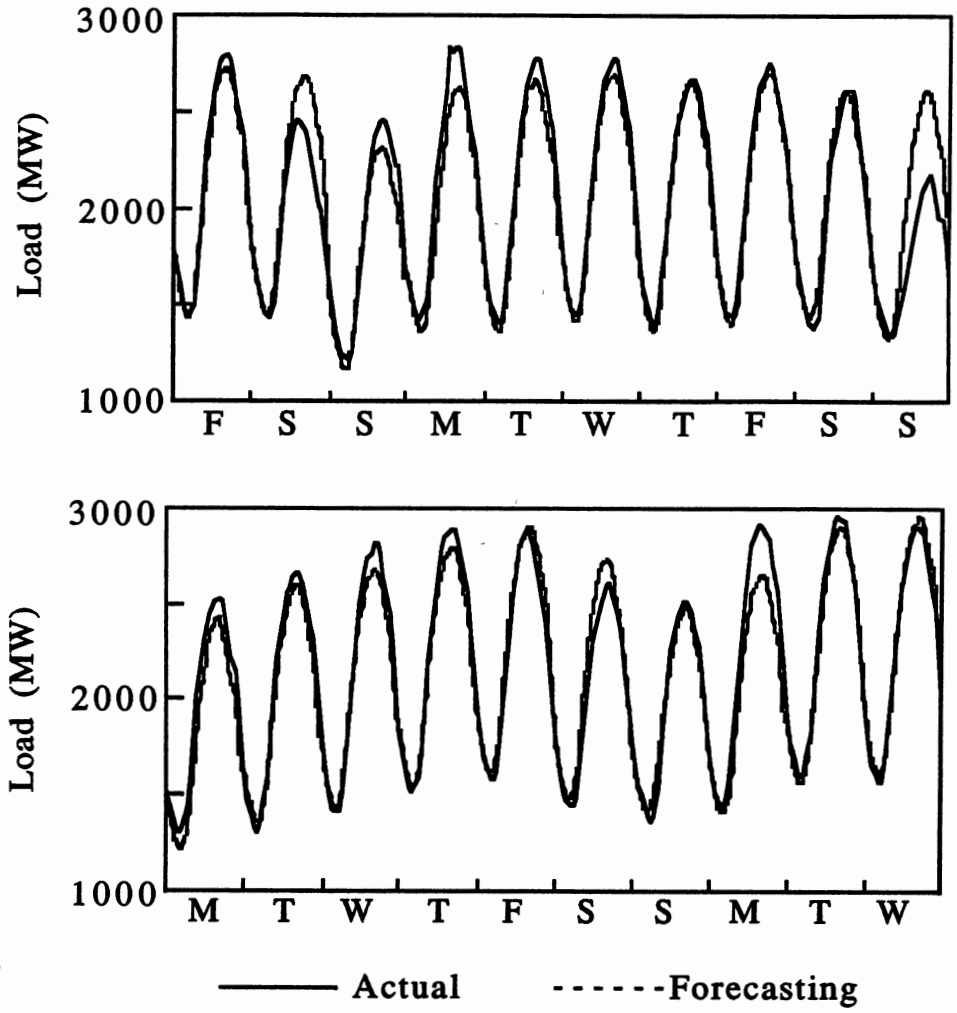

Figure 12. Summer Load Forecasts Using Model (II.37)
July 29-August 17, 1983

TABLE I

FORECAST FOR A TYPICAL SUMMER DAY

Time (hour)	Load (MW)	Forecast (MW)	Error% (Abs.)	Time (hour)	Load (MW)	Forecast (MW)	Error% (Abs.)
1	1547	1539	0.53	13	2168	2120	2.23
2	1452	1421	2.12	14	2292	2236	2.45
3	1356	1323	2.46	15	2379	2288	3.84
4	1300	1256	3.40	16	2441	2301	5.72
5	1252	1191	4.87	17	2454	2317	5.58
6	1229	1164	5.25	18	2446	2297	6.08
7	1220	1178	3.41	19	2395	2246	6.20
8	1255	1285	2.42	20	2321	2147	7.48
9	1439	1482	3.01	21	2229	2054	7.87
10	1658	1660	0.11	22	2224	2019	9.23
11	1828	1810	0.99	23	2064	1861	9.84
12	1991	1976	0.76	24	1860	1647	11.40

TABLE II

ARIMA MODELS

Season	Load period for model development	Model
Summer	July 11-Aug.7,1983	$(1-.26B-.18B^{2})(1-.3B^{168})\nabla_1\nabla_{24}Z_t = (1-.73B^{24}-.09B^{48})a_t$
Fall	Oct.3-Oct.30,1983	$(1+.044B)(1-.55B^{168})\nabla_1\nabla_{24}Z_t = (1-.65B^{24}-.19B^{48})a_t$
Winter	Jan.3-Jan.30,1983	$(1-.146B)(1-.69B^{168})\nabla_1\nabla_{24}Z_t = (1-.61B^{24}-.12B^{48})a_t$
Spring	April 4-May 1,1983	$(1-.21B)(1-.55B^{168})\nabla_1\nabla_{24}Z_t = (1-.53B^{24}-.16B^{48})a_t$

TABLE III

AVERAGE ABSOLUTE PERCENT ERRORS
USING ARIMA MODELS

Season	Forecasting Period	Ave. Abs. Error%
Summer	July 29-Aug.17,1983	4.17
Fall	Oct.21-Nov.9,1983	4.68
Winter	Jan.21-Feb.9,1983	3.85
Spring	April 22-May 11,1983	5.24

The ARIMA models for fall, winter and spring seasons were similarly developed and summarized in Table II. Table III shows the average absolute percent errors for each model. It is clear from these results that the ARIMA models are well suited for power load processes. Consequently, the ARIMA model with the systematic model development procedure forms one of the most popular approaches to the short term load forecasting.

One of the drawbacks of the ARIMA model is the inability to explicitly include weather variables such as temperature. In next chapter, we will consider a linear transfer function model, which allows temperature as an explicit input, and we will show that the model indeed improves the forecasting performance.

CHAPTER III

LINEAR TRANSFER FUNCTION MODELS

In this chapter, we consider another type of linear model, called the transfer function model, for the purpose of short term load forecasting. This model allows temperature as an explicit variable and, thus, may be expected to more exactly model the power load than ARIMA models.

The Transfer Functions

The transfer function may be described as a linear filter which connects an input X_t to an output Z_t as depicted in Figure 13.

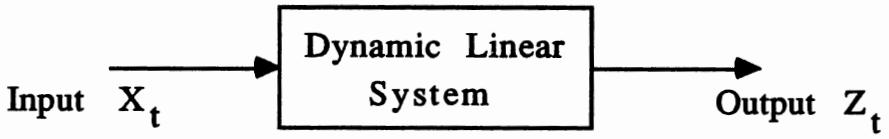

Figure 13. Transfer Function Diagram

Suppose that the linear system is causal and initially relaxed, then the transfer function model can be expressed as

$$Z_t = V(B)X_t \qquad \text{(III.1)}$$

where

$$V(B) = \sum_{i=0}^{\infty} v_i B^i \qquad \text{(III.2)}$$

is called the transfer function of the linear system.

In chapter II, the ARMA model was given by

$$\phi(B)Z_t = \theta(B)a_t$$

This equation can be rewritten as

$$Z_t = \phi^{-1}(B)\theta(B)a_t \qquad \text{(III.3)}$$

where $\phi^{-1}(B)$ is the inverse of $\phi(B)$ and assumed to exist. Comparing (III.3) with (III.1), we see that (III.3) may be viewed as a special transfer function model with white noise a_t as "input".

In general, the transfer function V(B) can be best represented by

$$V(B) = \delta^{-1}(B)\omega(B)B^b \qquad \text{(III.4)}$$

where $\delta(B) = (1 - \delta_1 B - \ldots - \delta_r B^r)$

$$\omega(B) = (\omega_0 - \omega_1 B - \ldots - \omega_s B^s) \qquad \text{(III.5)}$$

and b is a delay parameter representing a possible delayed response of the system. Considering the fact that a disturbance noise always exists in the working system, a general transfer function model may be expressed as

$$Z_t = \delta^{-1}(B)\omega(B)X_{t-b} + N_t$$

or $$Z_t = V(B)X_t + N_t \qquad \text{(III.6)}$$

where N_t is a disturbance noise and assumed to be independent from the input X_t. Equation (III.6) represents the class of transfer function models for bivariate stochastic process (X_t,Z_t). Note that the input X_t, output Z_t and the disturbance noise N_t are all assumed to be zero mean stationary processes. In practice, suitable differences on them may be needed to obtain stationary processes.

As noted earlier, a problem with ARIMA models is that weather variables such as temperature cannot be included explicitly. The transfer function model corrects this by taking temperature as its input and, in the meanwhile, allows N_t to be modeled by an ARIMA process. Therefore, the transfer function models include past load history as well as temperature explicitly and hence might be more exact than the ARIMA models for the power loads.

Identification and Estimation of TF Models

In this section, we first present two data analysis tools, namely the impulse response function and the cross correlation function. Then we discuss how they are applied to identify the order (r,s) of a transfer function. The importance of the prewhitening technique is discussed also.

Impulse Response Function

Consider the transfer function model (III.1). If the input X_t were an impulse defined by

$$X_t = \begin{cases} 1, & t = 0 \\ 0, & t \neq 0 \end{cases} \qquad \text{(III.7)}$$

then the output sequence Z_t, Z_{t-1}, Z_{t-1} ... would simply be the weights v_0, v_1, v_2, ... associated with the impulse input sequence X_t, X_{t-1}, X_{t-2}, ···. These weights are called the impulse response function, forming a basic analysis tool in the transfer function model building.

By combining (III.2) with (III.4), we have

$$\delta(B)\left(\sum_{i=0}^{\infty} v_i B^i\right) = \omega(B)B^b \qquad \text{(III.8)}$$

or equivalently

$$(1 - \delta_1 B - \delta_2 B^2 - \ldots - \delta_r B^r)(v_0 + v_1 B + v_2 B^2 + \ldots) = (\omega_0 - \omega_1 B - \omega_2 B^2 - \ldots - \omega_s B^s)B^b \qquad \text{(III.9)}$$

Equating coefficients of B results in

$$\begin{aligned} v_j &= 0 && j < b \\ v_j &= \delta_1 v_{j-1} + \delta_2 v_{j-2} + \ldots + \delta_r v_{j-r} + \omega_0 && j = b, \\ v_j &= \delta_1 v_{j-1} + \delta_2 v_{j-2} + \ldots + \delta_r v_{j-r} - \omega_{j-b} && j = b+1, b+2, \ldots, b+s \\ v_j &= \delta_1 v_{j-1} + \delta_2 v_{j-2} + \ldots + \delta_r v_{j-r} && j > b+s \end{aligned} \qquad \text{(III.10)}$$

Obviously, these relations may be used to identify the order (r,s) and the delay parameter b if all v's are known. More specifically, the parameter b may be determined by the pattern that $v_j=0$ for $j<b$; the orders r and s may be identified by the fact that the values v_b, v_{b+1}, ... ,v_{b+s-r} follow no fixed pattern and v_j's for $j \geq b+s-r+1$ have the pattern set forth by the r^{th} order difference equation. Note that

(III.10) gives initial estimates of the δ's and ω's also by solving the equations simultaneously.

Cross Correlation Function

In chapter II, we noted that the autocorrelation function played an important role in identifying the univariate stochastic process such as the ARMA process. Similarly, the cross correlation function is extensively used in the bivariate stochastic process (X_t,Z_t) and defined as

$$\rho_{xz}(k) = \gamma_{xz}(k)/\sigma_x\sigma_z \qquad k = 0, \pm1, \pm2, \ldots \qquad \text{(III.11)}$$

where $$\gamma_{xz}(k) = E[(X_t - \mu_x)(Z_{t+k} - \mu_z)], \quad k = 0, \pm1, \pm2, \ldots \qquad \text{(III.12)}$$

Note that in general $\gamma_{xz}(k)\neq\gamma_{zx}(k)$ and $\gamma_{xz}(k)\neq\gamma_{xz}(-k)$. This is different from the autocovariance and autocorrelation functions that are symmetric. Note also that (X_t,Z_t) are assumed to be stationary. Similar to dealing with some non-stationary univariate processes, appropriate differences on (X_t,Z_t) may be needed to obtain the corresponding stationary process.

The natural estimates of $\gamma_{xz}(k)$ and $\rho_{xz}(k)$ are

$$\hat{\gamma}_{xz}(k) = \frac{1}{N}\sum_{t=1}^{N-|k|}(x_t - \hat{\mu}_x)(Z_{t+|k|} - \hat{\mu}_z) \qquad k = 0,\pm1,\pm2,\ldots$$

$$\hat{\rho}_{xz}(k) = \frac{\hat{\gamma}_{xz}(k)}{s_x s_z} \qquad k = 0,\pm1,\pm2,\ldots \qquad \text{(III.13)}$$

where s_x and s_z are estimates for σ_x and σ_z, respectively. It will be seen later that $\rho_{xz}(k)$ is directly involved in calculating the estimates

of the impulse response weights v_j, which in turn may be used to determine the order (r,s) and the initial estimates of parameters of a transfer function.

Determination of (r,s)

One of the key steps toward the preliminary identification of transfer functions is to prewhiten the input X_t so that a white noise input to the system is obtained. The basic idea behind "prewhitening" is that if the input were white noise then the problem of estimating the impulse response function would be simple. This point will be clear as we proceed.

In the case of power loads, the temperature is considered as the input X_t to the linear dynamic system. Since it is a real valued stochastic process, an ARMA model may be employed

$$\phi_x(B)X_t = \theta_x(B)\alpha_t \qquad \text{(III.14)}$$

Assuming $\theta_x(B)$ to be invertible, (III.14) may be rewritten as

$$\alpha_t = \theta_x^{-1}(B)\phi_x(B)X_t \qquad \text{(III.15)}$$

Clearly, a white noise process α_t is obtained by "prewhitening" X_t through an ARMA transformation. If this transformation is applied to the general transfer function model

$$Z_t = V(B)X_t + N_t \qquad \text{(III.16)}$$

then a new, prewhitened, model is created

$$\beta_t = V(B)\alpha_t + \theta_x^{-1}(B)\phi_x(B)N_t$$

or $$\beta_t = V(B)\alpha_t + \varepsilon_t \qquad \text{(III.17)}$$

where

$$\varepsilon_t = \theta_x^{-1}(B)\phi_x(B)N_t$$

$$\beta_t = \theta_x^{-1}(B)\phi_x(B)Z_t \qquad \text{(III.18)}$$

Unlike α_t, β_t and ε_t are not necessarily white.

With the prewhitened transfer function model (III.17), the impulse response function V(B) can be easily calculated. By multiplying both sides of (III.17) by α_{t-k} and taking expectations, we have

$$E[\alpha_{t-k}\beta_t] = \gamma_{\alpha\beta}(k) = v_k\sigma_\alpha^2 \qquad \text{(III.19)}$$

where, as before, we assumed that the input α_t is statistically independent from the disturbance noise ε_t and, thus, $E[\alpha_{t-k}\varepsilon_t]=0$. From (III.19),

$$v_k = \frac{\gamma_{\alpha\beta}(k)}{\sigma_\alpha^2} = \frac{\rho_{\alpha\beta}(k)\sigma_\beta}{\sigma_\alpha}$$

and $$\hat{v}_k = \frac{\hat{\rho}_{\alpha\beta}(k)s_\beta}{s_\alpha}, \qquad k = 0, \pm1, \pm2, \ldots \qquad \text{(III.20)}$$

In summary, the procedure to identify the order parameter (r,s) of a transfer function is as follows:

(1) Identify the ARIMA model for the input X_t;

(2) Compute the prewhitened input α_t and the similarly transformed output β_t;

(3) Estimate the variances of α_t and β_t and the cross correlation function between them;

(4) Calculate the impulse response using (III.20), and

(5) Estimate the order (r,s).

Identification of the noise model

Once we obtain the estimates of the impulse response v(k), the noise N_t can be estimated using (III.6)

$$\hat{N}_t = Z_t - \hat{V}(B)X_t \qquad \text{(III.21)}$$

From these noise values, a tentative ARMA(p,q) model for N_t may be determined using, again, the identification techniques presented in chapter II. Therefore, an initial transfer function would be

$$Z_t = \delta^{-1}(B)\omega(B)X_{t-b} + \phi^{-1}(B)\theta(B)a_t \qquad \text{(III.22)}$$

where $N_t=\phi^{-1}(B)\theta(B)a_t$ and a_t is white noise. The parameters in (III.22) may be estimated by employing a nonlinear optimization routine [9], which minimizes the sum of squares of a_t in (III.22).

Diagnostic Checks and Forecasts

In the case of the transfer function model, we usually examine (1) the autocorrelation function of the residuals a_t from the fitted model, and (2) the cross correlation function between the prewhitened input α_t and the residual a_t in order to decide the

accuracy of the model. The common ways to check the model adequacy by (1) are outlined in chapter II, namely Bartlett's boundary test and Q value test. Unlike the linear ARMA model, checking the autocorrelation function only is not sufficient. For instance, if the acf shows some correlation patterns, it will be impossible to decide whether it is due to the incorrect transfer function or due to the incorrect noise model since either part may cause the autocorrelation patterns. However, if we check both (1) and (2) and find marked autocorrelation patterns but no evidence of the existence of any cross correlation pattern, then the noise model must be incorrect. On the other hand, the transfer function must be incorrect and needs to be adjusted if there exists any cross correlation pattern [9]. Therefore, checking both (1) and (2) provides us with more information about the system and thus allows us to adjust the model properly.

Similar to the Q value test for whiteness of a_t process, the S value defined by

$$S = n \sum_{k=0}^{K} r_k^2(\alpha\beta) \tag{III.23}$$

is, under the hypothesis that the model is a good fit, approximately distributed as χ^2(K-r-s) on (K-r-s) degrees of freedom, where $r_k(\alpha\beta)$ denotes the approximation of the cross correlation function between α_t and β_t. A high average value of S, comparing with the corresponding χ^2 percentage point, indicates the inaccuracy of the transfer function.

Once the model is checked and selected, it is ready for

forecasting. The general procedures for forecasting using the transfer function model are similar to those outlined in chapter II, where the forecasts were made based on the ARIMA models.

Simulation Results

For the purpose of comparison, in this section we use the same time periods which were used as for the ARIMA models. The temperature readings were provided by the weather bureau for Tulsa.

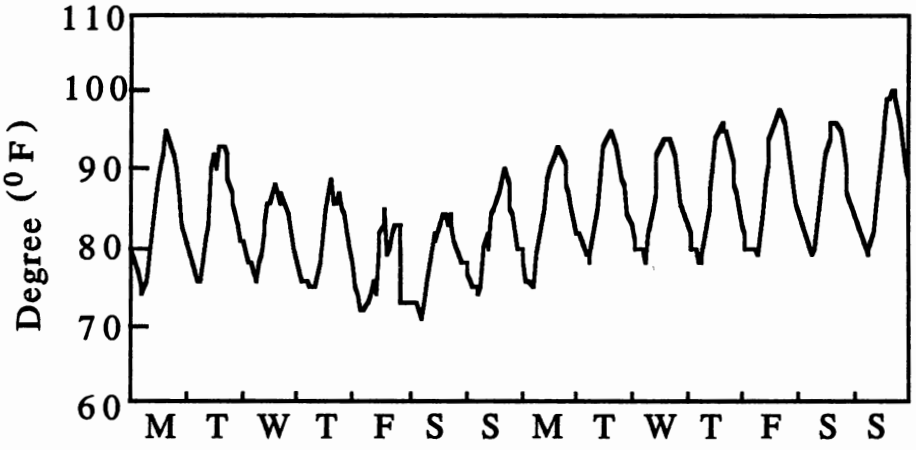

Figure 14. Summer Temperature: July 11 - July 24, 1983

Now consider, again, the summer data for the same four week period (July 11 - August 7, 1983), the first step in identifying the transfer function is to identify the ARIMA model for the temperature

input series X_t. Figure 14 is the plot of the first two weeks of data. The acf of the series indicated its nonstationarity. A stationary process was obtained after a difference of $\nabla_1\nabla_{24}$ as shown in Figure 15.

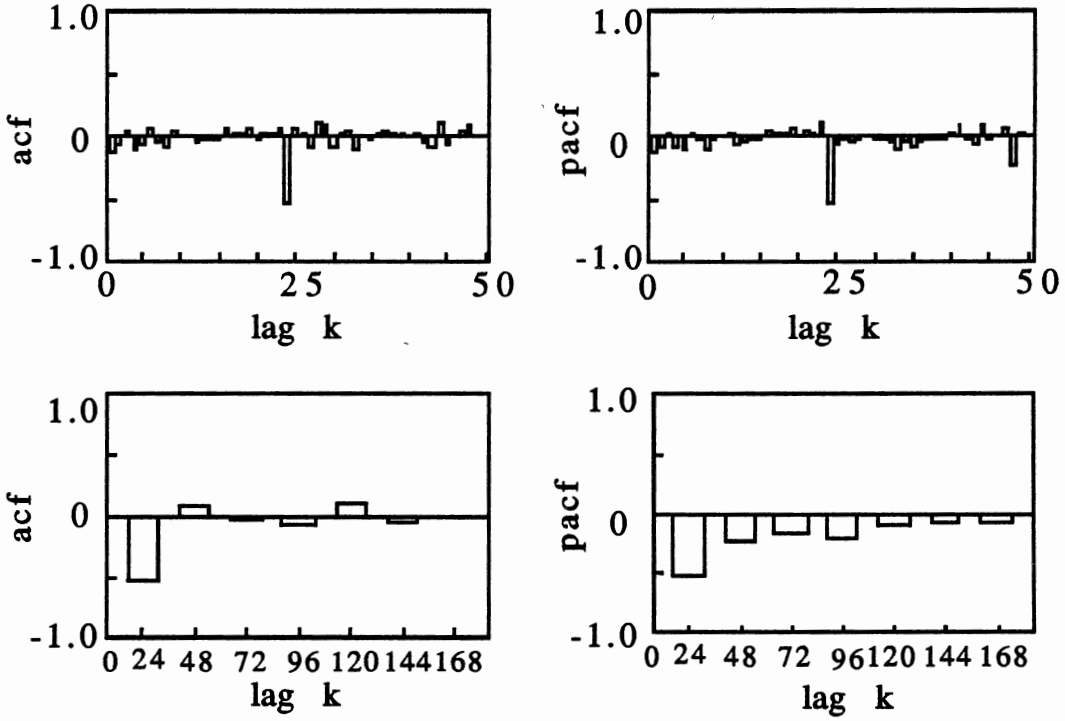

Figure 15. ACF and PACF of Summer Temperature Series Differenced $\nabla_1\nabla_{24}$

From these graphs, a process with an hourly AR(1) and a daily MA(1) was identified and, using the maximum likelihood routine [18], the $ARIMA_x$ model

$$(1+.098B)\nabla_1\nabla_{24}X_t = (1-.909B^{24})\alpha_t \qquad \text{(III.24)}$$

was found to be a good fit to the temperature series $\{X_t\}$. Using (III.24), the prewhitened input α_t may be computed by

$$\alpha_t = (1-.909B^{24})^{-1}(1+.098B)\nabla_1\nabla_{24}X_t \qquad \text{(III.25)}$$

Applying this transformation to the general transfer function model (III.6), we obtain

$$\beta_t = V(B)\alpha_t + \varepsilon_t \qquad \text{(III.26)}$$

where $\beta_t = (1-.909B^{24})^{-1}(1+.098B)\nabla_1\nabla_{24}Z_t$

$\varepsilon_t = (1-.909B^{24})^{-1}(1+.098B)\nabla_1\nabla_{24}N_t$

Once we obtain the sequences $\{\alpha_t\}$ and $\{\beta_t\}$, the variances of α_t, β_t and the cross-correlation function $\rho_{\alpha\beta}$ may be estimated using (II.3) and (III.13), respectively. These calculations allow us to find the estimates of the impulse response function, which in turn may be used to compute the noise series from (III.6). Theoretically, the impulse response weights may be used to identify the order (r,s) of the transfer function. However, in our power load application, the impulse response turns out to be too noisy to be helpful for the order determination (see Figure 16). Fortunately, we have had good success by simply setting r=2 and s=1.

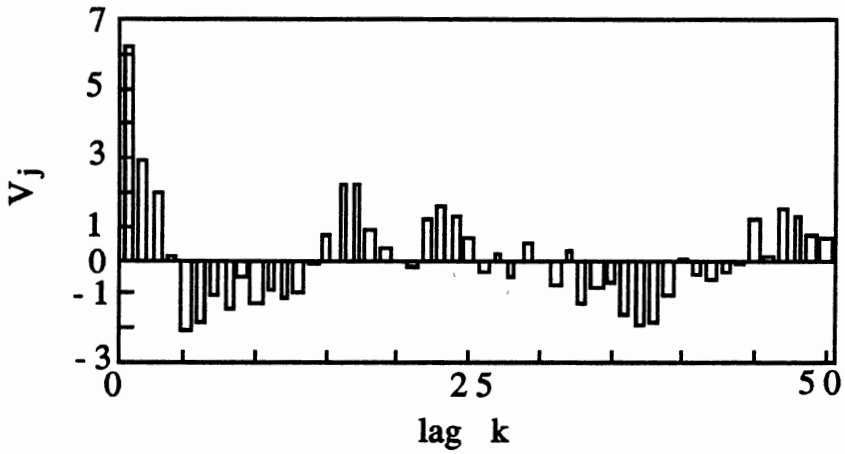

Figure 16. Impulse Response Estimates from Summer TF Model (III.26)

The next step in identifying the transfer function model is to identify the noise model. With the generated noise series at hand, the ARMA noise model can be easily found by employing a series of steps similar to those outlined in chapter II.

The resulting summer transfer function model is

$$Z_t=\frac{4.50+1.45B}{1-.848B+.231B^2}X_t+\frac{1-.703B^{24}-.111B^{48}}{\nabla_1\nabla_{24}(1-.136B-.182B^2)(1-.470B^{168})}a_t \qquad \text{(III.27)}$$

Using (III.27), 1 to 24 hour ahead load forecasts were made for the same three week period (i.e., July 29 - August 17, 1983). Figure 17 shows the average percent errors for each day with the total average percent error being 4.30, indeed better than the 4.49 obtained by the ARIMA model.

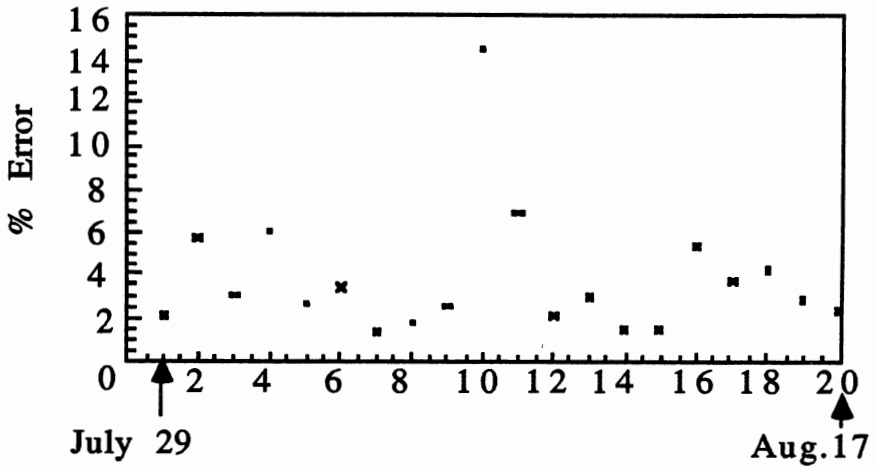

Figure 17. Average Percent Forecasting Errors Using (III.27)

TABLE IV

COMPARISON OF FORECASTING ERRORS FOR THE ARIMA AND TF MODELS (1983) AVERAGE ABSOLUTE % ERROR

Season	Forecasting period	ARIMA	Standard TF
Summer	Jul 29-Aug 17, 1983	4.17	3.82
Fall	Oct.21-Nov.9,1983	4.68	4.49
Winter	Jan 21-Feb 9, 1983	3.85	3.35
Spring	April 22-May 11,1983	5.24	5.53

The transfer function models for the other seasons were similarly developed. Table IV shows the average absolute percent errors for each model. From the above results, we may have the following conclusions:

(1) The transfer function model, with temperature as an explicit input, does provide a better performance, on the average, than the ARIMA model. This is especially true for summer and winter seasons when the load curve is greatly influenced by temperatures;

(2) Since the power load is a multi-variable (e.g., weather factors, shutdown of industrial plants, strikes, ect.) and, thus, a complex process, the past load history and hence the noise model are dominant in load forecasts. This may be explained by the fact that the past load itself actually inexplicitly contains all the factors including temperature information that embody a load curve, and

(3) A nonlinear model might produce better results and this will be the major topic in the following chapters.

A Nonlinear Extension

Before we extensively study nonlinear models for the STLF, in this section we make a simple nonlinear extension to the previous standard transfer function model with the goal to indicate that a nonlinear model development might be a fruitful research area for improvement of the STLF.

To see the existence of some nonlinear relationship contained in the power load, a plot of 1983 peak daily power load readings vs.

1983 peak daily temperature readings for the Tulsa area is shown in Figure 18, where the solid line represents the third order polynomial which is the best fit to the data using time series/regression method [3]

$$Y = 1810.0 + 21.4T - 1.04T^2 + 0.0093T^3 \qquad \text{(III.38)}$$

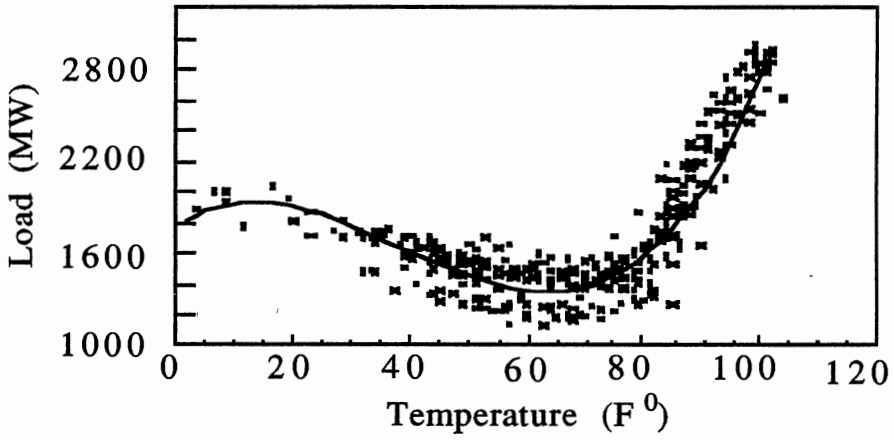

Figure 18. Scatter Diagram of Hourly Loads vs Hourly Temperatures for 1983

Now if we use this "new temperature" Y instead of temperature T itself as input to a transfer function, and then take a series of steps similar to developing the standard transfer function models, the nonlinear extended transfer function models will be found and will be more accurate, because larger weights are added to the summer and winter temperature variations than to the other seasons. Intuitively, this can be understood by the fact that the temperature variation during the summer and winter seasons has a

much larger effect on load than that during the spring and fall seasons and, hence, the nonlinear transfer function model more closely reflects the actual relationship between the loads and temperatures.

Table V compares the forecasting results with those from the previous linear models. From this Table, we see that a simple nonlinear extension provides us with a significant improvement in load forecasting. This encouraging result offers some promise for load forecasting improvement by further investigations of nonlinear models.

TABLE V

COMPARISON OF FORECASTING ERRORS
FOR THE THREE DIFFERENT MODELS
AVERAGE ABSOLUTE % ERROR

Season	Forecasting Period	ARIMA	Standard TF	Nonlinear TF
Summer	Jul 29-Aug 17, 1983	4.17	3.82	3.55
Fall	Oct.21-Nov.9,1983	4.68	4.49	3.41
Winter	Jan 21-Feb 9, 1983	3.85	3.35	2.84
Spring	April 22-May 11,1983	5.24	5.53	5.16

CHAPTER IV

GENERAL NONLINEAR MODELS

In the previous chapters, we developed the ARIMA models and the TF models. These models are used to describe those processes that satisfy the twin assumptions of linearity and stationarity which are commonly made in the conventional time series analysis. However, a nonlinear extended transfer function model indicated that further nonlinear investigation might be very fruitful for the STLF. In this chapter, the general nonlinear model is considered for the purpose of clarifying the concepts of linear models and nonlinear models. A statistical test for testing the assumption of linearity of a process is then presented. Some numerical examples of testing the departures from linearity are included also.

Volterra Series Expansions

In chapter II, the most general form of model for $\{Z_t\}$ was given by a function f(.) which describes the relationship between past, current and future observations (see (II.9)). Just as dealing with linear models, we assume that Z_t depends only on past values and not on future ones. Thus (II.9) takes the form

$$f(Z_t,Z_{t-1},Z_{t-2}, \ldots) = a_t \qquad \text{(IV.1)}$$

Suppose that the model (IV.1) is "invertible". In other words, (IV.1)

may be "solved" so that Z_t is expressed by

$$Z_t = g(a_t, a_{t-1}, a_{t-2}, \ldots) \tag{IV.2}$$

where g(.) is some (nonlinear) function of a_t, a_{t-1}, a_{t-2}, $\cdots$. With a Taylor series expansion on g(.) about the point $\mathbf{0}=(0,0,0,\ldots)$, we obtain the Volterra series as follows

$$Z_t = \alpha + \sum_{u=0}^{\infty} \theta_u a_{t-u} + \sum_{u=0}^{\infty}\sum_{v=0}^{\infty} \theta_{uv} a_{t-u} a_{t-v} + \sum_{u=0}^{\infty}\sum_{v=0}^{\infty}\sum_{w=0}^{\infty} \theta_{uvw} a_{t-u} a_{t-v} a_{t-w} + \ldots \tag{IV.3}$$

where

$$\alpha = g(0), \theta_u = \left(\frac{\partial g}{\partial a_{t-u}}\right)\Big|_{\theta=0}, \quad \theta_{uv} = \left(\frac{\partial^2 g}{\partial a_{t-u}\partial a_{t-v}}\right)\Big|_{\theta=0},$$

$$\theta_{uvw} = \left(\frac{\partial^3 g}{\partial a_{t-u}\partial a_{t-v}\partial a_{t-w}}\right)\Big|_{\theta=0}, \text{etc.} \tag{IV.4}$$

(IV.3) provides an important representation for general nonlinear models. If we approximate Z_t by the second term in (IV.3), then Z_t is described by a linear model which, under the assumption of invertibility, is identical to the general linear model (II.11).

From Volterra series point of view, a linear model is a special case of nonlinear models. In general, in order to be able to properly describe a given process a nonlinear model investigation is necessary. For instance, if the system is linear, then a nonlinear model will give no improvement in forecasting and this information

helps us to determine whether the system is linear or not. On the other hand, if the system is indeed nonlinear, then a simple linear model approximation may not be sufficient and a nonlinear model should be considered.

Note that, unlike a linear model which may be defined by a single transfer function, a nonlinear model requires an infinite sequence of higher-order transfer functions to characterize the relationship (IV.3), namely,

$$\Gamma_1(\omega_1) = \sum_{u=0}^{\infty} \theta_u e^{-i\omega_1 u}$$

$$\Gamma_2(\omega_1,\omega_2) = \sum_{u=0}^{\infty}\sum_{v=0}^{\infty} \theta_{uv} e^{-i(\omega_1 u + \omega_2 v)}$$

$$\Gamma_3(\omega_1,\omega_2,\omega_3) = \sum_{u=0}^{\infty}\sum_{v=0}^{\infty}\sum_{w=0}^{\infty} \theta_{uvw} e^{-i(\omega_1 u + \omega_2 v + \omega_3 w)} \qquad \text{(IV.5)}$$

Tests for Linearity

In order to be able to properly describe a given process, a statistical test for testing the assumption of linearity of the process is considered in this section. This test provides us with helpful information about the system. The importance of bispectral analysis in studying the possible departures from linearity is also discussed.

Spectral and Bispectral Density Functions

Let $\{Z_t\}$ have finite moments and be weakly stationary up to the sixth-order with the first three moments given by

$$\mu = E[Z(t)]$$

$$R(k) = \mathrm{Cov}(Z_t, Z_{t+k}),\ k = 0,\pm1,\pm2,\ldots$$

and
$$C(t_1,t_2) = E[(Z_t-\mu)(Z_{t+t_1}-\mu)(Z_{t+t_2}-\mu)] \qquad \text{(IV.6)}$$

It is easy to check that the following symmetric relations hold:

$$R(k) = R(-k)$$

$$C(t_1,t_2) = C(t_2,t_1) = C(-t_1,t_2-t_1) = C(t_1-t_2,-t_2), \qquad \text{(IV.7)}$$

The spectral density function, $f(\omega)$, and the bispectral density function, $f(\omega_1,\omega_2)$, are defined as the Fourier transforms of $R(k)$ and $C(t_1,t_2)$

$$f(\omega) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} R(k)e^{-ik\omega},\quad |\omega| \le \pi$$

$$f(\omega_1,\omega_2) = \frac{1}{(2\pi)^2}\sum_{t_1=-\infty}^{\infty}\sum_{t_2=-\infty}^{\infty} C(t_1,t_2)e^{-it_1\omega_1 - it_2\omega_2},$$

$$-\pi \le \omega_1,\omega_2 \le \pi \qquad \text{(IV.8)}$$

Using the symmetry properties (IV.7), we have

$$f(\omega) = f(-\omega)$$

$$f(\omega_1,\omega_2) = f(\omega_2,\omega_1) = f(\omega_1,-\omega_1-\omega_2) = f(-\omega_1,-\omega_2)^*, \qquad \text{(IV.9)}$$

where * denotes the complex conjugate. The estimates of μ and $R(k)$ are given in (II.3) and, similarly, the estimate of $C(t_1,t_2)$ is given by

$$\hat{C}(t_1,t_2) = \frac{1}{N}\sum_{t=1}^{N-\gamma}(Z_t-\hat{\mu})(Z_{t+t_1}-\hat{\mu})(Z_{t+t_2}-\hat{\mu}) \qquad \text{(IV.10)}$$

where $\gamma=\max(0,t_1,t_2)$; $t_1\ge0$, $t_2\ge0$.

The estimates of $f(\omega)$ and $f(\omega_1,\omega_2)$ are given by

$$\hat{f}(\omega)=\frac{1}{2\pi}\sum_{k=-M}^{M}\lambda(\frac{k}{M})\hat{R}(k)\cos(k\omega)$$

$$\hat{f}(\omega_1,\omega_2)=\frac{1}{(2\pi)^2}\sum_{t_1=-M}^{M}\sum_{t_2=-M}^{M}\lambda(\frac{t_1}{M},\frac{t_2}{M})\hat{C}(t_1,t_2)e^{-it_1\omega_1-it_2\omega_2} \qquad \text{(IV.11)}$$

where M is known as the truncation point, and $\lambda(.)$ and $\lambda(.,.)$ are one dimensional and two dimensional lag windows, respectively. The two dimensional lag windows may be constructed from one dimensional lag windows by

$$\lambda(s_1,s_2) = \lambda(s_1)\lambda(s_2)\lambda(s_1-s_2) \qquad \text{(IV.12)}$$

If $\lambda(s)=\lambda(-s)$, it is easy to show that (IV.12) satisfies the following symmetric relations

$$\lambda(s_1,s_2) = \lambda(s_2,s_1) = \lambda(-s_1,s_2-s_1) = \lambda(s_1-s_2,-s_2) \qquad \text{(IV.13)}$$

Many standard lag windows are available for the estimates of $f(\omega)$ and $f(\omega_1,\omega_2)$ as shown in Table VI. The choice of a specific window is usually arbitrary. The reader is referred to [13] for further details about lag windows.

It is well known (see [14],[15]) that $\hat{f}(\omega)$ is a consistent estimate of $f(\omega)$ and $\hat{f}(\omega_1,\omega_2)$ is a consistent estimate of $f(\omega_1,\omega_2)$.

TABLE VI

STANDARD LAG WINDOWS

Daniell window	$\lambda_D(s)=\frac{\sin(s\pi)}{s\pi}$	
Tukey-Hamming window	$\lambda_T(s)=\begin{cases}0.54+0.46\cos\pi s\\0\end{cases}$	$\begin{matrix}\|s\|<1\\\text{otherwise}\end{matrix}$
Parzen window	$\lambda_p(s)=\begin{cases}1-6s^2+6\|s\|^3\\2(1-\|s\|)^3\\0\end{cases}$	$\begin{matrix}\|s\|\le\frac{1}{2}\\\frac{1}{2}\le s\le 1\\\text{otherwise}\end{matrix}$

Principle and Procedure of the Test

Consider the linear representation (II.11) of a process $\{Z_t\}$

$$\sum_{i=-\infty}^{\infty}\phi_i Z_{t-i}=a_t$$

or
$$\phi(B)Z_t=a_t \qquad \text{(IV.14)}$$

If $\phi(B)$ is invertible, then (IV.14) may be written as

$$Z_t=\phi^{-1}(B)a_t=\sum_{u=-\infty}^{\infty}\theta_u a_{t-u} \qquad \text{(IV.15)}$$

where $\{a_t\}$ is a sequence of zero mean independent, identically distributed random variables. It can be shown [16] that $f(\omega)$ and $f(\omega_i,\omega_j)$ of the process $\{Z_t\}$ satisfying (IV.15) are given by

$$f(\omega)=\frac{\sigma_a^2}{2\pi}|H(\omega)|^2$$

$$f(\omega_i,\omega_j)=\frac{\mu_3}{(2\pi)^2}H(-\omega_i-\omega_j)H(\omega_i)H(\omega_j) \qquad \text{(IV.16)}$$

where

$$H(\omega)=\sum_{u=0}^{\infty}\theta_u e^{-iu\omega} \qquad \text{(IV.17)}$$

let

$$X_{ij}=\frac{\left|f(\omega_i,\omega_j)\right|^2}{f(\omega_i)f(\omega_j)f(\omega_i+\omega_j)} \qquad \text{(IV.18)}$$

then from (IV.16) and (IV.18) we have

$$X_{ij}=\frac{\mu_3^2}{2\pi\sigma_a^6}, \qquad \text{for all } i,j \qquad \text{(IV.19)}$$

The ratio X_{ij} is called the normalized bispectrum, and is a constant for all ω_i and ω_j if $\{Z_t\}$ is a linear process. This implies that if X_{ij} turns out to be a function of (ω_i,ω_j), then $\{Z_t\}$ must be a nonlinear process. Therefore, the constancy of X_{ij} may be used as a test for linearity of the process.

To begin the test of the constancy of X_{ij}, we first form a px1 column vector $\mathbf{Y}=(Y_1,Y_2,.....,Y_p)^T$ with $Y_k=\hat{X}_{ij}$ for some pairs of (i,j), where k=1,2,.....,p and $\hat{X}_{ij}$ is an estimate of X_{ij}

$$\hat{X}_{ij} = \frac{\left|\hat{f}(\omega_i, \omega_j)\right|^2}{\hat{f}(\omega_i)\hat{f}(\omega_j)\hat{f}(\omega_i + \omega_j)} \qquad \text{(IV.20)}$$

Now, under the null hypothesis that $\{Z_t\}$ is linear process, $E[Y_1] = E[Y_2] = \ldots = E[Y_p]$. Next, for each Y_k, a set of n uncorrelated estimates of Y_k is formed as shown in Figure 19. Consequently, we obtain a random sample of n estimates of $\mathbf{Y}$, which are denoted by $\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n$. The mean and variance of $\mathbf{Y}$ may be estimated by

$$\overline{\mathbf{Y}} = \frac{1}{n}\sum_{i=1}^{n} \overline{\mathbf{Y}}_i, \qquad \mathbf{S}_{\mathbf{Y}} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{Y}_i - \overline{\mathbf{Y}})(\mathbf{Y}_i - \overline{\mathbf{Y}})^T \qquad \text{(IV.21)}$$

Introducing the difference matrix $\mathbf{B}$ of order QxP, where Q=P-1, as follows

$$\mathbf{B} = \begin{bmatrix} 1 & -1 & 0 & .. & 0 \\ 0 & 1 & -1 & .. & 0 \\ . & . & & . & \\ . & . & & . & \\ 0 & & 0 & 1 & -1 \end{bmatrix} \qquad \text{(IV.22)}$$

a new column vector $\boldsymbol{\beta}$ of order Q defined by $\mathbf{BY}$ is then asymptotically jointly normally distributed with mean vector $\mathbf{0}$ and the variance-covariance matrix $\mathbf{BS_Y B}^T$. Let $\overline{\boldsymbol{\beta}}=\mathbf{B}\overline{\mathbf{Y}}$ and $\mathbf{S}=\mathbf{BS_Y B}^T$, the likelihood ratio test for testing $\overline{\boldsymbol{\beta}}=\mathbf{0}$ against the alternative $\overline{\boldsymbol{\beta}}^T\mathbf{S}^{-1}\overline{\boldsymbol{\beta}} > 0$ leads to the rejection of the hypothesis if the statistic $T^2=\overline{\boldsymbol{\beta}}^T\mathbf{S}^{-1}\overline{\boldsymbol{\beta}} > \lambda_o$, where λ_o is a constant determined by the significance level α. The statistic $F_2=[(n-Q)/Q]/T^2$ has, under the null hypothesis, an F

distribution with (Q,n-Q) degrees of freedom [17].

Remarks on the choice of parameters

(1) Figure 19 shows a typical bispectrum sample for the linearity test with parameters K=6, P=7 and r=2, where K is the number of equally spaced points in the interval $(0,\pi)$, P is the number of the "fine" grids and n=4r+1 is the total number of points in each "fine" grid.

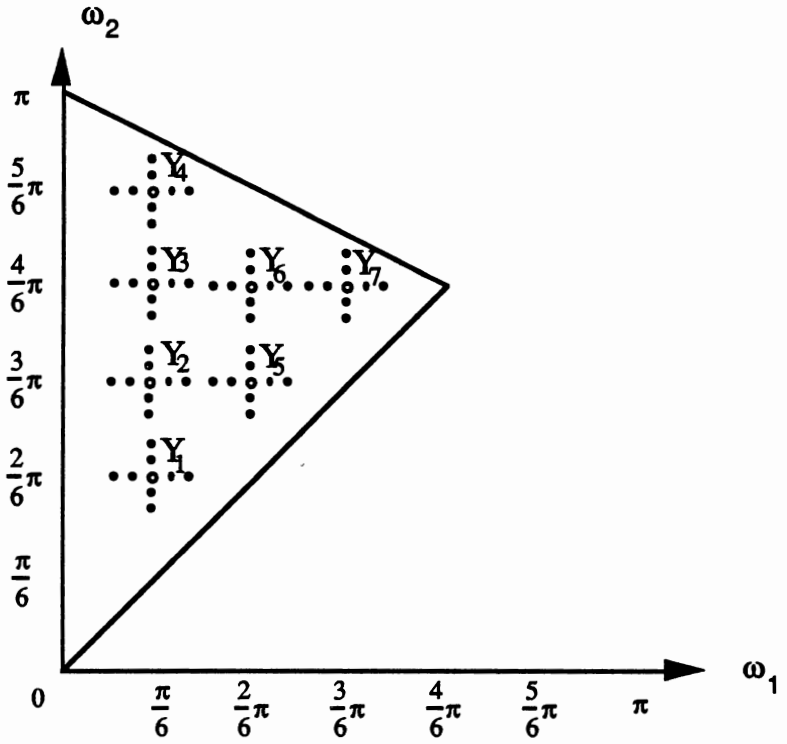

Figure 19. Bispectrum Sample for Linearity Test with K=6, r=2, p=7, n=9

(2) Another important parameter to choose in implementing the test is the "distance" d, which is defined in the value $\pi d/N$, the distance between two neighbouring points on the "fine" grid, where N is the number of observations. On the one hand, to ensure that the spectral and bispectral estimates are uncorrelated, d should be chosen as large as possible. On the other hand, to ensure no overlap between "fine" grids, we require $d < N/[K(2r+1)]$. Therefore, with the parameters K and r given, the value d is determined by N as above (e.g., if $K = 6$ and $r = 2$ then $d < N/30$).

(3) The value of truncation point M in (IV.11) is a very important parameter in implementing the linearity test. In general, increasing the parameter M means to increase the variance and decrease the bias of the estimates of $f(\omega)$ and $f(\omega_1,\omega_2)$, and vice versa. Thus, the value of M should be chosen such that both the variance and the bias are considered. The rule of thumb is that if it is possible, M should be less than the square root of N [13]. In addition, the value of M is different from window to window. Equation (IV.23) shows the relationship between M's using different windows

$$\frac{M_1}{M_{R,1}} = \frac{M_2}{M_{R,2}} \qquad \text{(IV.23)}$$

where M_R is some window related parameter. (IV.23) implies that the ratios between the values of M and M_R using different windows are the same for all windows. Table VII shows the M_R values for some standard windows [13].

As an example, suppose that Parzen and Daniell windows are used, then we have

$$\frac{M_{Par}}{M_{R,Par}} = \frac{M_{Dan}}{M_{R,Dan}}$$

If $M_{Par}=10$, then

$$M_{Dan} = \frac{M_{R,Dan}}{M_{R,Par}} M_{Par} = \frac{1.7}{3.1}(10) \cong 5$$

TABLE VII

THE M_R VALUES

Window	Daniell	Tukey	Parzen
M_R	1.7	1.86	3.1

Some Numerical Examples

In order to check the effectiveness of the above test, numerous simulated linear and nonlinear time series are generated. As an illustration, we consider the following series:

Series 1: $Z_t = 1.3Z_{t-1} - 0.4Z_{t-2} + a_t$

Series 2: $Z_t = 0.8a_{t-1} + a_t$

Series 3: $Z_t = 0.4Z_{t-1} + 0.4Z_{t-1}a_{t-1} + a_t$

Series 4: $Z_t = 0.4Z_{t-1} + 0.6Z_{t-1}a_{t-1} + a_t$

where $\{a_t\}$ is a white noise sequence. The first two series are linear ones since they are generated from the AR(2) and the MA(1) models, respectively. The third and fourth series are nonlinear because of the presence of the crossing term $Z_{t-1}a_{t-1}$. For all these series, the parameters for constructing the linearity test are as follows:

$$P = 7, K = 6, r = 2, n = 9, d = 15$$
$$N = 500, M=22 \qquad \text{(IV.24)}$$

The one dimensional and two dimensional Daniell windows are chosen as lag windows. Under the null hypothesis that the series $\{Z_t\}$ is linear, the statistic F_2 has an F distribution with (6,3) degress of freedom.

TABLE VIII

THE F_2 VALUES FOR EXAMPLE SERIES

Series	F_2	5% upper point of F(6,3)
1	1.8	8.94
2	6.7	8.94
3	42.1	8.94
4	54.6	8.94

Table VIII shows the F_2 values for the series. From this table, we can confidently conclude that series 3 and 4 are nonlinear because their F_2 values are much greater than the 8.94, the 5% upper point of F(6,3). However, in general we cannot conclude that series 1 and 2 are definitely linear based only on F_2 values, of course, they actually are in this illustration. This is because that the test is not exhaustive [13]. For example, the bispectrum for some nonlinear processes can be zero and, thus, the F_2 values may be less than the 5% upper point of F(6,3). Figures 20 - 23 illustrate the normalized bispectral estimates for series 1 - 4, respectively.

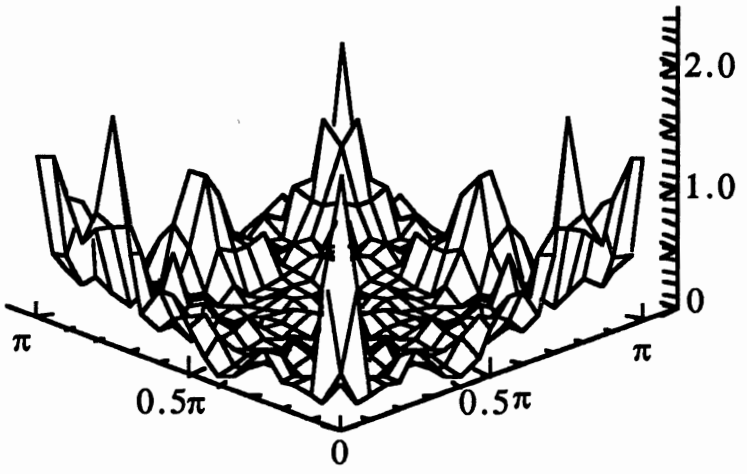

Figure 20. The Normalized Bispectral Estimates for Series 1

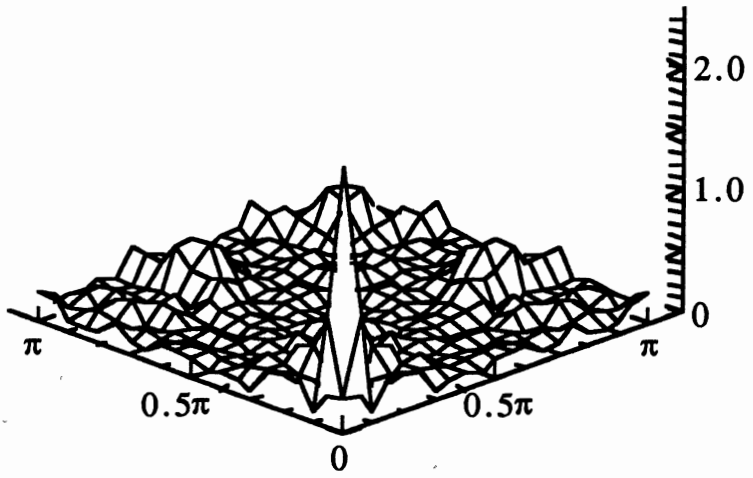

Figure 21. The Normalized Bispectral Estimates for Series 2

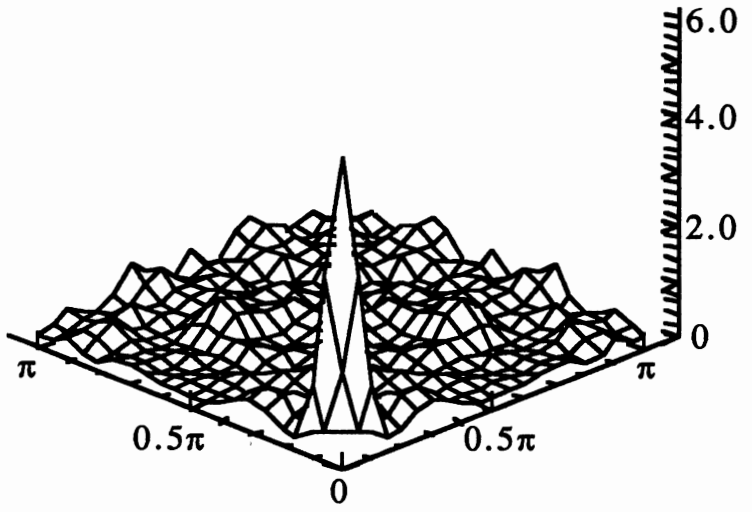

Figure 22. The Normalized Bispectral Estimates for Series 3

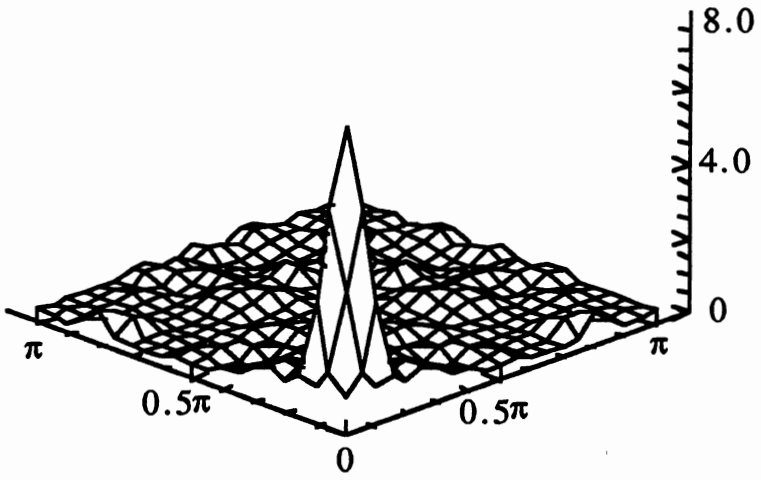

Figure 23. The Normalized Bispectral Estimates for Series 4

Figures 22 - 23 clearly show some trends (near zero the bispectrum is nonzero), indicating that there is some nonlinearity involved in the series, and that the larger the nonlinearity is, the more obvious is the trend. It is also interesting to note that although the series 1 and 2 are generated by linear models, their normalized bispectral estimates, show some kind of nonlinearity also. This is unexpected. The reason for it may be due to the limitation of data length and the accuracy of the white noise sequence $\{a_t\}$. An extended data set of length 1000 was generated for each series, and the resulting normalized bispectral estimates were much improved. The plots for series 3 and 4 were similar to those for 500 points, while smaller and erratic values were obtained for series 1 and 2. This confirmed our earlier suspicion that the trends seen in the normalized bispectrum of the linear models were due to the limited length of the sequence. Nevertheless, the test indeed provides

important information on the presence of nonlinearity, as shown in Table VIII.

TABLE IX

THE F_2 VALUES FOR POWER LOAD SERIES

Season	F_2	5% upper point of F(6,3)
Spring	40.2	8.94
Summer	95.3	8.94
Fall	93.9	8.94
Winter	20.1	8.94

With the same parameters as in (IV.24), the F_2 values for our differenced power load ($\nabla_1 \nabla_{24} Z_t$) are calculated using the same Daniell windows as before (see Table IX). As expected, these F_2 values clearly indicate that nonlinear models may better describe the power load and, thus, may make the load prediction more accurate. Figures 24 - 27 are plots of the normalized bispectral estimates for each season. As is seen from these figures, each one represents some nonlinear effect with different behaviors. This might suggest that a different nonlinear model may be needed for different season.

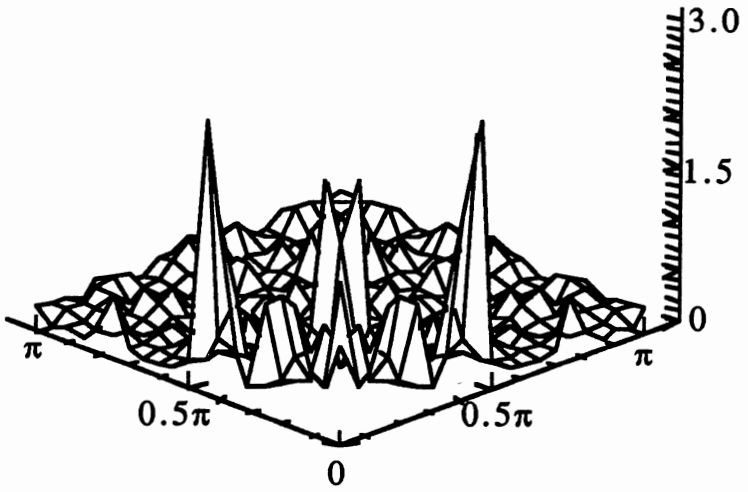

Figure 24. The Normalized Bispectral Estimates for Power Load in Spring Season

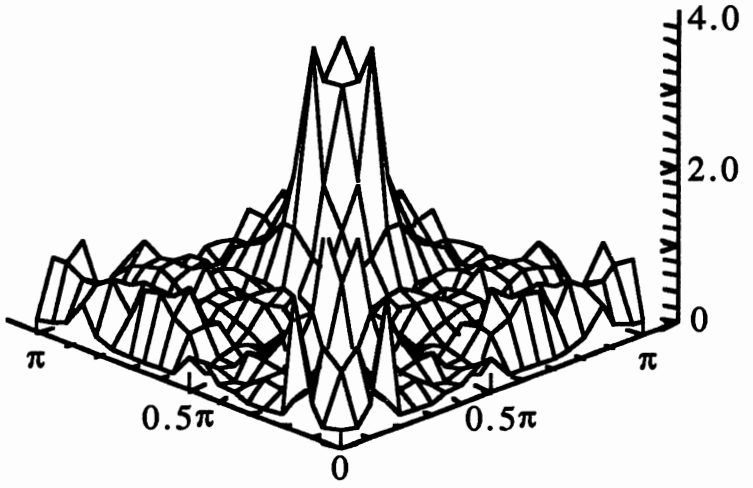

Figure 25. The Normalized Bispectral Estimates for Power Load in Summer Season

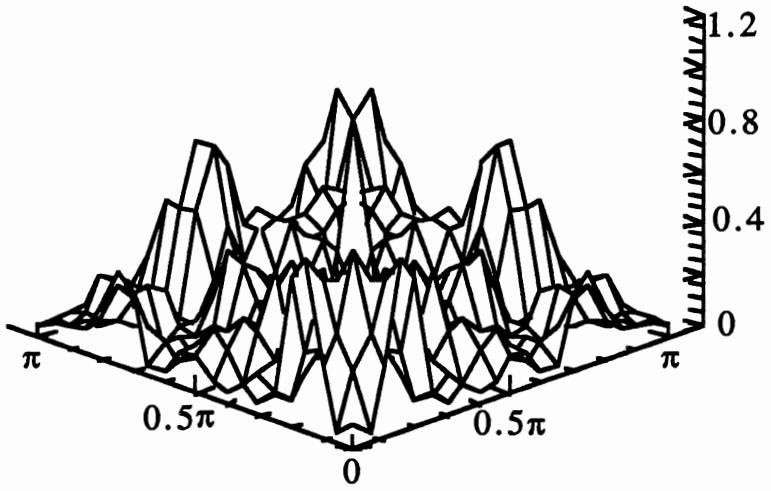

Figure 26. The Normalized Bispectral Estimates for Power Load in Fall Season

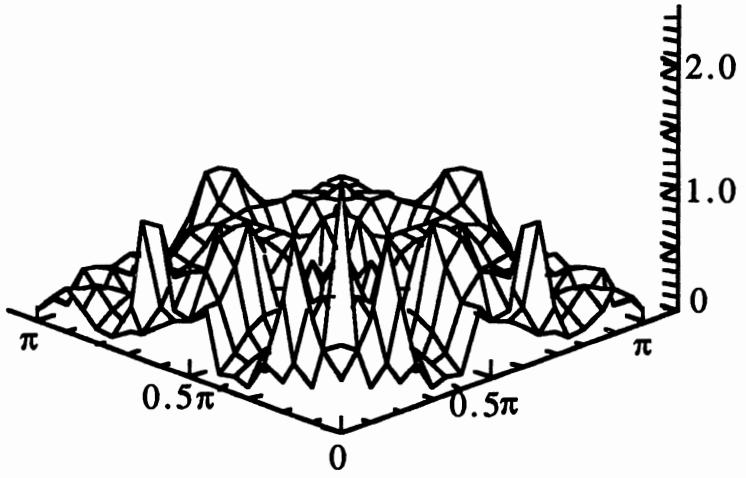

Figure 27. The Normalized Bispectral Estimates for Power Load in Winter Season

In summary, in this chapter we presented a general nonlinear model, for which linear models turn out to be special cases. We then considered a statistical test for testing the linearity of a given process. Numerical simulation results show that the test can provide us with important information about a given system. Specifically, the

positive testing result for our power loads indicates the need for further investigations of nonlinear models. In next chapter, and the chapters following, we discuss a few nonlinear model schemes. Hopefully, these nonlinear models may provide us with better forecasting performance.

CHAPTER V

BILINEAR TIME SERIES MODELS

In chapter IV, we demonstrated the existence of a nonlinear relationship between the power load $\{Z_t\}$ and the white noise "input" $\{a_t\}$. Accordingly, in order to be able to properly describe this relationship, a study of nonlinear models seems to be necessary. Although, in chapter IV, a general nonlinear model was also presented, in practice this model is not very useful in view of the fact that an infinite set of higher-order transfer functions are required to sufficiently describe a nonlinear system.

Recently a special class of nonlinear time series model, namely the bilinear time series model, has been proposed and studied extensively by Granger and Anderson [19], Priestly [20], and Subba Rao and Gabr [13]. It has been shown [13] that the bilinear models may be successfully applied to many real problems. In this chapter, we introduce the bilinear models to the power load application with the goal to better describe the relationship between the power load and the white noise "input".

One of the disadvantages of the conventional bilinear models is that the number of parameters to be estimated is large. A new form of bilinear model is considered and it is shown that this modified bilinear model may significantly reduce the number of parameters and, in the meanwhile, keep the similar features of the conventional

bilinear models. Another improvement over the conventional bilinear model is gained by adding periodic or "seasonal" terms to the model so that a more general bilinear model is created.

Using the proposed bilinear model, 4 load forecasting models, which correspond to spring, summer, fall and winter seasons, are developed. The forecasting ability of these bilinear models is then compared with those for the linear ones.

Bilinear Models

The conventional discrete time bilinear model BL(p,q,m,k) can be expressed as

$$Z_t + \sum_{j=1}^{p} \phi_j Z_{t-j} = \sum_{j=0}^{q} \theta_j a_{t-j} + \sum_{i=1}^{m} \sum_{j=1}^{k} b_{ij} Z_{t-i} a_{t-j} \qquad \text{(V.1)}$$

where $\{Z_t\}$ and $\{a_t\}$ are real valued time series and a strict white noise process, i.e. a sequence of independent zero mean random variables, respectively.

As is seen from the definition (V.1), an ARMA(p,q) model turns out to be a special case of the bilinear model (when $b_{ij} = 0$ for all i,j). To some extent this parallels the argument that a linear model is a special case of the general nonlinear model (IV.3). As a matter of fact, it can be shown [20] that the bilinear model can approximate to an arbitrary degree of accuracy any "well-behaved" Volterra series relationship over a finite time interval. In view of this, the bilinear models represent a powerful class of nonlinear models.

We now define the following bilinear model

$$\phi(B)Z_t = \theta(B)a_t + (\xi(B)Z_t)(\zeta(B)a_t) \qquad (V.2)$$

where $\phi(B)$ and $\theta(B)$ are given by (II.13) and (II.17), respectively and rewritten as follows:

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$$
$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q$$

$\xi(B)$ and $\zeta(B)$ are similarly defined as follows:

$$\xi(B) = B + \xi_2 B^2 + \ldots + \xi_m B^m$$
$$\zeta(B) = \zeta_1 B + \zeta_2 B^2 + \ldots + \zeta_k B^k \qquad (V.3)$$

In this case, we have

$$\begin{aligned}
&(\xi(B)Z_t)(\zeta(B)a_t) \\
&= ((B+\xi_2 B^2+\ldots+\xi_m B^m)Z_t)((\zeta_1 B+\zeta_2 B^2+\ldots+\zeta_k B^k)a_t) \\
&= (Z_{t-1}+\xi_2 Z_{t-2}+\ldots+\xi_m Z_{t-m})(\zeta_1 a_{t-1}+\zeta_2 a_{t-2}+\ldots+\zeta_k a_{t-k}) \\
&= \zeta_1 Z_{t-1}a_{t-1}+\zeta_2 Z_{t-1}a_{t-2}+\ldots+\zeta_k Z_{t-1}a_{t-k} \\
&\quad +\xi_2\zeta_1 Z_{t-2}a_{t-1}+\xi_2\zeta_2 Z_{t-2}a_{t-2}+\ldots+\xi_2\zeta_k Z_{t-2}a_{t-k} \\
&\qquad\qquad \vdots \\
&\quad +\xi_m\zeta_1 Z_{t-m}a_{t-1}+\xi_m\zeta_2 Z_{t-m}a_{t-2}+\ldots+\xi_m\zeta_k Z_{t-m}a_{t-k}
\end{aligned} \qquad (V.4)$$

Now with (V.3) and (V.4), we may easily see that models (V.2) and (V.1) have exactly the same form. The interest thing is that the number of parameters required by each model may be quite different. For instance, consider p=q=2 and m=k=8, up to 68

parameters are needed in (V.1) while only 19 parameters in (V.2). Obviously, the later situation is much easier to handle for both identification and forecasting procedures.

Another important feature of model (V.2) is that we may easily generalize it to include periodic terms which, as we have seen earlier, appear to reflect intrinsic behaviors of many real processes. Similar to the "seasonal" ARIMA models given by (II.23), a general "seasonal" bilinear model may be defined as

$$\phi_p(B)\phi'_{p'}(B^{24})\phi''_{p''}(B^{168})W_t = \theta_q(B)\theta'_{q'}(B^{24})\theta''_{q''}(B^{168})a_t$$
$$+(\xi_m(B)\xi'_{m'}(B^{24})\xi''_{m''}(B^{168})W_t)(\zeta_k(B)\zeta'_{k'}(B^{24})\zeta''_{k''}(B^{168})a_t) \qquad (V.5)$$

where

$$\xi'_{m'}(B^{24}) = 1 + \xi'_1B^{24} + \xi'_2B^{48} + \ldots + \xi'_{m'}B^{24m'} \quad m\neq 0$$
$$= B^{24} + \xi'_2B^{48} + \ldots + \xi'_{m'}B^{24m'} \quad \text{otherwise}$$

$$\xi''_{m''}(B168) = 1 + \xi''_1B^{168} + \ldots + \xi''_{m''}B^{168m''} \quad m\neq 0 \text{ or } m'\neq 0$$
$$= B^{168} + \ldots + \xi''_{m''}B^{168m''} \quad \text{otherwise}$$

$$\zeta'_{k'}(B^{24}) = 1 + \zeta'_1B^{24} + \zeta'_2B^{48} + \ldots + \zeta'_{k'}B^{24k'} \quad k\neq 0$$
$$= B^{24} + \zeta'_2B^{48} + \ldots + \zeta'_{k'}B^{24k'} \quad \text{otherwise}$$

$$\zeta''_{k''}(B^{168}) = 1 + \zeta''_1B^{168} + \ldots + \zeta''_{k''}B^{168k''} \quad k\neq 0 \text{ or } k'\neq 0$$
$$= B^{168} + \ldots + \zeta''_{k''}B^{168k''} \quad \text{otherwise}$$

(V.6)

and

$$W_t = \nabla_1^d\nabla_{24}^D\nabla_{168}^{D'}Z_t.$$

Now model (V.5) has the ability to model both multiperiodicity and nonlinearity for a given process and, thus, forms a very useful model class for general applications.

It should be noted that model (V.4) may produce a little bit larger residual sum of squares than model (V.1) of the same order due to the reduced number of degrees of freedom. However, in view of the AIC value, which will be discussed in detail shortly, and the ability to model multiperiodicity, the "seasonal" bilinear model given by (V.5) seems to be superior to the conventional bilinear model.

Identification and Estimation

For simplicity, let us consider a bilinear model of the form

$$Z_t + \sum_{j=1}^{p} \phi_j Z_{t-j} = \alpha + (\xi(B)Z_t)(\zeta(B)a_t) + a_t \qquad (V.7)$$

Obviously, (V.7) represents a particular case of the bilinear model (V.2) with the pure MA terms dropped and a constant α added. The identification and estimation procedure for (V.7) disscussed in the following can be easily extended for the general bilinear model (V.5).

Similar to the conventional identification procedures for the ARMA models, the steps of fitting (V.7) to a given stationary process $\{Z_t, t=1,2,...,N\}$ includes

(1) preliminary identification, i.e. determing the orders p, m, k;

(2) for given p, m, and k, estimating the parameters α, $\{\phi_j\}$, $\{\xi_j\}$, $\{\zeta_j\}$ and σ_a.

The tool used for order determination is based on a well-known criterion, namely the Akaike information criterion (AIC), which takes

into account the estimated residual variance σ_a^2 of the fitted model as well as the number of parameters in the model. It will be seen later that, when we use the AIC as the basis for the order selection procedure, it will be more convenient to discuss parameter estimation first and then order selection, even though it would appear logical to carry out the identification procedure in an opposite order.

Parameter Estimation

Suppose the order parameters p, m, k are given. The parameters of (V.7) may be estimated using a method which is similar to that for the linear models, i.e. the maximum likelihood algorithm. This is true because, in the case of the bilinear model, we assumed that the residuals a_t are independent $N(0,\sigma_a)$ random variables and that the mappings between Z_t and a_t are one-one.

Now let $\mathbf{h} = [h_1,h_2,\ldots,h_n]$ denote the complete set of parameters with

$$\begin{aligned} &h_i = \phi_i, \qquad i = 1, 2, \ldots, p; \\ &h_{p+1} = 1.0, \quad h_{p+i} = \xi_i, \quad i= 2, 3, \ldots, m; \\ &h_{p+m+i} = \zeta_i, \quad i = 1, 2, \ldots, k; \\ &h_{p+m+k+1} = \alpha \end{aligned} \tag{V.8}$$

where n=p+m+k+1 is the total number of parameters plus 1. The conditional maximum likelihood estimates of **h** are given by minimizing

$$S(\mathbf{h}) = \sum_{t=\gamma+1}^{N} a_t^2 \tag{V.9}$$

where N is the number of observations and $\gamma=\max(p,m,k)$, representing the number of initial values needed to evaluate the residuals $\{a_t\}$ from the model (V.7).

Using the recursive Newton-Raphson algorithm, the optimal estimates of $\mathbf{h}$ by minimizing $S(\mathbf{h})$ are given by

$$\mathbf{h}^{(i+1)} = \mathbf{h}^{(i)} - H^{-1}(\mathbf{h}^{(i)})G(\mathbf{h}^{(i)}) \tag{V.10}$$

where $\mathbf{h}^{(i)}$ is the estimated parameter vector obtained at the i^{th} iteration, and gradient vector G and Hessian matrix H are given by

$$G(\mathbf{h}) = [\frac{\partial S}{\partial h_1}, \frac{\partial S}{\partial h_2}, \ldots, \frac{\partial S}{\partial h_n}]^T$$

$$H(\mathbf{h}) = \{\frac{\partial^2 S}{\partial h_i \partial h_j}\} \tag{V.11}$$

It is not difficult to show that

$$\frac{\partial S}{\partial h_i} = 2 \sum_{t=\gamma+1}^{N} a_t \frac{\partial a_t}{\partial h_i}, \qquad i=1,2,\ldots,n \tag{V.12}$$

$$\frac{\partial^2 S}{\partial h_i \partial h_j} = 2 \sum_{t=\gamma+1}^{N} \frac{\partial a_t}{\partial h_i}\frac{\partial a_t}{\partial h_j} + 2 \sum_{t=\gamma+1}^{N} a_t \frac{\partial^2 a_t}{\partial h_i \partial h_j}, \qquad i,j=1,2,\ldots,n \tag{V.13}$$

It may be quite tedious to compute all the terms of the second sum of (V.13). However, if the parameter vector $\mathbf{h}^{(i)}$ converges to such values that the residuals $\{a_t\}$ are independent, then close to $\mathbf{h}^{(i)}$ the

second sum of (V.13) will be approximately zero since

$$E[a_t \frac{\partial^2 a_t}{\partial h_i \partial h_j}] = 0 \tag{V.14}$$

In this case

$$\frac{\partial^2 S}{\partial h_i \partial h_j} \approx 2 \sum_{t=\gamma+1}^{N} \frac{\partial a_t}{\partial h_i} \frac{\partial a_t}{\partial h_j}, \quad i,j = 1,2,\ldots,n \tag{V.15}$$

It can be shown that when (V.15) is used the estimate of the Hessian $H(\mathbf{h})$ is always assured to be positive semidefinite [20]. Note that the estimated Hessian $\tilde{H}(\mathbf{h})$ may be singular or close to singular, and, thus, some numerical problems may arise. One common way to deal with this problem is the Levenberg-Marquardt procedure [9]. In this procedure, an approximation

$$\tilde{H}(\mathbf{h}) + \delta I \tag{V.16}$$

is used for the Hessian, where δ is some small positive scalar and I is the nxn identity matrix.

With the above modified Hessian matrix (V.16), all the partial derivatives needed to implement the modified Newton-Raphson algorithm are now only those of (V.12). A set of recursive equations for these derivatives can be easily obtained. Differentiating (V.7) w.r.t. each of the parameters, we have

$$\frac{\partial a_t}{\partial \phi_j} = -\psi(\phi_j) + Z_{t-j}, \qquad j=1,2,\ldots,p$$

$$\frac{\partial a_t}{\partial \xi_j} = -\psi(\xi_j) - Z_{t-j}(\zeta(B)a_t), \qquad j=2,\ldots,m;$$

$$\frac{\partial a_t}{\partial \zeta_j} = -(\xi(B)Z_t)a_{t-j} - \psi(\zeta_j), \qquad j=1,2,\ldots,k;$$

$$\frac{\partial a_t}{\partial \alpha} = -\psi(\alpha) - 1 \qquad \text{(V.17)}$$

where

$$\psi(h_l) = (\xi(B)Z_t)(\zeta(B)\frac{\partial a_t}{\partial h_l}) \qquad \text{(V.18)}$$

The recursive Newton-Raphson algorithm is now complete and summarized as follows:

(1) Initialization

(i) set

$$a_t = \frac{\partial a_t}{\partial h_i} = 0, \qquad t=1,2,\ldots,\gamma; i=1,2,\ldots,n \qquad \text{(V.19)}$$

(ii) select a proper set of initial values for h's to obtain a good set of estimates of the parameters. A method for obtaining this type of initial values is considered in the following section.

(2) For a given set of parameters **h**(i), calculate the residuals $\{a_t, t=\gamma+1,\ldots,N\}$ using (V.7) and the derivatives $\partial a_t/\partial h_j$ using the recursive equations (V.17), (V.18) and (V.19).

(3) Evaluate the gradient vector G using (V.11) and (V.12) and

the modified Hessian matrix using (V.11), (V.15) and (V.16).

(4) Update the parameter estimate $\mathbf{h}^{(i+1)}$ using (V.10).

(5) Stop the iteration if the required accuracy is obtained (the criterion for accuracy is usually set by relative change of each parameter value for two adjacent iterations or the mean square error of residuals or both), otherwise go to step 2.

Remarks on Calculation of the Derivatives

In addition to calculating the derivatives directly by (V.17) and (V.18), numerical estimates of the derivatives have been found to be a very good alternative. In this method, the derivative of a_t w.r.t. a particular parameter h_j is obtained by perturbing h_j and, in the meanwhile, keeping the remaining parameters constant, i.e.

$$\frac{\partial a_t}{\partial h_j} \approx \left\{a_t(\mathbf{h}_j) - a_t(\mathbf{h}_0)\right\} / \delta_j, \quad j = 1,2,\ldots,n \tag{V.20}$$

where

$$\mathbf{h}_0 = [h_1 \ldots h_j \ldots h_n]$$
$$\mathbf{h}_j = [h_1 \ldots h_j + \delta_j \ldots h_n]$$

and δ_j is a small real number, typically δ_j=0.001. Clearly, the numerical method (V.20) has the advantage of general applicability and simplicity of implementation, because for any model all we need for the derivatives are just calculations of the residuals a_t's rather than their derivatives, which typically are much more involved.

Order Determination

As mentioned earlier, the Akaike information criterion AIC is used to determine suitable order values of p, m, k. The AIC is defined as

$$AIC = (N-\gamma)\log\hat{\sigma}_a^2 + 2(\text{number of independent parameters}) \tag{V.21}$$

with

$$\hat{\sigma}_a^2 = \frac{1}{N-\gamma} S(\hat{\mathbf{h}}) = \frac{1}{N-\gamma} \sum_{t=\gamma+1}^{N} \hat{a}_t^2 \tag{V.22}$$

where $\hat{\mathbf{h}}$ and $\hat{a}_t$ are the estimates of the parameters and the corresponding residuals, respectively. In using the AIC criterion, we fit a group of models with different combinations of p, m, k; the model with minimum AIC value is chosen. Clearly, to minimize the AIC value we need to strike a balance between reducing the value of the residual variance and reducing the number of model parameters.

In general, searching over a three dimensional grid for optimal values of p, m and k would be very costly. Rao [13] has suggested the following procedure.

(1) Find the best AR(p') model for Z_t.

(2) Set an upper bound for p, m, k, say Γ, which is greater than or equal to p'.

(3) For a given value of p, fit an AR(p) model.

(4) Fit a BL(p,0,1,1) model using the modified Newton-Raphson algorithm as described above with the AR(p) coefficients as initial

values for the $\{\phi_i\}$ and α, and $b_{11}=0$. Calculate the corresponding σ_a^2 and AIC value.

(5) Fit BL(p,0,1,2) and BL(p,0,2,1) models using the parameters of a BL(p,0,1,1) model as initial values and setting the remaining initial values to zero. Calculate the corresponding residual variances and AIC values for both fitted models and choose the one which has the smaller residual variance (This is equivalent to the choice of the smaller AIC value) to provide the starting values for BL(p,0,2,2) model.

(6) Similarly, fit BL(p,0,3,1), BL(p,0,1,3), BL(p,0,3,2), BL(p,0,2,3), BL(p,0,3,3), . . . until either m or k have reached the common upper bound Γ or the AIC value starts to increase as m and/or k increase. Figure 28 illustrates this searching procedure.

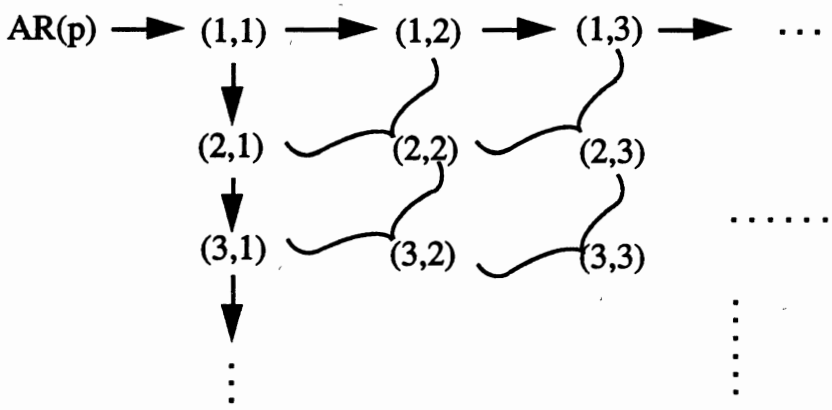

Figure 28. The "Nested" Search Scheme for Order Selection

(7) Repeat the steps from (3) to (6) for p=1,2,...,Γ.

The final model will be the one which has the minimum AIC value. Obviously, by this procedure we have determined the order parameters p, m, k and also the estimates of parameter values. Note that the steps (5) and (6) above correspond to the conventional bilinear models. In the case of the modified bilinear model, since the number of parameters is p+m+k instead of p+mk+1, the searching process may be significantly reduced.

Diagnostic Checks and Forecasts

In the case of linear models, we check the whiteness of a residual sequence by examining its autocorrelation functions as discussed in chapter II and chapter III. However, this kind of test for "white noise" based on the behavior of the autocorrelation functions may not be adequate when nonlinear models are considered. As an example, consider the process defined by

$$a_t = e_t + \beta e_{t-1} e_{t-2} \qquad \text{(V.23)}$$

where e_t is a strictly independent random variable with zero mean and constant variance. Clearly, a_t has zero mean and constant variance also. It is easy to show that the autocorrelation function of a_t satisfies $\rho_a(k)=0$, for all $k\neq 0$. This example clearly indicates that for nonlinear processes the residual sequence can be white even when there is some structure in the process which has not been adequately modeled. Second order analysis is not enough; a higher moment analysis is necessary. One method of checking the independence of

the a_t's through higher moment analysis may be constructed as follows.

Let $y_t=a_t^2$ (if a_t is independent, so is y_t). Define

$$W=\frac{\hat{\rho}_y(1)\hat{R}_y(0)\sqrt{N}}{\sqrt{\hat{R}_y(0,1,1)}} \tag{V.24}$$

where $\hat{R}_y(0)$ is the estimated variance of $\{y_t\}$, $\hat{\rho}_y(1)$ the estimated autocorrelation function of $\{y_t\}$ at lag 1 (see (II.2)) and $\hat{R}_y(0,1,1)$ the estimated fourth order central moment with $t_1=0$ and $t_2=t_3=1$, which is defined by

$$R_y(t_1,t_2,t_3)=E[(y_t-\mu_y)(y_{t+t_1}-\mu_y)(y_{t+t_2}-\mu_y)(y_{t+t_3}-\mu_y)] \tag{V.25}$$

The natural estimates of μ_y and $R_y(t_1,t_2,t_3)$ are obtained by

$$\bar{y}=\frac{1}{N}\sum_{t=1}^{N}y_t, \quad \hat{R}_y(t_1,t_2,t_3)=\frac{1}{N}\sum_{t=1}^{N-\gamma}(y_t-\bar{y})(y_{t+t_1}-\bar{y})(y_{t+t_2}-\bar{y})(y_{t+t_3}-\bar{y}) \tag{V.26}$$

where $\gamma=\max(t_1,t_2,t_3)$; $t_1\geq 0$, $t_2\geq 0$, $t_3\geq 0$. It is known [13] that the statistic W, under the hypothesis that y_t's are independent, is asymptotically distributed as standard normal N(0,1). Like the Q test, which was discussed in chapter II, if the average value of W is high compared with some percentage point of N(0,1), then y_t is not independent. Note that the W test is valid only for those processes which are stationary up to the fourth order.

We now consider forecasting using the bilinear model (V.7), which can be equivalently written as follows when all the

parameters are determined,

$$Z_t + \sum_{j=1}^{p} \phi_j Z_{t-j} = \alpha + \sum_{i=1}^{m} \sum_{j=1}^{k} b_{ij} Z_{t-i} a_{t-j} + a_t \qquad (V.27)$$

Similar to linear models, the minimum mean square error predictor of a future value Z_{t+m} may be found by taking the conditional expectation on both sides of (V.27). For bilinear models, however, the situation is more complicated and more computation is involved than for linear models. This is mainly due to the evaluation of expressions such as $E[Z_{t+m-i}a_{t+m-j}|Z_t,Z_{t-1},\ldots,Z_{t-n}]$ for $i<j<m$. Therefore, to obtain m step ahead forecasts by a given bilinear model we need, in addition to the procedure outlined in chapter II for linear model forecasts, the following properties:

(1) The variables Z_{t+r} and a_{t+s} are independent for $s>r$.

(2)

$$E[Z_{t+m-i}\, a_{t+m-j} | Z_t, Z_{t-1}, \ldots] = \begin{cases} Z_{t+m-i}\, a_{t+m-j} & i,j \geq m \\ 0 & j<i \text{ and } j<m \\ Z_t(m-i)a_{t+m-j} & i<m \text{ and } j \geq m \\ ZA_t(m-i,m-j) & i<j<m \end{cases} \qquad (V.28)$$

where, in the last case, the variables Z_{t+r} and a_{t+s} are not independent since $s<r$. In this case, we may recursively evaluate $ZA_t(r,s)$ by substituting t+L in (V.27) for t and then multiplying a(t+L') on both sides (L,L'=1,2,...), and finally taking conditional expectation. It can be shown that

$$ZA_t(r,r) = E[a_{t+r}^2] = \sigma_a^2 \qquad (V.29)$$

As an illustration of using above algorithm for forecasting, consider the following bilinear model.

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \alpha + b_{2,1} Z_{t-2} a_{t-1} + b_{1,3} Z_{t-1} a_{t-3} + b_{2,4} Z_{t-2} a_{t-4} \tag{V.30}$$

Applying the properties just presented as well as those for linear models to (V.29), we have

$$Z_t(1) = \phi_1 Z_t + \phi_2 Z_{t-1} + \alpha + b_{2,1} Z_{t-1} a_t + b_{1,3} Z_t a_{t-2} + b_{2,4} Z_{t-1} a_{t-3}$$
$$Z_t(2) = \phi_1 Z_t(1) + \phi_2 Z_t + \alpha + b_{1,3} Z_t(1) a_{t-1} + b_{2,4} Z_t a_{t-2}$$
$$Z_t(3) = \phi_1 Z_t(2) + \phi_2 Z_t(1) + \alpha + b_{1,3} Z_t(2) a_t + b_{2,4} Z_t(1) a_{t-1}$$
$$Z_t(4) = \phi_1 Z_t(3) + \phi_2 Z_t(2) + \alpha + b_{1,3} ZA_t(3,1) + b_{2,4} Z_t(2) a_t$$
$$Z_t(5) = \phi_1 Z_t(4) + \phi_2 Z_t(3) + \alpha + b_{1,3} ZA_t(4,2) + b_{2,4} ZA_t(3,1)$$
$$\vdots$$

To find $ZA_t(3,1)$, $ZA_t(4,2)$, . . . , substitute t+2 for t in (V.30) and then multiply both sides by a_{t+1}, we obtain

$$Z_{t+2} a_{t+1} = \phi_1 Z_{t+1} a_{t+1} + \phi_2 Z_t a_{t+1} + \alpha a_{t+1} + b_{2,1} Z_t a_{t+1} a_{t+1} + b_{1,3} Z_{t+1} a_{t-1} a_{t+1} + b_{2,4} Z_t a_{t-2} a_{t+1}$$

Taking the conditional expectations on both sides leads to

$$ZA_t(2,1) = \phi_1 \sigma_a^2 + b_{2,1} Z_t \sigma_a^2 + b_{1,3} a_{t-1} \sigma_a^2$$

Similarly,

$$ZA_t(3,1) = \phi_1 ZA_t(2,1) + \phi_2\sigma_a^2 + b_{1,3}ZA_t(2,1)a_t + b_{2,4}a_{t-1}\sigma_a^2$$

$$ZA_t(3,2) = \phi_1\sigma_a^2 + b_{2,1}Z_t(1)\sigma_a^2 + b_{1,3}a_t\sigma_a^2$$

$$ZA_t(4,2) = \phi_1 ZA_t(3,2) + \phi_2\sigma_a^2 + b_{2,4}a_t\sigma_a^2$$

.

.

.

Simulation Results

Using the identification procedure presented in previous sections, 4 seasonal bilinear models were developed based on the same 4 week hourly load periods as for the ARIMA models. These models are summarized as follows:

Summer:

$$(1-.24B-.19B^2)(1-.39B^{168})W_t = (1-.74B^{24}-.10B^{48})a_t + W_{t-168}(-.0010B-.0004B^2-.0027B^3)a_t \quad \text{(V.31)}$$

Fall:

$$(1-.06B)(1-.46B^{168})W_t = (1-.63B^{24}-.21B^{48})a_t + W_{t-168}(-.0015B^{24}-.0007B^{48}+.0050B^{72}+.0022B^{96})a_t \quad \text{(V.32)}$$

Winter:

$$(1-.14B)(1-.68B^{168})W_t = (1-.62B^{24}-.14B^{48})a_t + W_{t-168}(-.0015B+.0007B^2-.0028B^3)a_t \quad \text{(V.33)}$$

Spring:

$$(1-.21B)(1-.50B^{168})W_t = (1-.53B^{24}-.16B^{48})a_t + W_{t-168}(-.0033B^{24}-.0017B^{48})a_t \quad \text{(V.34)}$$

Each model above corresponds to the one which has the minimum AIC value in the range of order values with $0 \le m \le 5$, $0 \le k \le 5$ and selected combinations of m', m", k' and k" for that particular season of the year 1983. In the identification stage, we used the ARIMA models developed in chapter II as the best ARMA models. Thus, the searching was actually over the combinations of m, m', m" and k, k' and k". Since the values of m" and k" relate to the delay of multiple of 168, m" and k" are set to 1 if needed to keep a reasonable size of data base. The initial values for each parameter were set to be zeros.

It is interesting to note that for each model above they all have one weekly term W_{t-168} and may or may not have daily terms a_{t-24} in the bilinear part, indicating the usefulness of the seasonal extension of the conventional bilinear models. Another interesting feature is that the bilinear coefficients are small, but their inclusion has a significant effect on the results of both the residual variance and the AIC value as shown in Table X.

To test a_t's independency, the W values for each bilinear model are calculated using (V.24). As seen from Table XI, these W values clearly indicate that a_t's are independent. A whiteness test for a_t gives no indication of model inadequacy also.

TABLE X

COMPARISON OF RESIDUAL VARIANCES AND AIC VALUES FOR ARIMA AND BILINEAR MODELS

Season	ARIMA		Bilinear	
	σ_a^2	AIC	σ_a^2	AIC
Summer	628.72	3083.64	610.99	3075.99
Fall	505.98	2984.26	486.02	2973.03
Winter	419.58	2894.77	410.35	2890.13
Spring	545.88	3020.55	530.42	3010.81

TABLE XI

THE W VALUES FOR THE BILINEAR MODELS

Season	|W|	5% upper point of N(0,1)
Summer	1.84	1.95
Fall	1.70	1.95
Winter	1.66	1.95
Spring	1.63	1.95

We now consider forecasting using the bilinear models (V.31)-(V.34). Since each model includes some periodic bilinear terms, this makes it possible to avoid the evaluation of expressions such as $E[Z_{t+r}a_{t+s}| Z_t, Z_{t-1}, \ldots, Z_{t-n}]$ for $r>s>0$ and, thus, extremely simplifies the forecasting process.

For the same three week forecasting periods as for the ARIMA models, 1 to 24 hour ahead load forecasts for the stationary series $\{W_t\}$ are first calculated for each of the above models. Then the forecasts for the original data $\{Z_t\}$ are found using the relationship between Z_t and W_t given by

$$\begin{aligned} W_t &= \nabla_1\nabla_{24}Z_t \\ &= (1 - B)(1 - B^{24})Z_t \\ &= (1 - B - B^{24} + B^{25})Z_t \\ &= Z_t - Z_{t-1} - Z_{t-24} + Z_{t-25} \end{aligned}$$

For example,

$$\begin{aligned} Z_t(1) &= W_t(1) + Z_t + Z_{t-23} - Z_{t-24} \\ Z_t(2) &= W_t(2) + Z_t(1) + Z_{t-22} - Z_{t-23} \\ Z_t(3) &= W_t(3) + Z_t(2) + Z_{t-21} - Z_{t-22} \end{aligned}$$

and so on. Table XII shows the comparison of forecasting errors for the ARIMA and bilinear models.

TABLE XII

COMPARISON OF FORECASTING ERRORS FOR THE ARIMA AND BILINEAR MODELS (1983) AVERAGE ABSOLUTE % ERROR

Season	Forecasting Period	ARIMA	Bilinear
Summer	July 29-Aug.17,1983	4.17	4.38
Fall	Oct.21-Nov.9,1983	4.68	4.26
Winter	Jan.21-Feb.9,1983	3.85	3.79
Spring	April 22-May 11,1983	5.24	4.71

From this table and Table X, it is clear that the bilinear model provides a better description of the relationship between the power load and the white noise "input" than the ARIMA models and, thus, improves the forecasting ability.

A combination of bilinear models and the standard TF models discussed in chapter III is also considered. These combined TF-Bilinear models are given by

Summer:

$$Z_t = \frac{4.32 + 2.26B}{1 - .70B + .104B^2} T_t + \frac{(1 - .73B^{24} - .111B^{48})a_t + BL1}{(1 - .13B - .17B^2)(1 - .469B^{168})\nabla_1 \nabla_{24}}$$

$$BL1 = \nabla_1 \nabla_{24} Z_{t-168}(-.0013B - .0016B^2 - .0017B^3)a_t$$

Fall:

$$Z_t = \frac{-.019+.466B}{1+.985B+.884B^2} T_t + \frac{(1-.612B^{24}-.088B^{48}-.167B^{72})a_t + BL2}{(1-.053B)(1-.479B^{168})\nabla_1\nabla_{24}}$$

$$BL2 = \nabla_1\nabla_{24}Z_{t-168}(-.00157B^{24}-.00062B^{48}+.0054B^{72}+.00289B^{96})a_t$$

Winter:

$$Z_t = \frac{-1.615-1.24B}{1-.587B+.148B^2} T_t + \frac{(1-.66B^{24}-.076B^{48})a_t + BL3}{(1-.042B)(1-.744B^{168})\nabla_1\nabla_{24}}$$

$$BL3 = \nabla_1\nabla_{24}Z_{t-168}(-.0018B+.0014B^2-.0025B^3)a_t$$

Spring:

$$Z_t = \frac{-.679+.145B}{1-.999B+.328B^2} T_t + \frac{(1-.531B^{24}-.154B^{48})a_t + BL4}{(1-.190B)(1-.515B^{168})\nabla_1\nabla_{24}}$$

$$BL4 = \nabla_1\nabla_{24}Z_{t-168}(-.0035B^{24}-.0017B^{48})a_t$$

Table XIII illustrates the results of residual variances and AIC values for the standard TF models and the combined TF-Bilinear models. Table XIV shows the comparison of forecasting errors for the linear and bilinear models. From these tables, we see that the bilinear models indeed have potential ability to improve the forecasting performance in view of the fact of smaller AIC values than their corresponding linear models. The reason that no significant forecasting improvement was found from our simulation results may be due to the limited number of samples. In a long range, the bilinear models should provide better results than the

ARIMA and TF models.

TABLE XIII

COMPARISON OF RESIDUAL VARIANCES AND AIC VALUES FOR THE TF AND TF-BILINEAR MODELS

Season	Transfer Function		TF-Bilinear	
	σ_a^2	AIC	σ_a^2	AIC
Summer	523	2991.71	509	2984.46
Fall	489	2965.56	471	2955.48
Winter	391	2857.10	384	2854.83
Spring	541	3011.67	528	3004.47

TABLE XIV

COMPARISON OF FORECASTING ERRORS FOR THE LINEAR AND BILINEAR MODELS AVERAGE ABSOLUTE % ERROR

Season	ARIMA	Bilinear	TF	TF-Bilinear
Summer	4.17	4.38	3.82	4.45
Fall	4.68	4.26	4.49	4.10
Winter	3.85	3.79	3.35	2.91
Spring	5.24	4.71	5.53	5.08

In summary, a general bilinear time series model was proposed and applied to the STLF with the goal to exploit the nonlinearity inherent in the power load. It is found that the proposed bilinear models with the ability to model both nonlinearity as well as multiperiodicity provide a very useful model class for general applications. In addition, using the modified bilinear model, the number of parameters to be estimated may be reduced significantly comparing with the conventional bilinear models. This will be very helpful for both estimation and forecasting procedures.

Using the proposed bilinear model, 4 regular bilinear models and 4 combined TF-Bilinear models, which correspond to spring, summer, fall and winter seasons, were developed. The forecasting ability of these bilinear models was then compared with that for the ARIMA and TF models.

In the next chapter, we investigate another type of nonlinear model which is particularly suitable for describing the relationship between load and temperature.

CHAPTER VI

HAMMERSTEIN NONLINEAR MODELS

In the last chapter we presented a bilinear model to describe the relationship between the power load and the white noise "input". The purpose is to exploit the overall nonlinearity inherent in the power load process. In this chapter we intend to further investigate the nonlinear load-temperature relationship discussed in chapter III. Recall that the nonlinear function described in chapter III was obtained by fitting to the data using a regression method. This method, however, produced an averaged result and, thus, could not accurately track the slowly-changing characteristics of the load-temperature relationship. A nonlinear model called the Hammerstein model is presented in this chapter and is expected to alleviate the problem encountered earlier and, thus, improve the forecasting performance.

The Hammerstein Model

The Hammerstein nonlinear model was originally considered by Narendra and Gallman [21] and developed by Chang, Haist, Gallman and Greblicki [22]-[25]. Examples of using this model for engineering application can be found in [5],[26]. Figure 29 illustrates the nonlinear Hammerstein model structure with correlated noise in the

output.

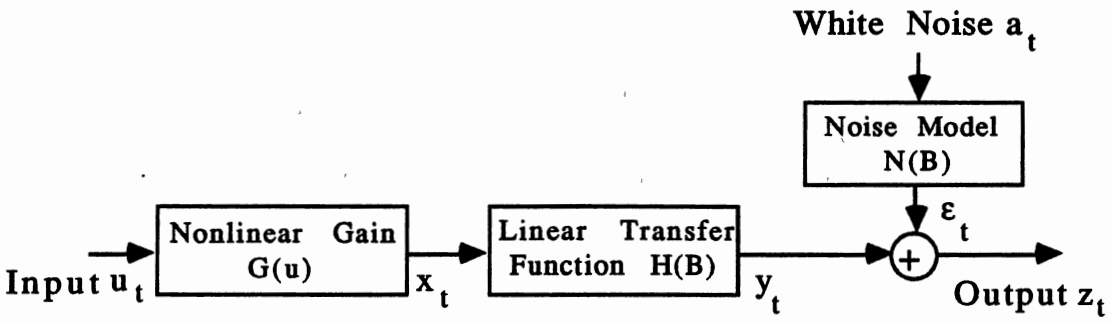

Figure 29. Hammerstein Model for the STLF

Input u_t and output z_t are assumed to be stationary. For our temperature T_t and power load Z_t, we have

$$u_t = \nabla_1 \nabla_{24} T_t \quad \text{and} \quad z_t = \nabla_1 \nabla_{24} Z_t \tag{VI.1}$$

Mathematically, the model can be expressed as

$$z_t = \frac{\omega_0 + \omega_1 B^{-1} + \ldots + \omega_s B^{-s}}{1 + \delta_1 B^{-1} + \ldots + \delta_r B^{-r}} (\gamma_0 + \gamma_1 u_t + \ldots + \gamma_k u_t^k) + \frac{1 + \theta_1 B^{-1} + \ldots + \theta_q B^{-q}}{1 + \phi_1 B^{-1} + \ldots + \phi_p B^{-p}} a_t \tag{VI.2}$$

Clearly, the nonlinear Hammerstein model structure is similar to that of simple nonlinear model described in chapter III. This structural

similarity allows us to adopt the orders of those models in (VI.2). Accordingly, all we need next is the estimates of the γ's as well as the other estimates of parameters in (VI.2). In next section, we will derive a systematic algorithm for these estimates and show that the γ's so obtained are time-varying least squares estimates instead of fixed ones.

The basic idea of the Hammerstein model is that the parameters are to be estimated as a whole instead of separately. This allows us to obtain a function which can change with the seasons, rather than one which is fixed. This should lead to an improved description of the load-temperature relationship.

The Identification Algorithm

Consider the first term of (VI.2)

$$y_t = \frac{\omega_0 + \omega_1 B^{-1} + \ldots + \omega_s B^{-s}}{1 + \delta_1 B^{-1} + \ldots + \delta_r B^{-r}} (\gamma_0 + \gamma_1 u_t + \ldots + \gamma_k u_t^k) \qquad \text{(VI.3)}$$

By expressing y_t by its available past values and input u_t's, (VI.3) becomes

$$y_t = \alpha + \sum_{i=1}^{k} M_i(B) u_t^i - \sum_{j=1}^{r} \delta_j y_{t-j} \qquad \text{(VI.4)}$$

where $\alpha = (\omega_0 + \omega_1 + \ldots + \omega_s)\gamma_0$

$M_i(B) = (\omega_0 + \omega_1 B^{-1} + \ldots + \omega_s B^{-s})\gamma_i$

Similarly, the second term of (VI.2) can be expressed as

$$\varepsilon_t = \sum_{i=1}^{q} \theta_i a_{t-i} - \sum_{j=1}^{p} \phi_j \varepsilon_{t-j} + a_t \qquad \text{(VI.5)}$$

Define the following k(s+1)+r+p+q+1 column vectors

$$q(t)=[u_t,\ldots,u_{t-s},\ldots,u_t^k,\ldots,u_{t-s}^k,y_{t-1},\ldots,y_{t-r},$$
$$a_{t-1},\ldots,a_{t-q},\varepsilon_{t-1},\ldots,\varepsilon_{t-p},1]^T$$
$$\mathbf{h} = [\omega_0\gamma_1,\omega_1\gamma_1,\ldots,\omega_s\gamma_1,\ldots,\omega_0\gamma_k,\omega_1\gamma_k,\ldots,\omega_s\gamma_k,-\delta_1,\ldots,-\delta_r,$$
$$\theta_1,\ldots,\theta_q,-\phi_1,\ldots,-\phi_p,\alpha]^T \qquad \text{(VI.6)}$$

Where T denotes the transpose. The system output is then given by

$$z_t = y_t + \varepsilon_t = q(t)^T\mathbf{h} + a_t \qquad \text{(VI.7)}$$

By minimizing the sum of squares of the errors a_t

$$S(\mathbf{h}) = \sum_{t=1}^{N} a_t^2 \qquad \text{(VI.8)}$$

where N is the data record length, an efficient estimates of the parameters are obtained by

$$\mathbf{h} = [\sum_{t=1}^{N} q(t)q(t)^T]^{-1} \sum_{t=1}^{N} q(t)z_t \qquad \text{(VI.9)}$$

where the vector q(t) may be constructed by a proceduce which will be presented shortly. The major computation of (VI.9) is the inversion of a large (k(s+1)+r+p+q+1) x (k(s+1)+r+p+q+1) matrix.

Alternatively, the recursive least square (RLS) algorithm [27]

may be used as an adaptive method for the parameter estimation. The RLS algorithm may be summarized as follows:

$$\mathbf{h}(t+1)=\mathbf{h}(t)+\frac{p(t)q(t+1)a_{t+1}}{\lambda+q(t+1)^{T}p(t)q(t+1)} \qquad \text{(VI.10)}$$

where

$$p(t+1)=\frac{1}{\lambda}(p(t)-\frac{p(t)q(t+1)q(t+1)^{T}p(t)}{\lambda+q(t+1)^{T}p(t)q(t+1)}) \qquad \text{(VI.11)}$$

where λ is a forgetting factor between 0 and 1, and p(t) is a gain matrix with initial value $p(0)=\tau I$, where τ is a positive number, say 100, and I an identity matrix.

Before using (VI.9) or (VI.10) and (VI.11), the unknown sequences $\{y_t\}$, $\{\varepsilon_t\}$ and $\{a_t\}$ must be specified. The following procedure may be employed for this purpose:

1. Consider a noise free system as shown in Figure 30.

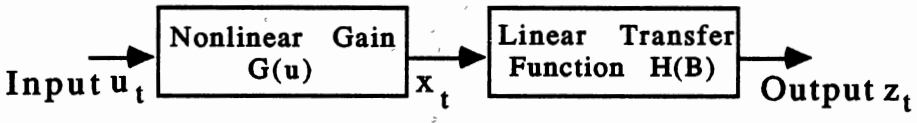

Figure 30. Hammerstein Model of a Noise Free System

In this case, (VI.6) becomes

$$q(t)=[u_t,\ldots,u_{t-s},\ldots,u_t^k,\ldots,u_{t-s}^k,z_{t-1},\ldots,z_{t-r},1]^{T}$$

$$\mathbf{h} = [\omega_0\gamma_1,\omega_1\gamma_1,\ldots,\omega_s\gamma_1,\ldots,\omega_0\gamma_k,\omega_1\gamma_k,\ldots,\omega_s\gamma_k,-\delta_1,\ldots,-\delta_r,\alpha]^T \qquad (VI.12)$$

where $\{y_{t-j}\}$ are replaced by actual output sequence $\{z_{t-j}\}$. Since all the elements of q(t) are now available, using (VI.9) and (VI.12) we obtain the initial estimate of the parameters in the deterministic portion of the noise correlated Hammerstein model.

2. With the initial estimates obtained in step 1, the estimates of the sequence $\{y_t\}$ can be generated and the sequence $\{\varepsilon_t\}$ can be found by

$$\varepsilon_t = z_t - y_t \qquad (VI.13)$$

3. Once $\{\varepsilon_t\}$ is known, we can estimate the parameters in the noise model as we did in earlier chapters and then find the estimates of the sequence $\{a_t\}$.

4. With these sequences and initial parameter values, improved estimates of the parameters can be found from (VI.9) or (VI.10). These improved estimates produce another sequence $\{y_t\}$ and sequence $\{\varepsilon_t\}$. At this stage, however, the sequence $\{a_t\}$ may be generated directly by

$$a_t = z_t - q(t)^T\mathbf{h} \qquad (VI.14)$$

Steps 2-4 are continued until some specified criterion is obtained.

Remarks:

1. There is some redundancy in estimating the parameters of ω's and γ's because s+k+2 parameters need to be determined for ω's and γ's but (s+1)k+1 parameters are actually estimated as shown in

(VI.6).

Without loss of generality, (VI.3) can be rearranged as

$$y_t = \frac{1+\omega_1' B^{-1}+\ldots+\omega_s' B^{-s}}{1+\delta_1 B^{-1}+\ldots+\delta_r B^{-r}}(\gamma_0' + \gamma_1' u_t + \ldots + \gamma_k' u_t^k) \qquad \text{(VI.15)}$$

where

$$\omega_i' = \frac{\omega_i}{\omega_0} \qquad i=1,2,\ldots,s$$

$$\gamma_j' = \omega_0 \gamma_j \qquad j=1,2,\ldots,k \qquad \text{(VI.16)}$$

Clearly, γ_j' is the direct estimate of (VI.6) while

$$\omega_i' = \frac{h_{(s+1)(L-1)+i+1}}{h_{(s+1)(L-1)+1}}, \qquad i=1,2,\ldots,s; L=1,2,\ldots,k \qquad \text{(VI.17)}$$

From these k set of ω's, we choose the set which has the minimum root mean square error. Then γ_0' can be found by

$$\gamma_0' = \alpha/(1+\omega_1'+\ldots+\omega_s') \qquad \text{(VI.18)}$$

2. Consider a typical example where the orders of (VI.2) are as follows:

$$s = 1,\ r = 2,\ k = 3,\ p = 4,\ q = 2$$

For this example, the number of parameters to be determined is

$$k(s+1) + r + p + q + 1 = 15$$

This requires the inversion of a 15x15 matrix. This seems to add a

large computational burden to the algorithm. However, reference [23] showed that a matrix of that size is adequate for most engineering systems and can be inverted without a real problem.

3. Since the power load is a slowly-changing process, updating parameters every hour when a new hourly reading is available may not be necessary. In this case, a batching method of modified Newton-Raphson algorithm discussed in chapter V is found extremely efficient. To calculate the derivatives numerically, the residuals $\{a_t\}$ of the Hammerstein model for a given set of estimates of parameters may be found by the following stages:

(1)

$$x_t = \nabla_1 \nabla_{24} (\gamma_0' + \gamma_1' T_t + \ldots + \gamma_k' T_t^k)$$

(2)

$$y_t = \frac{1 + \omega_1' B^{-1} + \ldots + \omega_s' B^{-s}}{1 + \delta_1 B^{-1} + \ldots + \delta_r B^{-r}} x_t$$

(3)

$$\varepsilon_t = z_t - y_t$$

(4)

$$a_t = (1 + \phi_1 B^{-1} + \ldots + \phi_p B^{-p})\varepsilon_t - (\theta_1 B^{-1} + \ldots + \theta_q B^{-q}) a_t$$

With these residuals, it is easy to estimate the gradient vector G and Hessian matrix H and, thus, update parameter vector **h**.

Once the parameters of (VI.2) are determined, the diagnostic checks and forecasts follow exactly the same procedure as for the simple nonlinear model, which was described in chapter III.

Simulation Results

For the purpose of comparison, the same four week load and temperature readings as for linear TF models are used to develop the corresponding nonlinear Hammerstein models. Since the Hammerstein models have a structure similar to the simple extended nonlinear models, which were presented in chapter III, the parameters of the simple nonlinear models were used as good initial values for the Hammerstein model development. Using the identification procedure presented above, 4 Hammerstein models are developed and summarized as follows:

Summer:

$$Z_t = \frac{1+.629B}{1-.633B+.086B^2}(279.00-157.72T_t+1.83T_t^2-.0069T_t^3) + \frac{1-.716B^{24}-.103B^{48}}{(1-.10B-.162B^2)(1-.504B^{168})\nabla_1\nabla_{24}}a_t \qquad \text{(VI.19)}$$

Fall:

$$Z_t = \frac{1-.477B}{1-.996B+.193B^2}(381.25+14.20T_t-.38T_t^2+.0028T_t^3) + \frac{1-.750B^{24}+.127B^{48}-.230B^{72}}{(1+.091B)(1-.704B^{168})\nabla_1\nabla_{24}}a_t \qquad \text{(VI.20)}$$

Winter:

$$Z_t = \frac{1+.535B}{1-.589B+.184B^2}(238.80+1.076T_t-.098T_t^2+.0009T_t^3) + \frac{1-.643B^{24}-.064B^{48}}{(1-.024B)(1-.758B^{168})\nabla_1\nabla_{24}}a_t \qquad \text{(VI.21)}$$

Spring:

$$Z_t = \frac{1+.842B}{1-.108B-.332B^2}(243.00-6.246T_t+.027T_t^2+.0002T_t^3) + \frac{1-.547B^{24}-.130B^{48}}{(1-.095B)(1-.612B^{168})\nabla_1\nabla_{24}}a_t \qquad \text{(VI.22)}$$

Unlike the simple nonlinear models, which use the same nonlinear function to describe the load-temperature relationship for a whole year, the Hammerstein model produces 4 different nonlinear functions, each for one season of the year and, thus, may track the changing characteristics of the load-temperature relationship. These functions are plotted in Figures 31-32. Figures 31(a)-(d) illustrate the overall behaviors of the nonlinear functions fitted by the Hammerstein models. Figures 32(a)-(d) depict the details of each nonlinear function corresponding to an appropriate temperature span.

By comparing the Hammerstein models with the simple nonlinear models, we can observe not only the changes of the nonlinear function itself but also some changes of the parameters of

transfer function and noise model components. This further indicates the necessity of adjusting all parameters simultaneously. Table XV compares the forecasting results of the Hammerstein models with those for the simple nonlinear TF models.

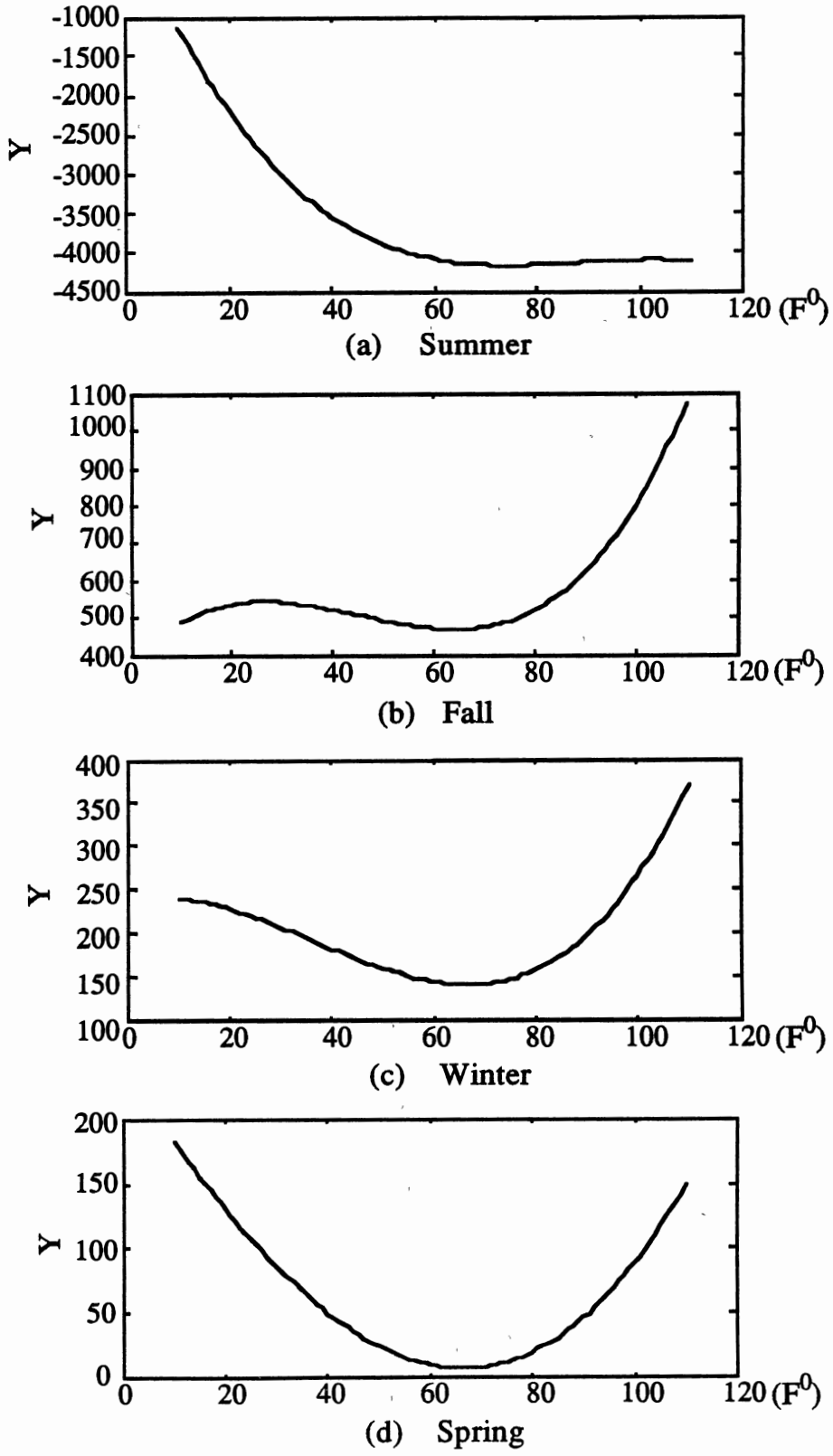

Figure 31. Nonlinear Functions from Hammerstein Models

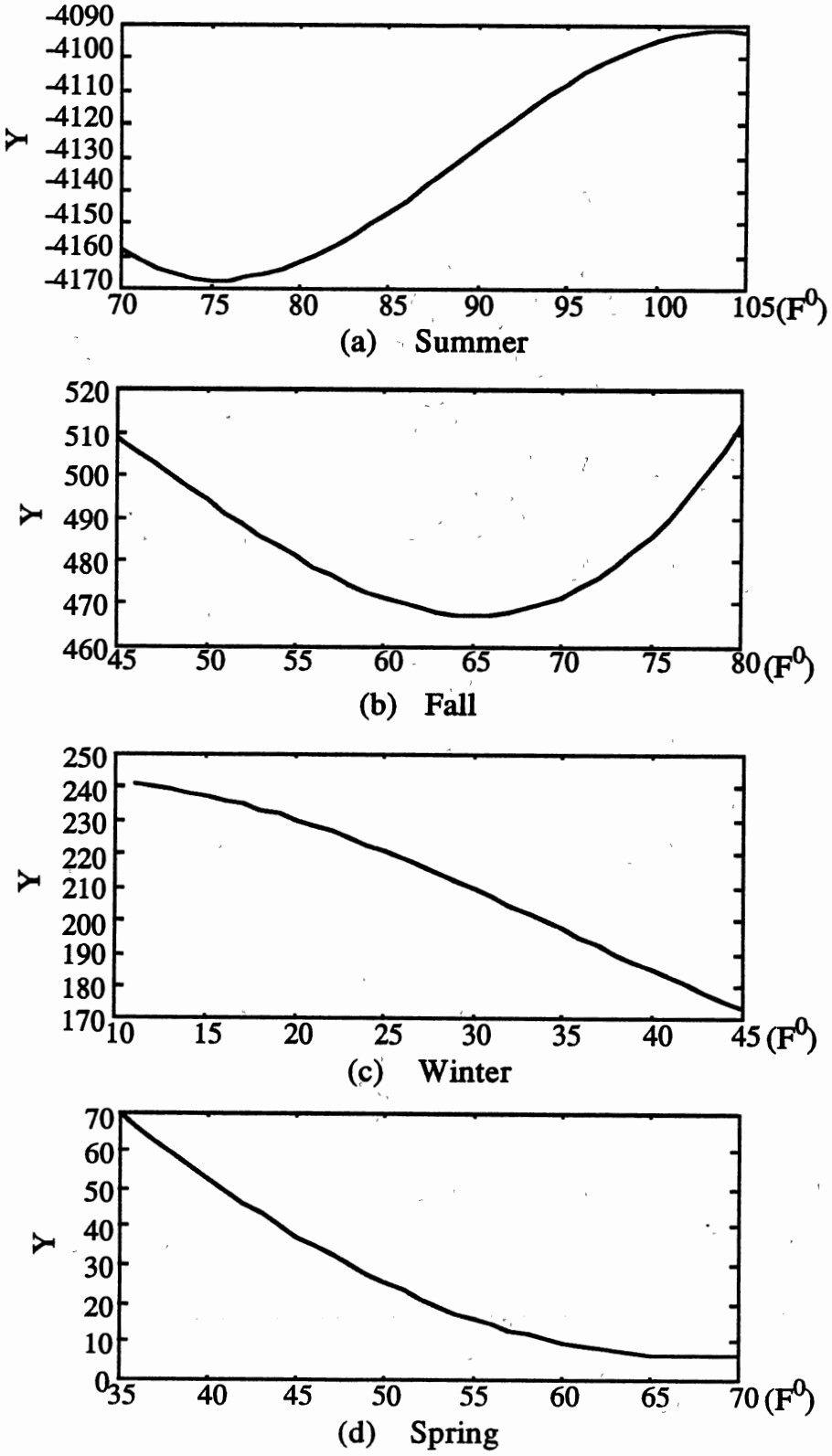

Figure 32. Plots of Figure 31 with Proper Temperature Span

TABLE XV

COMPARISON OF FORECASTING ERRORS FOR THE SIMPLE NONLINEAR TF AND HAMMERSTEIN MODELS (1983) AVERAGE ABSOLUTE % ERROR

Season	Nonlinear TF	Hammerstein
Summer	3.55	3.41
Fall	3.41	3.42
Winter	2.84	2.85
Spring	5.16	4.54

TABLE XVI

COMPARISON OF FORECASTING ERRORS FOR SIX DIFFERENT MODELS AVERAGE % ERROR OVER FOUR SEASONS (1983)

ARIMA	Bilinear	TF	TF-Bilinear	Nonlinear TF	Hammerstein
4.49	4.29	4.30	4.14	3.74	3.56

Averaged over the four seasons, the Hammerstein model had 3.56 average percent error, comparing with 3.74 average percent error for the simple nonlinear TF model. This translates into improvement of 5%. Table XVI compares the average percent errors over the four seasons for each forecasting method discussed in the paper.

From Table XVI, it is clear that the Hammerstein model, with the ability to include temperature as an explicit input and model slowly-changing nonlinearity, gives the best result for the STLF. Now we may have the following conclusions:

1. The electric power load is a nonlinear slowly-changing stochastic process. An appropriate nonlinear model may be more accurate for the STLF than a linear one;

2. Weather variables such as temperature are very important to load forecasting. Their explicit inclusion may improve the forecasting performance significantly. Therefore, further investigation of temperature effects on load consumption should be a fruitful research area for the STLF and this forms our major topic in the next chapter.

CHAPTER VII

TEMPERATURE ENHANCED MODELS

From the previous chapters we found that temperature may influence the forecasting results significantly. Whenever temperature, as an explicit variable, is added to a model (e.g., TF model, TF-Bilinear model), it always improves the forecasting performance to some extent. This clearly indicates the possibility of improvement of forecasts by further investigating temperatue effects on load consumption.

One possible way to obtain such forecasting improvement by temperature enhancement is to utilize k^{th} step ahead forecasting errors or the sum of 1 to k step ahead forecasting errors to adjust model parameters, instead of using the one step ahead forecasting error (residual) as before. The idea behind this comes from the fact that one hour ahead (i.e., one step ahead due to hourly readings) temperature change may not have much effect on load pattern changes. This is also true for sudden temperature changes, in which case it usually takes several hours before obvious temperature effects on load begin to be effective. In other words, the traditional TF model can not make full use of temperature information.

This fact may be further viewed by figure 33, which illustrates the summer load forecastes using the standard TF model (III.27). As

is seen from the figure, the forecastes are almost always either overpredicton or underprediction of loads for several days, corresponding to decreasing or increasing trends of the actual loads, respectively. This may suggust that if model parameters could be adjusted such that no obvious trends of underprediction and overprediction exist, then we would expect better forecasting results.

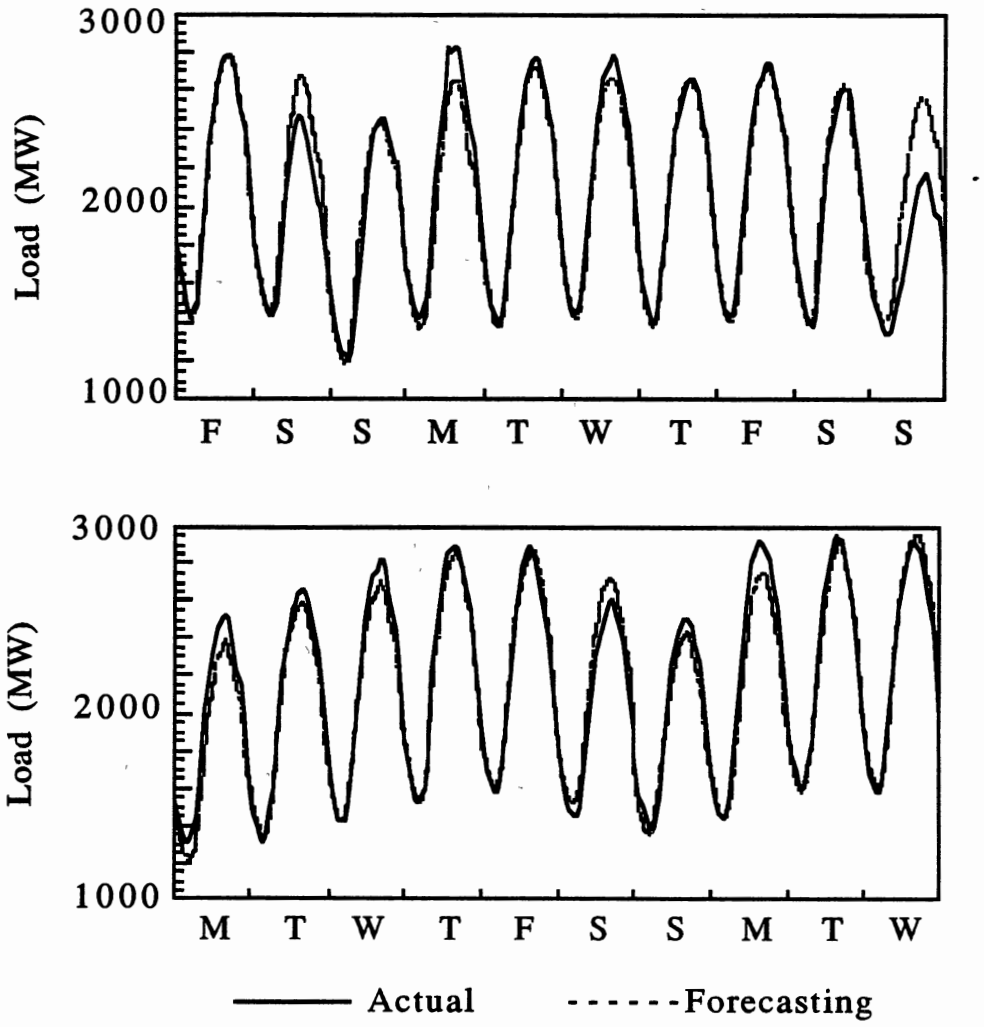

Figure 33. Summer Load Forecasts Using Model (III.27)
July 29-August 17, 1983

In this chapter, we discuss algorithms which may be used to apply the above temperature enhancement idea to the standard TF model and the Hammerstein model and, hopefully, obtain further forecasting improvement.

The Algorithm

The method used for temperature enhanced model development is the same modified Newton-Raphson algorithm as used for the bilinear models and the Hammerstein models. The only difference from the previous models is that the k^{th} step ahead or the sum of 1 to k step ahead forecasting errors are used to compute the gradient G and the Hessian matrix H instead of the residuals, which are the one step ahead forecasting errors. This is equivalent to using 1 to k step ahead temperature information as well as the load, because 1 to k step ahead temperature readings must be available to make the k^{th} step ahead forecasts and, thus, obtain the k^{th} step ahead forecasting errors.

It is clear that the temperature enhanced model may be applied to both standard TF and Hammerstein models since, in each case, we may use 1 to k hour ahead temperature information to compute the corresponding forecasts and, thus, the forecasting errors. These errors, in turn, may be used to update parameters.

When the sum of 1 to k step ahead forecasting errors is used to adjust parameters, the gradient vector G and the Hessian matrix H may be computed as follows.

let $a_{t+i} = Z_{t+i} - Z_t(i)$ represent the i^{th} step ahead forecasting

error, the sum of squares of a_{t+i} for ll≤i≤ul is given by

$$S_t = \sum_{i=ll}^{ul} a_{t+i}^2 = a_{t+ll}^2 + a_{t+ll+1}^2 + \ldots + a_{t+ul}^2, \quad 1 \le ll \le ul \le 24 \tag{VII.1}$$

where ll denotes the lower limit and ul the up limit for forecasting steps. The total sum squares of error is then given by

$$S(\mathbf{h}) = \sum_{t=\gamma}^{N} S_t = \sum_{t=\gamma}^{N} (a_{t+ll}^2 + a_{t+ll+1}^2 + \ldots + a_{t+ul}^2) \tag{VII.2}$$

Following the same discussions for Eqs (V.6)-(V.10), we have

$$\frac{\partial S(\mathbf{h})}{\partial h_i} = 2 \sum_{t=\gamma}^{N} (a_{t+ll} \frac{\partial a_{t+ll}}{\partial h_i} + \ldots + a_{t+ul} \frac{\partial a_{t+ul}}{\partial h_i}) \tag{VII.3}$$

$$\frac{\partial^2 S(\mathbf{h})}{\partial h_i \partial h_j} \approx 2 \sum_{t=\gamma}^{N} (\frac{\partial a_{t+ll}}{\partial h_i} \frac{\partial a_{t+ll}}{\partial h_j} + \ldots + \frac{\partial a_{t+ul}}{\partial h_i} \frac{\partial a_{t+ul}}{\partial h_j}) \tag{VII.4}$$

let

$$G_k = \{2 \sum_{t=\gamma}^{N} a_{t+k} \frac{\partial a_{t+k}}{\partial h_i}\} \tag{VII.5}$$

$$H_k = \{2 \sum_{t=\gamma}^{N} \frac{\partial a_{t+k}}{\partial h_i} \frac{\partial a_{t+k}}{\partial h_j}\} \tag{VII.6}$$

then

$$G = \sum_{k=ll}^{ul} G_k \quad \text{and} \quad H = \sum_{k=ll}^{ul} H_k \tag{VII.7}$$

clearly, when k=ll=ul=1, it comes back to our early algorithm.

Once the gradient vector G and the Hessian matrix H are obtained, parameter update, model diagnostic checks and forecasts follow exactly the same procedure as for the standard TF and Hammerstein models.

Simulation Results

Again, the same four week load and temperature readings as before are used for model development. Then, each model developed is applied to make 1 to 24 hours ahead load forecasts for a three week period. Since the goal of this chapter is to investigate the enhanced temperature effects on the STLF, the same structures of the standard TF and the Hammerstein models developed in earlier chapters are used to build models.

Table XVII illustrates the forecasting results using the temperature enhanced TF models obtained by adjusting parameters through 1, 6, 12 and 18 hour ahead forecasting errors, respectively. From this table, it is easy to see that the temperature enhanced TF models do provide better forecasting performance.

Table XVIII shows the forecasting results for the temperature enhanced Hammerstein models. Comparing with Table XVII, the Hammerstein models, even without temperature enhancement, are still better than the TF models, further indicating the importance of the nonlinearity involved in the STLF application. On the other hand,

when the temperature enhancement technique is considered, an obvious forecasting improvement is achieved.

TABLE XVII

FORECASTING ERRORS USING TEMPERATURE ENHANCED TF MODELS (1983)
AVE. ABS. % ERROR

Season \ Step	1	6	12	18
Summer	3.82	3.19	2.78	2.66
Fall	4.49	4.33	3.92	3.91
Winter	3.35	2.52	2.63	2.78
Spring	5.53	5.35	5.30	5.50
Ave. Error	4.30	3.85	3.66	3.71

TABLE XVIII

FORECASTING ERROR USING TEMPERATURE ENHANCED HAMM. MODELS (1983) AVE. ABS. % ERROR

Season \ Step	1	6	12	18
Summer	3.41	2.96	2.50	2.92
Fall	3.42	2.90	3.31	3.22
Winter	2.85	2.53	2.62	2.75
Spring	4.54	3.92	4.18	3.86
Ave. Error	3.56	3.08	3.15	3.19

In order to study more details about the temperature enhanced models, the plots of Summer load and temperature readings for the forecasting period are shown by figures 34-35. The dash line in figure 35 represents the forecasts using the temperature enhanced Hammerstein model developed by adjusting parameters through the twelfth step ahead forecasting errors, which is given by

$$Z_t = \frac{1-.602B}{1-.999B+.073B^2}(-17.967-5.583T_t-.08449T_t^2+.001195T_t^3) + \frac{1-.560B^{24}-.266B^{48}}{(1+.299B+.637B^2)(1-.591B^{168})\nabla_1\nabla_{24}}a_t \qquad \text{(VII.8)}$$

From these plots, it is clear that the significant drop of power load on the second Sunday was mainly due to the sudden temperature

change, which jumped from 104 ^{0}F on Saturday down to 87 ^{0}F on Sunday. Remarkably, the temperature enhanced Hammerstein model can still handle such a tough situation, keeping the forecasting errors very small. In addition, by closely examining Figure 35, we may observe that the problems of underprediction and overprediction encountered earlier have been effectively alleviated also.

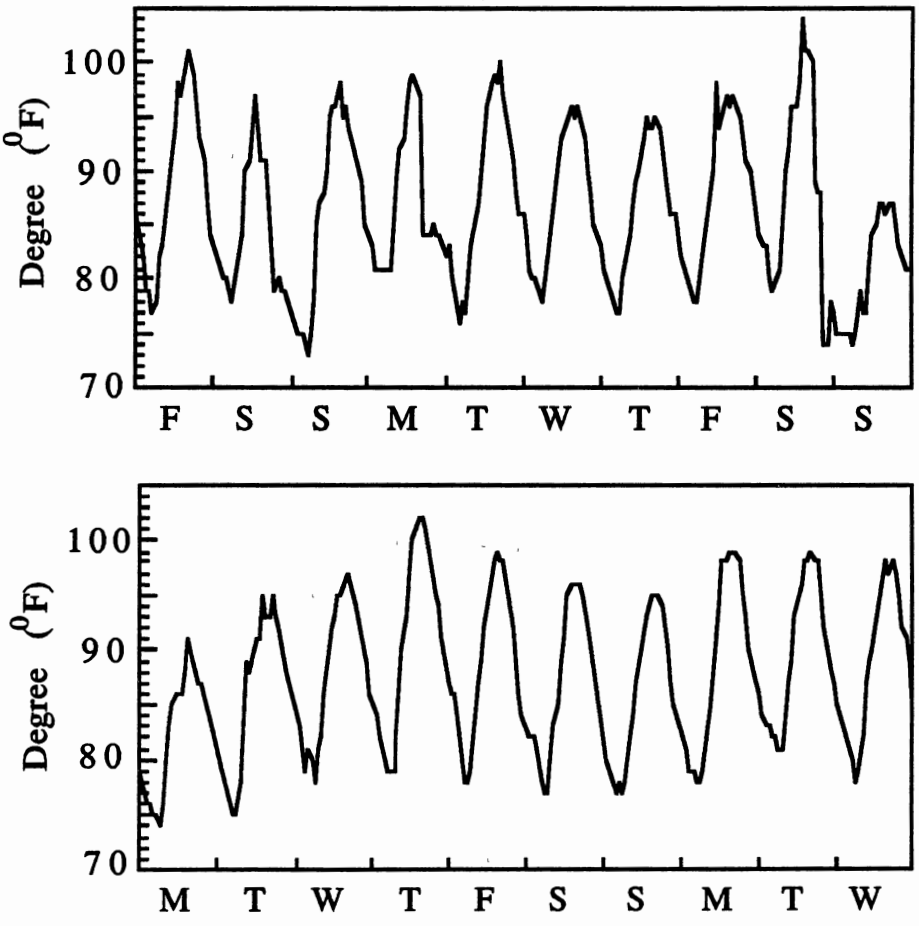

Figure 34. Summer Temperature: July 29-August 17, 1983

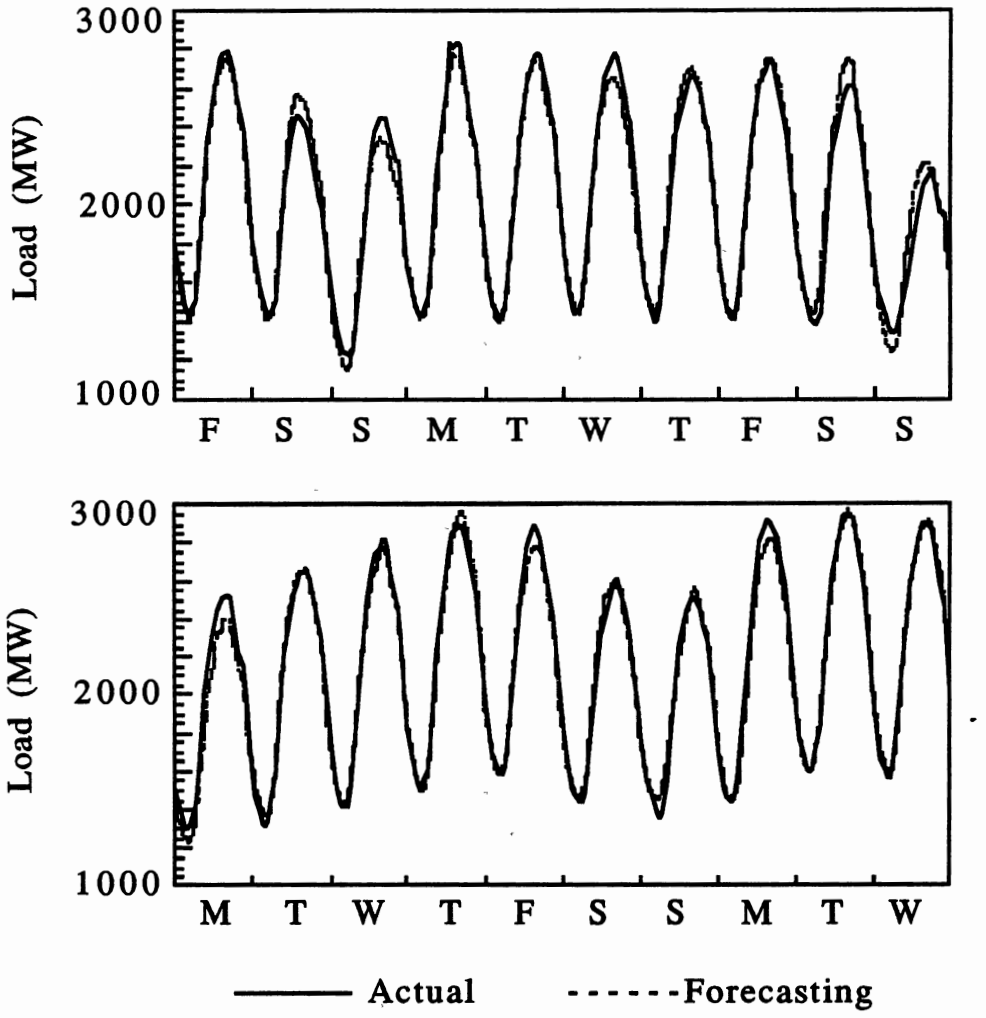

Figure 35. Summer Load Forecasts Using Model (VII.8)
July 29-August 17, 1983

Models built by adjusting parameters through the sum of 1 to k step ahead forecasting errors were also investigated. As an example, Table XIX shows the forecasting results using the temperature enhanced Hammerstein models which were obtained by adjusting paramters through the sum of 1 to 18 step ahead forecasting errors. Although no further forecasting improvement is achieved, comparing

with previous temperature enhanced Hammerstein models, these models may provide a consistant result in view of the fact that no prior knowledge of a specific number of leading steps is required in this case.

TABLE XIX

FORECASTING ERRORS USING TEMPERATURE ENHANCED HAMM. MODEL (1-18 STEPS) AVE. ABS % ERROR

Summer	Fall	Winter	Spring
2.88	3.06	2.60	3.74

For reference, some of the temperature enhanced models developed in this chapter are summarized as follows:

1. TF models using the twelfth hour ahead temperature readings

Summer:

$$Z_t = \frac{5.998 - 3.087B}{1 - .999B + .073B^2} T_t + \frac{1 - .484B^{24} - .297B^{48}}{(1 + .161B + .742B^2)(1 - .532B^{168})\nabla_1 \nabla_{24}} a_t$$

Fall:

$$Z_t = \frac{.804-.066B}{1-.968B+.609B^2} T_t + \frac{1-.288B^{24}-.281B^{48}-.236B^{72}}{(1+.768B)(1-.697B^{168})\nabla_1\nabla_{24}} a_t$$

Winter:

$$Z_t = \frac{-4.03-1.297B}{1+.163B-.367B^2} T_t + \frac{1-.733B^{24}-.141B^{48}}{(1+.825B)(1-.975B^{168})\nabla_1\nabla_{24}} a_t$$

Spring:

$$Z_t = \frac{-1.205+.765B}{1-.398B-.484B^2} T_t + \frac{1-.528B^{24}-.294B^{48}}{(1+.589B)(1-.817B^{168})\nabla_1\nabla_{24}} a_t$$

2. Hammerstein models using the sixth hour ahead temperature readings.

Summer:

$$Z_t = \frac{1+26.65B}{1-.368B+.059B^2}(-32.03T_t+.367T_t^2-.001376T_t^3) + \frac{1-.758B^{24}-.110B^{48}}{(1+.152B+.007B^2)(1-.756B^{168})\nabla_1\nabla_{24}} a_t$$

Fall:

$$Z_t = \frac{1-.508B}{1-.999B+.203B^2}(7.757+8.744T_t-.3088T_t^2+.002583T_t^3) + \frac{1-.447B^{24}-.036B^{48}-.122B^{72}}{(1+.700B)(1-.830B^{168})\nabla_1\nabla_{24}}a_t$$

Winter:

$$Z_t = \frac{1+.439B}{1-.552B+.210B^2}(13.16+1.929T_t-.1372T_t^2+.00125T_t^3) + \frac{1-.640B^{24}-.191B^{48}}{(1+.508B)(1-.950B^{168})\nabla_1\nabla_{24}}a_t$$

Spring:

$$Z_t = \frac{1+.004B}{1+.021B-.525B^2}(-20.87-6.663T_t-.0273T_t^2+.0008T_t^3) + \frac{1-.457B^{24}-.269B^{48}}{(1+.400B)(1-.854B^{168})\nabla_1\nabla_{24}}a_t$$

3. Hammerstein models using 1 to 18 hours ahead temperature readings.

Summer:

$$Z_t = \frac{1+34.44B}{1-.135B-.156B^2}(-48.85-35.00T_t+.401T_t^2-.00151T_t^3) + \frac{1-.888B^{24}+.053B^{48}}{(1+.507B+.282B^2)(1-.777B^{168})\nabla_1\nabla_{24}}a_t$$

Fall:

$$Z_t = \frac{1-.386B}{1-.998B+.217B^2}(5.089+1.842T_t-.1589T_t^2+.001607T_t^3) + \frac{1-.316B^{24}+.089B^{48}-.061B^{72}}{(1+.536B)(1-.870B^{168})\nabla_1\nabla_{24}}a_t$$

Winter:

$$Z_t = \frac{1+.046B}{1-.769B+.229B^2}(-38.3+.518T_t-.0977T_t^2+.000902T_t^3) + \frac{1-.716B^{24}-.124B^{48}}{(1+.479B)(1-.961B^{168})\nabla_1\nabla_{24}}a_t$$

Spring:

$$Z_t = \frac{1+.134B}{1-.0008B-.435B^2}(-15.61-8.084T_t-.0069T_t^2+.000755T_t^3) + \frac{1-.474B^{24}-.218B^{48}}{(1+.512B)(1-.915B^{168})\nabla_1\nabla_{24}}a_t$$

Finally, it should be noted that all the results above were obtained based on the actual temperature readings, not forecasted

ones. The reasons for that are (1) the forecasted temperature is not available to us; (2) the forecasted temperature may be assumed to be very close to actual ones for a short time period, say 1 to 24 hours ahead.

CHAPTER VIII

CONCLUSIONS

In this paper, we have reviewed the linear time series approach to short term load forecasting. This approach, which includes the ARIMA and TF models, is capable of modeling multiperiodicity and nonstationarity, as well as some weather effects such as temperature. As was shown in chapter II and chapter III, the ARIMA models and TF models were relatively easily developed using a systematic procedure and could produce good forecasting results. All these features have made the linear time series approach the most popular STLF procedure.

One disadvantage of the ARIMA and TF models is that these models can not describe the nonlinearity of the power load, which, as was demonstrated by the simple nonlinear model in chapter III and the nonlinearity test in chapter IV, seems to be an important characteristic of the load. In order to exploit the inherent nonlinearity in the power load, two different types of nonlinear models, and their corresponding identification algorithms, were proposed with the goal to further improve load forecasting performance.

The bilinear models modify the ARIMA models by adding cross terms between the output z_t and the white noise "input" a_t. This model may be considered as a description of a general nonlinear time

series.

Considering that in practice many time series present some periodicities or "seasonal" behaviors and these characteristics may be described by bilinear terms also, a modified form of the conventional bilinear model was created. It is found that the proposed bilinear model, with the ability to model both nonlinearity as well as multiperiodicity, provides a very useful model class for general applications. In addition, the modified bilinear model may reduce the number of parameters to be estimated significantly, when compared with the conventional bilinear model and, thus, provides an environment in which both estimation and forecasting procedures are much easier to implement than before.

Motivated by the performance improvement of the simple nonlinear TF model, a nonlinear Hammerstein model is presented in chapter VI. Since the parameters of the linear and nonlinear parts of the model can now be estimated systematically as a whole by the Newton-Raphson algorithm, the nonlinear Hammerstein model is able to better describe the nonlinear relationship between load and temperature for short term load forecasts.

Finally, in chapter VII, we presented a temperature enhanced model, which modifies the Hammerstein model by adjusting parameters through the k^{th} step ahead or the sum of 1 to k step ahead forecasting errors instead of residuals, which are one step ahead forecasting errors. The basic idea behind this model is based on the fact that the temperature information is very important in power load forecasting and that the better this information is used, the more accurate the load forecasts will be. Table XX summarizes

the forecasting results for various models.

TABLE XX

COMPARISON OF FORECASTING ERRORS FOR THE LINEAR AND NONLINEAR MODELS (1983) AVE. ABS. % ERROR

Season	ARIMA	TF	Simple Nonlin.	Hamm.	Temp. Enhanced Hamm. Step 6	Steps 1-18
Summer	4.17	3.82	3.55	3.41	2.96	2.88
Fall	4.68	4.49	3.41	3.42	2.90	3.06
Winter	3.85	3.35	2.84	2.85	2.53	2.60
Spring	5.24	5.53	5.16	4.54	3.92	3.74
Ave. Error	4.49	4.30	3.74	3.56	3.08	3.07

As is seen from this table, the temperature enhanced Hammerstein models obtained by adjusting parameters through the sixth step ahead and the sum of 1 to 18 step ahead forecasting errors turn out to be the best, with average absolute percent error 3.08 and 3.07 over four seasons, respectively. Comparing with the previous result, 3.74, obtained by the simple nonlinear model, this translates into improvements of 17.65% and 17.91%, respectively. If compared with the 4.49 obtained by the ARIMA models, the improvements will be 31.40% and 31.63%, respectively. Conclusively, the nonlinear

temperature enhanced Hammerstein model is the best of the models and provides an efficient procedure for the STLF application.

One topic for future research would be the development of an adaptive model for STLF. Since the power load is a slowly time-varying process, a model which can adapt itself to the changes might be used and should provide a better description of the load process. As an example of the need for model adaptation, table XXI shows the forecasting results for three 20 day periods in 1984 (winter, spring and summer) using the models above. As is seen from this table, although the temperature enhanced models give the best forecasting errors for spring and winter seasons, they produce the worst results for the summer season.

TABLE XXI

MEAN ABSOLUTE % FORECASTING ERROR (1984)

Season	ARIMA	TF	Simple Nonlin.	Hamm.	Temp. Enhanced Hamm. Step 6	Steps 1-18
Spring	4.92	5.10	4.29	4.35	4.19	4.08
Summer	5.25	3.93	4.02	3.69	4.46	6.16
Winter	4.58	3.71	2.87	2.81	2.51	2.33

By reestimating the parameters of the temperature enhanced Hammerstein models based on the load and temperature data of summer 1984, we then have 3.41% and 2.61% forecasing errors for the summer 1984, respectively. This result clearly indicates that the load and temperature patterns have changed to a certain extent since a year ago and that an adaptive model is necessary to track these changes and, thus, to obtain the most accurate forecast.

Another interesting model to be considered is the artificial neural network model, which has been reported to be able to approximate any linear/nonlinear process without imposing any functional asumption on them. Although this model has recently been studied by several authors [6], [28] and was shown to be an effective approach for STLF, we feel that it is still worth further study, at least for model comparison purposes.

REFERENCES

[1] B. Abraham and J. Ledolter, Statistical Methods for Forecasting. John Wieldy &. Sons, Inc., 1983

[2] D.D. Belik, D.J. Nelson, and D.W. Olive, "Use of the Karhunen-loeve expansion to analyze hourly load requirements for a power utility," Paper A78 225-5, presented at the IEEE Power Engineering Society Winter Meeting, New York, N.Y. Jan. Feb. 1978 (LS, TD, W, AD, EX)

[3] M.T. Hagan and S.M. Behr, "The Time Series Approach to Short Term Load Forecasting," IEEE Transactions on Power Systems, Vol 2, No. 3, August 1987, pp. 785-791.

[4] J. Toyoda, M. chen, and Y. Inouye, "An Application of State Estimation to Short-Term Load Forecasting, Part I and Part II," IEEE Trans. Power APP. Syst., Vol. PAS-89, No. 7, Sept. /Oct. 1970, pp.1678-1688

[5] Q.C. Lu, W.M. Grady, M.M. Crawford, and G.M. Anderson, "An Adaptive Nonlinear Predictor with Orthogonal Escalator Structure for short-term load forecasting," IEEE Trans. on Power system, Vol. 4, No. 1, Feb. 1989 PP.

[6] D.C. Park, M.A. El-Sharkawi, R.J. Marks II, L.E. Atlas and M.J. Damborg, "Electric Load Forecasting Using An Artificial Neural Network," IEEE Trans. on Power Systems, Vol. 6, No. 2, May 1991, pp.442-448.

[7] S. Rahman and R. Bhatnagar,"An expert system based algorithm for short term load forecasting," Paper 87 WM 082-1, presented at the IEEE Power Engineering Society Winter Meeting, New Orleans, LA, Feb. 1987. [W,DY. ARMA,AD,STD,EX]

[8] G. Gross, and F.D. Galiana,"Short-term Load Forecasting," Proceedings of the TEEE, Vol. 75, No.12, Dec. 1987, pp.1558-1573.

[9] C.E.P. Box and G.N. Jenkins, Time Series Analysis Forecasting and Control, Holden Day, San Francisco, 1970.

[10] L. Ljung and T. Soderstrom, Theory and Practice of Recursive Identification. Cambridge, MA:MIT press, 1983 [REF]

[11] T.C. Hsia, System Identification, Lexington Books, D.C. Heath and Company, Lexington, Massachusetts, Toronto, 1978

[12] M.B. Priestley, Non-linear and Non-stationary Time Series Analysis, Academic press. 1988.

[13] T.S. Rao, and M.M. Gabr, An Introduction to Bispectral Analysis and Bilinear Time Series Models, Springer-Verlag, Berlin, 1984.

[14] J.W. Van Ness, Asymptotic Normality of Bispectral Estimates. Ann. Math. Statist. 37, PP1257-1275, 1966.

[15] M.B. Priestley, Spectral Analysis and Time Series. Academic Press, London. 1981.

[16] D.R. Brillinger, An Introduction to Polyspectrum. Ann. Math. Statist. 36, pp1357-1374, 1965.

[17] T.W. Anderson, An Intro. to Multivariate Statistical Analysis, Wiley, New York, 1958.

[18] D.J. Pack, Computer programs for the analysis of univariate time series models and single input transfer function models

using the methods of Box and Jenkins. The Ohio State University, Dec. 1, 1974.

[19] C.W.J. Granger and A.P. Andersen, Introduction to Bilinear Time Series Models, Vanderhoeck and Ruprecht: Gottingen, 1978.

[20] M.B. Priestley, Nonlinear models in time series analysis, The statistician 27, 1978.

[21] K.S. Narendra and P.G. Gallman, "An iterative method for the identification of nonlinear systems using the Hammerstein model," IEEE Trans. Automat. Contr., Vol. AC-11, pp. 546-550 ,1966.

[22] F.H.I. Chang and R. Luus, "A noniterative method for identification using the Hammerstein model," IEEE Trans. Automat. Contr., vol. AC-16, pp. 464-468m 1971.

[23] N.D. Haist, F.H.I. Chang, and R. Luus, " Nonlinear identification in the presence of correlated noise using a Hammerstein model," IEEE Trans. Automat. Contr., vol. AC-18, pp. 552-555, 1973.

[24] P.G. Gallman,"A comparison of two Hammerstein model identification algorithms," IEEE Trans. Automat. Contr., vol. AC-21, pp.124-126, 1976.

[25] W. Greblicki and M. Pawlak, "Identification of discrete Hammerstein system using Kernel regression estimates," IEEE Contr., vol. AC-31, pp.74-77, 1986.

[26] T.C. Hsia and A.L. Bailey, "Learning model approach for nonlinear system identification," in 1986 IEEE System Science and Cybernetics conf. Rec., pp.228-232.

[27] C.F.N. Cowan and P.M. Grant, Adaptive Filters, Englewood Cliffs, NJ. Prentice-Nall, 1985.

[28] K. Y. Lee, Y. T. Cha, and J. H. Park, "Short-Term Load Forecasting Using an Artificial Neural Network", IEEE Trans. on Power Systems, Vol. 7, No. 1, February, 1992, pp.124-132.

VITA

Yongqing Zhang

Candidate for the Degree of

Doctor of Philosophy

Thesis: LINEAR AND NONLINEAR MODELING AND FORECASTING OF ELECTRIC POWER LOADS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Tianjin, China, January 21, 1958, the son of Zhisheng and Shuzhen Zhang.

Education: Received Bachelor of Science degree in Physics from Nankai University, China in January, 1982; received Master of Science degree from Oklahoma State University in May, 1987; completed requirements for the Doctor of Philosophy degree at Oklahoma State University in July, 1992.

Personal Experience: Assistant Electrical Engineer, Shijiazhuang Communications Lab, China, February, 1982, to December, 1985. Research Assistant, Department of Electrical Engineering, Oklahoma State University, August, 1988, to June, 1990. Production Engineer, Gardiner Communications Corporation, Dallas, Texas, July, 1990, to August, 1991. Teaching Assistant, Department of Electrical Engineering, Oklahoma State University, August, 1991, to May, 1992.