A New Algorithm for Automated Modeling of Seasonal Time Series Using Box-Jenkins Techniques

Qiang Song
R&D, RedPrairie Corporation
3905 Brookside Parkway
Alpharetta, GA 30022 U.S.A.
Email: qsong3@hotmail.com

Augustine O. Esogbue†
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205 U.S.A.
Email: aesogbue@isye.gatech.edu

Received Date, November 2006; Accepted Date, December 2007

Abstract. As an extension of a previous work by the authors (Song and Esogbue, 2006), a new algorithm for automated modeling of nonstationary seasonal time series is presented in this paper. Issues relative to the methodology for building automatically seasonal time series models and periodic time series models are addressed. This is achieved by inspecting the trend, estimating the seasonality, determining the orders of the model, and estimating the parameters. As in our previous work, the major instruments used in the model identification process are correlograms of the modeling errors while the least square method is used for parameter estimation. We provide numerical illustrations of the performance of the new algorithms with respect to building both seasonal time series and periodic time series models. Additionally, we consider forecasting and exercise the models on some sample time series problems found in the literature as well as real life problems drawn from the retail industry. In each instance, the models are built automatically avoiding the necessity of any human intervention.

Keywords: Automated Model Building Algorithms, Box-Jenkins Modeling Technique, Correlograms, Forecasting of Seasonal Time Series, Periodic Time Series Models, Seasonal Time Series, and Seasonal Time Series Models.

1. INTRODUCTION

This paper presents new algorithms for automatic building of seasonal and periodical models for seasonal time series using Box-Jenkins modeling techniques.

As is known, the major vehicles employed in Box-Jenkins time series modeling technique are the correlograms of the data which provide useful structural information of a time series. Despite its conceptual clarity and simplicity, this technique imposes a heavy burden on modelers who must visually inspect the correlogram plots and hypothesize on the orders and the types of the models for the time series of interest. Once a tentative model structure is determined with the visual inspection of the correlograms, parameters of the model are then estimated. Whether or not the identified model is proper is determined by the modeling error characteristics. In general, if the modeling errors behave strongly like a white noise process, then the model is considered to be proper. Otherwise, different model structures must be explored. Evidently, this modeling technique poses some difficulties in applications, and could be even harder to automate. For this reason, Box-Jenkins modeling technique has not enjoyed wide applications that it deserves (Talluri and van Ryzin, 2004; Liu, 2006, p. 4.1). In an automated modeling process, the major difficulty with Box-Jenkins modeling technique is the visual inspection of the correlograms. Unless an algorithm is developed to inspect the correlograms automatically by a computer system instead of by human beings, it is almost impossible to automate the modeling process satisfactorily and economically in line with the Box-

† : Corresponding Author

Jenkins technique. Although a number of different automated modeling algorithms have been reported in the literature (Reilly, 1980, 1987; Wu and Pandit, 1979), they have been shown to possess certain deficiencies as discussed for example in (Song and Esogbue, 2006). Most recently, Song and Esogbue (2006) have developed an algorithm to automate the modeling process of stationary ARMA time series using Box-Jenkins modeling techniques. It is believed that this newly developed algorithm rectifies, to a large extent, the key issues and concerns with existing algorithms while avoiding their major drawbacks. For notational convenience, we tentatively name this newly reported algorithm, the S-E algorithm.

A natural extension of the S-E algorithm is the automation of the modeling process of seasonal time series. In general, seasonal time series are classified as stationary and nonstationary seasonal time series. In this paper, we only consider a special class of nonstationary seasonal time series which may contain a deterministic local trend but have a constant variance. We focus our attention on the modeling of such seasonal time series. Specifically, we develop automated modeling algorithms for such seasonal time series via an extension of the S-E algorithm.

This paper is organized as follows. In Section 2, the literature of seasonal time series modeling algorithms is first reviewed followed by a brief review of the S-E algorithm in Section 3. The major result of this paper is presented in Section 4 where the algorithms for seasonal time series modeling are discussed in detail. Extensions of the main results to periodic time series modeling are presented in Section 5. Numerical illustrations which provide and compare modeling and forecasting results using different data sets found in the literature and in the retail industry are provided in Section 6. Conclusions and discussions are given in Section 7.

2. LITERATURE REVIEW

The primary instruments employed in Box-Jenkins time series modeling techniques are the correlograms of the observation data. It has been found that correlograms of different types of theoretical time series possess strikingly different characteristics. For example, a pure moving average (MA) time series of order q has an autocorrelation function plot which has a cut-off after q lags. A pure autoregressive (AR) time series of order p has a partial autocorrelation function plot which has a cut-off after p lags. These two features help a great deal in identifying models for real life time series if the corresponding correlogram has nearly zero values after a finite number of lags. A pure theoretical autoregressive-moving average (ARMA) time series, on the other hand, has a correlogram which lacks such salient characteristics that are easy to recognize visually. This observation creates significant difficulties in identifying ARMA models in practice and is one of the incentives for exploring new solutions.

The literature basically contains two types of models for a seasonal time series. One is the seasonal time series model, and the other periodic time series model. In applications of seasonal time series models, a model is used for all different seasons. That is to say, the model parameters are the same for all seasons in a seasonal time series while in the applications of periodic time series models an individual model is used for only one season. Superficially, the seasonal time series model employs fewer parameters than the periodic model and may not fit the data as well as the latter for all different seasons. Building a periodic time series model, on the other hand, could be much more difficult than building a seasonal time series model. These two types of models differ also in the autocorrelation structures (McLeod, 1992, 1994).

In the literature, there exist basically two strategies in identifying seasonal time series models and periodic time series models. One strategy is to utilize correlograms of the data, and the other is to employ information criteria such as AIC (Akaike Information Criterion) (Akaike, 1974), or BIC (Bayes Information Criterion) (Schwarz, 1978). With the first strategy, one tries to find cut-offs after a finite number of lags in the correlograms of the properly transformed data (Sakai, 1982). However, as it is unknown in advance how the data should be transformed and what structure the model may attain, this is a complicated trial-and-error process in seasonal time series model identification given the available literature (Box, Jenkins and Reinsel, 1994; Brockwell and Davis, 1996; Liu, 2006). To identify a periodic autoregressive time series model, for example, one may plot the partial autocorrelation function for each season, and try to find cut-offs after a finite number of lags (McLeod, 1992, 1994; Hurd and Gerr, 1991; Ula and Smadi, 2003; Wang *et al.*, 2005). To a large extent, this is still within the framework of the Box-Jenkins modeling technique. It appears that no satisfactory methods are available for building periodic autoregressive-moving average models (McLeod, 1994). With the information criterion strategy, one has to maintain a list of candidate models, either explicitly or implicitly, and enumerate the model list to minimize the chosen information criterion where each model should have a different combination of numbers of model parameters (Franses and Paap, 2004; Franses and Koehler, 1998; Franses and Paap, 1994; McLeod, 1992, 1994; Brockwell and Davis, 1996). With this strategy, once a model candidate is chosen, the model parameters must be estimated and the corresponding information criterion is calculated (Box, Jenkins and Reinsel, 1994; Adams and Goodwin, 1995). While this strategy imposes no difficulty for building seasonal time series models, it may become quite problematic when building periodic time series models as there might be too many different model candidates to consider (McLeod, 1992, 1994). In addition, the models identified with information criterion could be less parsimonious than those selected with the first strategy (McLeod, 1992). In addition to these two strategies in modeling seasonal time series, power spectra of time series are also used to identify models of seasonal time series (Box, Jenkins and Reinsel, 1994; Brockwell and Davis, 1996), and fast Fourier transformation can be used to model a seasonal time series (Tefsaye, Meershaert and Anderson, 2006). These are however, beyond the scope of this paper, and therefore will not be pursued here.

It should be pointed out that these aforementioned methods are not designed per se for automated modeling processes, albeit with proper modifications these methods could be automated. Therefore, it is highly desirable to have automated model building algorithms for seasonal and even periodic time series models.

From the application point of view, for an automated seasonal time series modeling algorithm, there exist at least four issues that must be resolved automatically. The first one is the trend detection. If there is a significant trend in the data, the algorithm should be able to detect it and de-trend the data by differencing the data properly (Franses and Taylor, 2000). The second is to detect and estimate the seasonality existing in the given time series as the periodicity estimation is crucial for modeling a seasonal time series. The next is to select a proper order of the model from a given family of models, and this must be done automatically as well. And the last one is to estimate the parameters of the model identified (Åström and Wittenmark, 1995; Box, Jenkins and Reinsel, 1994). To solve the first problem, a linear regression model can be set up using the data set, and if the slope of the regression line is significant, then there could be a trend in the data set. A more powerful tool is the Dickey-Fuller's algorithm (Dickey and Fuller, 1979) which tests if there is a unit-root in the model so as to determine if differencing is necessary. In this paper however, we adopt the first approach to detect possible trends in a time series while leaving the possibility of exploring the other approaches in the future. To detect the seasonality, a heuristic algorithm that is implicitly employed in Box, Jenkins and Reinsel (1994, p.342) can be applied. This algorithm utilizes the autocorrelation of the differenced time series, and seeks the lag of the maximal autocorrelation value. The corresponding lag is regarded as the period of the time series. To identify a proper model, the S-E algorithm can be used where the least square method or the maximum likelihood method shall be used to estimate the model parameters.

3. BRIEF REVIEW OF THE S-E ALGORITHM

In this section, we briefly review the S-E algorithm in order to appreciate how this algorithm can be used in modeling seasonal time series. The S-E algorithm consists of two sub-algorithms, Algorithm 1 and the Main Algorithm. Algorithm 1 is used to read the autocorrelation function or the partial autocorrelation function plots, a fundamental step in automated modeling process, while the Main Algorithm is the main body of model identification and estimation of which Algorithm 1 is a sub-routine.

To proceed, let us assume that $\{x_t\}$ is a stationary time series. Then, $\{x_t\}$ can be modeled in general by the following difference equation (Box, Jenkins and Reinsel, 1994):

$$\begin{aligned} x_t &= a_1 x_{t-1} + a_2 x_{t-2} + \cdots + a_p x_{t-p} \\ &\quad + \xi_t + b_1 \xi_{t-1} + b_2 \xi_{t-2} + \cdots + b_q \xi_{t-q} \end{aligned} \tag{1}$$

where $\{\xi_t\}$ is an *i.i.d.* noise process, p and q are the orders of the model. Such a model is called autoregressive-moving average(ARMA) model and denoted as ARMA (p, q), and the process is called an ARMA time series. To motivate our algorithm, let us rewrite ARMA(p, q) model (1) in a different form as follows:

$$x_t = a_1 x_{t-1} + a_2 x_{t-2} + \cdots + a_p x_{t-p} + \gamma_t \tag{2}$$

where

$$\gamma_t = \xi_t + b_1 \xi_{t-1} + b_2 \xi_{t-2} + \cdots + b_q \xi_{t-q} \tag{3}$$

and $\{\gamma_t\}$ can be regarded as an MA(q) time series. Obviously, (1) is equivalent to (2) and (3). That is, we purposefully decompose an ARMA(p, q) time series into two parts: one is an autoregressive process and the other a moving average process. $\{\gamma_t\}$ can be seen as the model error in (2). Note that it is not required here that $\{\gamma_t\}$ be uncorrelated, as implied by (3). Instead, $\{\gamma_t\}$ could be serially correlated and it is the dependence of $\{\gamma_t\}$ that we can draw information about q from based on a chosen value of p.

The S-E algorithm can be reviewed as follows. Suppose that both (2) and (3) are time series. We pick a value for p of the autoregressive part. Applying an estimation algorithm with the data, we obtain a set of model parameters of $a_1, a_2, \cdots, a_p$. Then, from this AR(p) model we obtain a model residual time series $\{\gamma_t\}$. If this time series is a white noise, then its autocorrelation function has a value of virtually zero for any non-zero lags, and this characteristic is very easy to recognize. In this case, we have identified p correctly and we know that the model is an AR(p). If $\{\gamma_t\}$ is correlated, and if its autocorrelation function has a cutoff after q lags, then $\{\gamma_t\}$ is an MA(q) time series and we need to add an MA(q) part to the model. Otherwise, if its autocorrelation function has tail-off, it means that $\{\gamma_t\}$ is either an autoregressive or a mixed time series. In either case, it suggests that in (2) the p value was not chosen properly. If so, we simply increase p by 1, and repeat the above process.

To determine the q value, we utilize a statistical hypothesis test. As is known, the autocorrelation function values of a white noise are zeros for all non-zero lags. For the estimated autocorrelation function of a white noise, the autocorrelation coefficients can be regarded as a random variable whose variance can be approximated with the following formula (Box, Jenkins and Reinsel, 1994), albeit better approximation algorithms exist,

$$\sigma = \frac{1}{\sqrt{N}} \tag{4}$$

where N is the sample size of the data used to calculate the autocorrelation function. Thus, for a given significance level α, if the percentage of autocorrelation coefficients of nonzero lags that are outside the confidence interval ($-k\sigma$, $k\sigma$) is less than 1-α where k is chosen properly, then there is a good reason to believe that the residual process is a white noise. Otherwise, we need to test further whether the residuals are a moving average or an autoregressive time series. The following paragraph describes Algorithm 1 concisely.

Algorithm 1

Step 1. Determine a significance level α, a positive k, and estimate σ using (4). Set $l = 0$.

Step 2. If ρ_l is outside the confidence interval ($-k\sigma$, $k\sigma$), set $l = l+1$, and repeat Step 2. Else, go to Step 3.

Step 3. Calculate the percentage ϕ of the autocorrelation coefficients that are outside of the confidence interval ($-k\sigma$, $k\sigma$) from lag l to the maximum lag. If ϕ is less than or equal to 1-α, then let $q = l$ and stop. Else, q is undetermined and stop.

Once the order of the model has been identified, we need to estimate the parameters in the model. After that, the probability structure of the model errors will be checked, and if needed the above algorithm will be executed again with a different order values.

We note that the approximation formula of (4) has some limitations. It is solely dependent on the number of data N. When N is sufficiently large, the estimated standard deviation of the correlation coefficients tends to be zero, and thus it is very likely to reject the null hypothesis that the model error is a white noise even though it is very close to it. For this reason, it is recommended to use the following formula instead to estimate the standard deviation of the correlation coefficients (Box, Jenkins and Reinsel, 1994),

$$\sigma_k = \frac{1}{N}\left(1 + 2\sum_{i=1}^{q} r_i^2\right) \tag{4′}$$

where $k > q$, r_i is the estimated autocorrelation coefficient, and i = 1, 2, $\cdots$, q. Next, we will review the major part of the S-E algorithm, which consists of three steps, to model stationary time series.

Main Algorithm:

Step 1. Use Algorithm 1 to test if the time series is AR or MA. If neither, let p = 0, and go to Step 2.

Step 2. Let $p = p+1$. Estimate the parameters of AR(p), and calculate model residuals. Use Algorithm 1 to test if the residuals are an MA time series. If yes, then go to Step 3 with q identified. Else, repeat Step 2.

Step 3. Estimate parameters for the tentative model ARMA(p, q), and calculate model residuals. Then, use Algorithm 1 to test if the residuals are a white noise process. If yes, stop. Else, go to Step 2.

The above algorithm has been applied to model a number of time series found in the literature and the results are very satisfactory. All models are built automatically without human interventions (Song and Esogbue, 2006).

We observe that using model residuals to help in model identification is not a new idea. In the literature, a different algorithm, called the extended autocorrelation function method developed by Tsay and Tiao (Liu, 2006) also uses the autocorrelation function of the residuals in helping model identification. However, to apply this algorithm one must construct a correlation matrix and identify a special structure formed by the zero elements of the matrix to help identify the orders of an ARMA model. This is not an easy process. For one, this algorithm requires many more calculations than does the S-E algorithm. For another, it has limited utilities in seasonal time series modeling (Liu, 2006, p. 3.13).

In the next section, we discuss how to build a seasonal time series model automatically by applying the S-E algorithm.

4. MAIN RESULTS

Suppose $\{x_t\}$ is a nonstationary seasonal time series with seasonality S. In addition, it is assumed that $\{x_t\}$ has a deterministic local trend. Let $\nabla = 1 - B$ be the differencing operator, and B the backshift operator. As differencing can remove trend in the time series, we may assume that a properly chosen positive integer d can be found so that the time series after d times of successive differencing becomes a stationary one, *i.e.*, $\{\nabla^d x_t\}$ is a stationary seasonal time series. Consequently, the stationary seasonal time series $\{\nabla^d x_t\}$ can be described by the following ARMA model

$$\Phi(B^s)\nabla^d x_t = \Theta(B^s)a_t \tag{5}$$

where $\Phi(B^s)$ and $\Theta(B^s)$ are polynomials of B^s, of proper orders. Evidently, if both $\Phi(B^s)$ and $\Theta(B^s)$ are properly determined, then all the seasonal components in the time series $\{\nabla^d x_t\}$ are modeled, and the residual process $\{a_t\}$ may be a stationary ARMA process. Therefore, both of these polynomials can be identified automatically by the S-E algorithm, as to be discussed later. Model (5) describes how data S lags apart are related to one another. However, it provides no information about how successive observations in the original time series $\{x_t\}$ are related. If x_t is also correlated with the most recent data in the past such as $x_{t-1}, x_{t-2}, \cdots$, then this may be reflected in the model residuals of (5) so that $\{a_t\}$ is a serially correlated series. And hence $\{a_t\}$ can be modeled by the following difference equation,

$$\phi(B)a_t = \theta(B)\varepsilon_t \tag{6}$$

where $\{\varepsilon_t\}$ is a white noise process. Substituting (6) into (5) will yield

$$\phi(B)\Phi(B^S)\nabla^d x_t = \theta(B)\Theta(B^S)\varepsilon_t \tag{7}$$

where the polynomials $\phi(B)$, $\Phi(B^S)$, $\theta(B)$ and $\Theta(B^S)$ are identical with the counterparts in (5) and (6). Our goal is to identify and estimate these four polynomials and determine the value of S and d in (7) automatically. One strategy might be to identify and estimate $\Phi(B^S)$ and $\Theta(B^S)$ in (5) first. If the residual $\{a_t\}$ is an MA or an AR process, then $\Phi(B^S)$ and $\Theta(B^S)$ could be determined easily. However, if $\{a_t\}$ is an ARMA process, then the automatic model building process could become very complicated because in this case the identification of $\Phi(B^S)$ and $\Theta(B^S)$ will be interwoven with that of $\{a_t\}$. Therefore, we will consider different strategies so that the automatic model building process could be as simple as possible in all cases. One different strategy is to identify the product of $\phi(B)$ and $\Phi(B^S)$, and the product of $\theta(B)$ and $\Theta(B^S)$ simultaneously. This may require imposing some constraints on the structures of all these polynomials. Nonetheless, this could make the model building process simpler.

Next, let us consider the general forms of $\phi(B)\Phi(B^S)$ and $\theta(B)\Theta(B^S)$ which have both seasonal and non-seasonal terms as follows,

$$\phi(B)\Phi(B^S) = 1 + \sum_{i=1}^{p}\left(a_i B^i + b_i B^{i+S-1}\right) \tag{9}$$

$$\theta(B)\Theta(B^S) = 1 + \sum_{i=1}^{q}\left(c_i B^i + d_i B^{i+S-1}\right) \tag{10}$$

Then, we have the following difference equation as the model of the stationary seasonal time series $\{\nabla^d x_t\}$,

$$\begin{aligned}\nabla^d x_t = &-\sum_{i=1}^{p} a_i \nabla^d x_{t-i} - \sum_{i=1}^{p} b_i \nabla^d x_{t-i-S+1} \\ &+ \sum_{j=0}^{q} c_j \varepsilon_{t-j} + \sum_{j=1}^{q} d_j \varepsilon_{t-j-S+1}\end{aligned} \tag{11}$$

where p and q are non-negative integers, $c_0 = 0$, a_i, b_i, c_j and d_j are real numbers for $i = 1$ to p, and $j = 1$ to q. We refer to (11) as the generalized seasonal ARMA model of $\{x_t\}$, or GSARMA for short.

It should be pointed out that model (11) bears a strong resemblance to the non-multiplicative seasonal models in Liu (2006, p. 3.15, p. 4.3), but it differs from Liu's in that the autoregressive polynomial of $\{\nabla^d x\}$ is also non-multiplicative in (11). In addition, by properly choosing parameters in (11), multiplicative models can be derived. For example, if $d = 0$, $a_1 = -1$, $b_{12} = 1$ and $b_{13} = -1$, and all other parameters in the autoregressive part are zeros, then model (11) will reduce to the multiplicative seasonal model in Box, Jenkins and Reinsel (1994, p. 333). Hence, if the time series is indeed a multiplicative seasonal time series, then (11) should be able to represent it well. Now, our task is to determine S, d and then identify p and q, and estimate the model parameters in (11) automatically for a nonstationary seasonal time series $\{x_t\}$. This can be achieved by applying the following algorithms.

4.1 Algorithm to Determine S

In Box, Jenkins and Reinsel (1994, p. 342), the autocorrelation function of $\{\nabla x_t\}$ is used to determine the seasonality of time series $\{x_t\}$. We employ the same algorithm for the same purpose here. This algorithm can be described as follows.

Algorithm to Determine S

Step 1. Difference time series $\{x_t\}$ to obtain $\{\nabla x_t\}$.

Step 2. Calculate the autocorrelation function values of $\{\nabla x_t\}$.

Step 3. Check the nonzero lag S at which the autocorrelation function value achieves the maximum. Then, S is the seasonality, and stop.

This algorithm is based on the observation that for seasonal time series, its autocorrelation function also exhibits the same seasonality. But, differencing the time series may remove the trend in the data so that a pure seasonal time series may be obtained, and hence it makes the autocorrelation function values more effective in detecting the seasonality. The maximum value and hence the corresponding lag of the autocorrelation function can be found easily with a linear search algorithm. Implicitly, this algorithm assumes that the autocorrelation function achieves its maximal value when the lag is coincident with the seasonal length. However, our experience and observation indicate that the effectiveness of this algorithm depends on, to a certain degree, the level of noise in the time series.

4.2 Algorithm to Determine *d*

One of the functionalities of the operator ∇^d is to eradicate any trends existing in the time series. Therefore, to determine the value of *d*, it is necessary to test if the existing trend can be eliminated by properly differencing the time series *d* times. This implies that if the differenced time series still has a trend, then it is necessary to difference the data again. To detect the trend in the data, a linear regression model can be set up using the original data, and the linear coefficient is tested (Neter, Wasserman and Kutner, 1990). If this coefficient is significant, then this indicates strongly that a linear trend exists in the data. In this case, differencing the time series is necessary, and after differencing the time series, a linear regression model is set up again using the differenced time series data, and the linear coefficient is tested again. This process may repeat for a few times. Therefore, *d* can be determined with the following algorithm.

Algorithm to Determine *d*

Step 1. Set $d = 0$, and define a significance level α.

Step 2. Set up a linear regression model using the current time series data.

Step 3. Check if the linear coefficient is significant. If yes, go to Step 4. Else, *d* is found and the algorithm stops.

Step 4. Set $d = d+1$ and difference the current time series to get a new one $\{\nabla^d x_t\}$. Go to Step 2.

We must point out that in the literature, such as in (Brockwell and Davis, 1996; Franses and Paap, 2004), different methodologies exist in modeling trend and trend eliminations. We will however, not pursue these well developed methodologies in this paper.

4.3 Algorithm to Determine $\phi(B)\Phi(B^S)$ and $\theta(B)\Theta(B^S)$

We propose the following algorithm to determine the product polynomial $\phi(B)\Phi(B^S)$ and the product polynomial $\theta(B)\Theta(B^S)$.

The Algorithm

Step 1. Set the order *p* of $\phi(B)\Phi(B^S)$ in (11) to be 1.

Step 2. Estimate the parameters a_i, b_i of $\phi(B)\Phi(B^S)$ for $i = 1$ to *p*, and calculate the model residuals. Use the S-E algorithm to test if the residuals are an MA series. If yes, then go to Step 3 with *q* identified. Else, set $p = p + 1$, and repeat Step 2.

Step 3. Set the order of $\theta(B)\Theta(B^S)$ in (11) to be *q*. Then, estimate the parameters of $\theta(B)\Theta(B^S)$, and calculate the model residuals. Apply the S-E algorithm to test if the residuals are a white noise. If yes, stop. Else, set $p = p + 1$, and go to Step 2.

This algorithm, with an extraordinary resemblance to the S-E algorithm, has obvious advantages. First, it relies on the probabilistic structure of the modeling errors to determine the properness of the order and parameters of a model. Thus, Algorithm 1 in Song and Esogbue (2006) can be used directly here to test the characteristics of model residuals. This makes it easy to automate the modeling process. In addition, it does not need more than only the necessary calculations to determine the model order as other algorithms may, such as the extended autocorrelation function method (Liu, 2006). This new algorithm is in line with the parsimony principle of modeling because it starts from the smallest possible values of *p* or *q*, and increases the value one at a time and terminates the first time when the modeling error becomes a white noise. The only constraint on the polynomials is that the seasonal and the non-seasonal parts have the same number of terms.

In general, we may consider polynomials with the following forms as the products of $\phi(B)$ and $\Phi(B^S)$, and of $\theta(B)$ and $\Theta(B^S)$, respectively,

$$\phi(B)\Phi(B^S) = 1 + \sum_{i=1}^{p}\sum_{j=0}^{k} a_{ij} B^{i+jS} \tag{12}$$

$$\theta(B)\Theta(B^S) = 1 + \sum_{i=1}^{q}\sum_{j=0}^{k} b_{ij} B^{i+jS} \tag{13}$$

In this form, the current value of the time series is associated with the values in the past *P* instants, and also the values *S*, 2*S*, $\cdots$, and *kS* lags apart from each of these past *P* instants. The same can be said about the observations and the model errors. A special case would be to choose $k = 2$ in (12) and (13). Then, we would have

$$\phi(B)\Phi(B^S) = 1 + \sum_{i=1}^{p}\left(a_i B^i + b_i B^{i+S-1} + c_i B^{i+2S-1}\right) \tag{14}$$

$$\theta(B)\Theta(B^S) = 1 + \sum_{i=1}^{q}\left(d_i B^i + e_i B^{i+S-1} + f_i B^{i+2S-1}\right) \tag{15}$$

More general cases can also be considered and the proposed algorithm can be easily applied with a minimum amount of modification.

5. EXTENSION TO PERIODIC TIME SERIES MODELS

The difference between a seasonal time series model and a periodic time series model lies in the fact that the former uses only one model and one set of parameters for all different seasons while the latter uses an individual model and an individual set of parameters for each season. From the modeling point of view, seasonal time series models employ much fewer parameters than periodic time series models for the same time series. For this reason, it is expected that periodic time series models can have better forecasts. However, this is not always the case.

In this section, we only consider extensions of the S-E algorithm to building periodic autoregressive models (PAR) for seasonal time series. As periodic autoregressive-moving average (PARMA) models are much harder to build automatically, extension to such cases will be handled differently and separately.

When using periodic autoregressive models, as shown in (McLeod, 1992, 1994), the autocorrelation functions of residuals of the models of different seasons are asymptotically uncorrelated. Theoretically, this provides the possibility of applying the S-E algorithm to build a seasonal time series model for each individual season independently, including model identification, parameter estimation, and model diagnostics. As this is a straightforward application of the new algorithm, we only outline the procedure below. Details of the process can be filled readily, and therefore will not be elaborated on in the sequel.

Suppose $\{x_t\}$ is a seasonal time series with S as the periodicity. Without loss of generality, suppose that the data can be partitioned into a few different segments each of which contains exactly S data points. Then, time index t can be expressed as $t = rS + m$ where $r = \mathrm{mod}(t,\ S)$, $m = 1, 2, \cdots, S$ and m is the seasonal index. To illustrate, if we have monthly data, then $S = 12$, and for each data we can locate the year of the data and also the month. Then, r and m can be determined accordingly. To proceed, suppose for each seasonal index m, the corresponding AR model is given below where the notation in Franses and Paap (2004) and McLeod (1994) is followed,

$$\begin{aligned} x_{(r,m)} &= a_{1,m}x_{(r,m)-1} + a_{2,m}x_{(r,m)-2} \\ &\quad + \ldots + a_{p_m,m}x_{(r,m)-p_m} + \varepsilon_{(r,m)} \end{aligned} \tag{16}$$

where p_m is the order of the PAR model, $a_{i,m}$ are model parameters, and $\varepsilon_{(r,m)}$ is the model residual, presumably a white noise. Note that model (16) is only for season m where $m = 1$ to S. For different seasonal indices m, both p_m and $a_{i,m}$ can be different. From the modeling point of view, model (16) describes how the current observation at season m of the time series is related to the most recent and adjacent observations. The estimated autocorrelation function of $\varepsilon_{(r,m)}$ is given by the following formula (McLeod, 1992),

$$\hat{\rho}_m(k) = \frac{\sum_r \hat{\varepsilon}_{(r,m)}\hat{\varepsilon}_{(r,m-k)}}{\sqrt{\sum_r \hat{\varepsilon}^2_{(r,m)} \sum_r \hat{\varepsilon}^2_{(r,m-k)}}} \tag{17}$$

where k is the lag and m is the seasonal index from 1 to S. Note that in calculating $\hat{\rho}_m(k)$ for season m, residuals of models of different seasons are required. The asymptotic distribution of $\hat{\rho}_m(k)$ can be found in (McLeod, 1978), and can be used in model identification for a PAR model. Moreover, as mentioned earlier, it has been shown that the theoretic autocorrelation functions of the residuals are asymptotically independent for different seasons (McLeod, 1992). This implies that model identification of PAR models can be approximately carried out independently for different seasons.

Nonetheless, it must be pointed out that the calculations of the estimated residual autocorrelation coefficients $\hat{\rho}_m(k)$ of different seasons are interdependent. This, as can be expected, creates difficulty in an automated model building process. This is because if it is found that the model residuals for a specific season are not a white noise, it is still hard to know whether it is due to the misspecification of the model for the current season, or due to the misspecification of models of other seasons. This causes difficulty in making a decision on model order selection, and this can lead to a combinatorial decision problem, which should be avoided for now. For this reason, in this paper we want to explore the possibility of building PAR models using only model errors of the same season. The assumption is that if all the models are properly built, then the model errors must be white noise for all seasons. In this case, for each season, the modeling errors must be also a white noise. Although the inverse is not true in general, we want to develop an approximate but practical algorithm for automated PAR modeling, and will verify the effectiveness through examples. Specifically, we can use the S-E algorithm to build a PAR model for each season automatically and independently of other seasons. When models for all seasons are identified properly, the modeling task is completed.

To proceed, note that although (16) is called a periodic time series model, it does not model the relation between the current observation and the observation of the same season explicitly, albeit when $P_m > S$ model (16) will contain observations of the same season. It is not clear to the author why this is the case. Therefore, it is interesting to see if observations of the same season can be included in the model directly in addition to the adjacent observations. The following model can be used to achieve this goal,

$$x_{(r,m)} = \sum_{i=1}^{P_m} a_{i,m}x_{(r,m)-i} + \sum_{i=1}^{P_m} b_{i,m}x_{(r,m)-i-S+1} + \varepsilon_{(r,m)} \tag{18}$$

or we can use the following alternative form as the model,

$$x_{(r,m)} = \sum_{i=1}^{P_m} a_{i,m} x_{(r,m)-i} + \sum_{i=1}^{R_m} b_{i,m} x_{(r,m)-i\cdot S} + \varepsilon_{(r,m)} \quad (19)$$

For notational convenience, (18) and (19) will be referred to as the generalized periodical AR models, or GPAR for short. Variants of these two models exist. From (18) or (19), it can be seen that the Algorithm presented in Section 4.3 of this paper can be used directly. Thus, extension of the S-E algorithm to periodical autoregressive models is quite straightforward. Note that for all these models, this new algorithm will first identify P_m and R_m for each season, then estimate the parameters, and finally diagnose the goodness-of-fit of the model. All the modeling process can be carried out automatically.

6. NUMERICAL EXAMPLES

In this section, we provide modeling and forecasting examples using the new algorithms applied to a number of data sets obtained from the literature and in the retail industry. Both seasonal time series models and periodic time series models will be illustrated. To apply the new algorithms, we implemented the algorithm in Section 4.1 using Java to identify periodicities, the algorithm in Section 4.2 to detect trends where the significance level is set to be 0.95, and the algorithm in Section 4.3 to identify model types and orders, used model (11) as the seasonal time series models for all the relevant illustration examples, and model (18) as the periodic time series models. In addition, only ACF of the residuals is used in model identification.

6.1 Illustration 1. Airline Data

The airline data have been used widely in the literature as examples of modeling seasonal time series. In Box, Jenkins and Reinsel (1994), a multiplicative seasonal (0, 1, 1) × (0, 1, 1)$_{12}$ model is used to model the airline data. The polynomial form of the model can be expressed as follows,

$$\nabla\nabla_{12} x_t = (1-\theta B)(1-\Theta B^{12})\varepsilon_t \quad (20)$$

where θ and Θ are the model parameters whose estimated values are $\hat{\theta} = 0.4$ and $\hat{\Theta} = 0.6$ respectively (Box, Jenkins and Reinsel,1994). The one-month-ahead forecasts for the last three years can be obtained with the MAPE being 1.26% and the forecasts are calculated using (20) recursively for the last 36 months. Figure 1 presents the forecast and the actual data.

For comparison purposes, a seasonal ARMA model is built automatically using the proposed algorithm. To do so, the last three years' data are not used in modeling, but are used only in forecasting accuracy calculation. Briefly, the first task for the algorithm is to test if there exists a trend in the data. After testing the existence of a trend, the algorithm differences the data once, and then tests the existence of trend in the differenced data. Then, after detecting no trends in the differenced data, the algorithm detects the existence of periodicity of the seasonal time series, and finds the periodicity to be 12. Next, the algorithm assumes a seasonal GSAR(1) model for the differenced time series, i.e., it assumes the following model $\nabla x_t = a_1 \nabla x_{t-1} + b_1 \nabla x_{t-12} + \varepsilon_t$ and identifies the parameters a_1 and b_1 using the least square method. Once the parameters are estimated, the model residuals are calculated and the autocorrelation functions of the residuals are es-

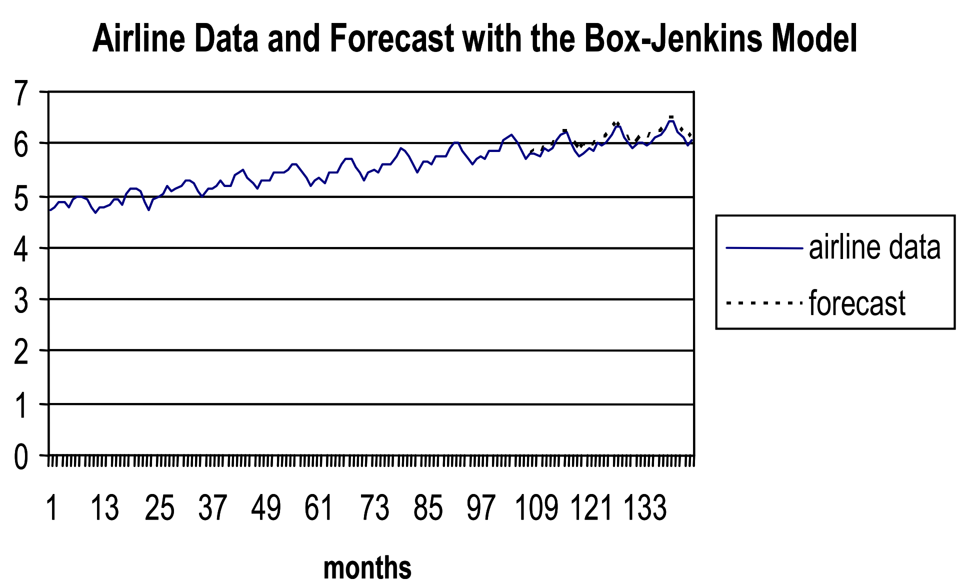

Figure 1. Actual airline data and the forecast with the Box-Jenkins model for the last three years

timated. The algorithm finds that the autocorrelation function of the residuals is not that of a white noise or an MA process. Hence, it increases the order of AR part by 1 and then a seasonal GSAR(2) model $\nabla x_t = a_1 \nabla x_{t-1} + a_2 \nabla x_{t-2} + b_1 \nabla x_{t-12} + b_2 \nabla x_{t-13} + \varepsilon_t$ is tested. After all the 4 parameters are estimated, the autocorrelation function of the model residuals is estimated, and the algorithm checks and finds that the autocorrelation function can be treated as that of a white noise at the significance level 90%. Then, the algorithm terminates. The final SAR(2) model built by the automated algorithm is shown below,

$$\begin{aligned}\nabla x_t &= -0.296 \nabla x_{t-1} - 0.035 \nabla x_{t-2} \\ &\quad + 0.913 \nabla x_{t-12} + 0.339 \nabla x_{t-13} + \varepsilon_t\end{aligned} \tag{21}$$

The estimated autocorrelation function is plotted in Figure 2 where it can be seen that at the given significance level the residuals can be treated as a white noise, but at a few lags the correlation values are still quite large. If we increase the significance level, the order of the identified model will be increased. To get forecasts $\hat{x}_t$ for $t = 109$ to

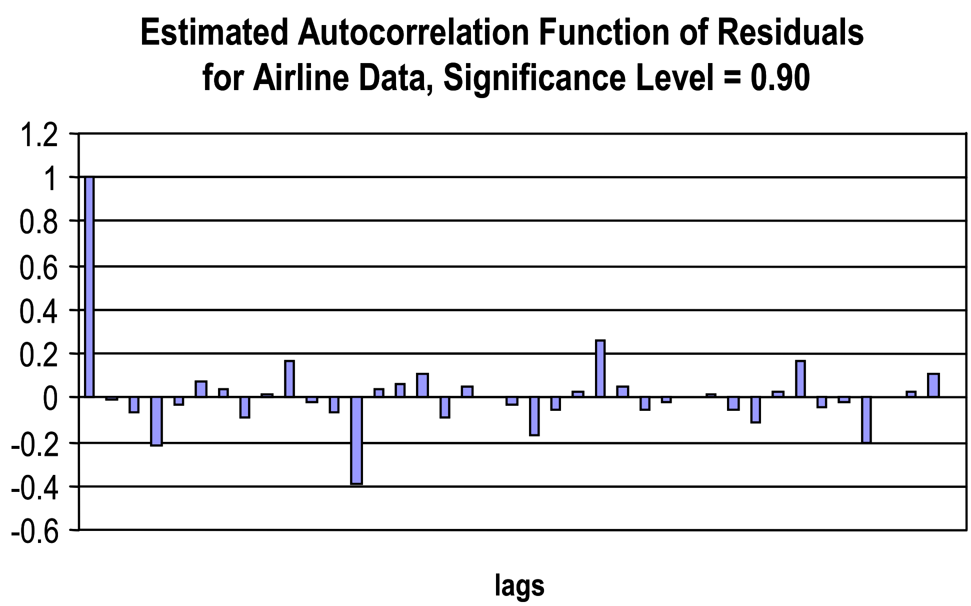

Figure 2. Estimated autocorrelation function of the residuals for the airline data.

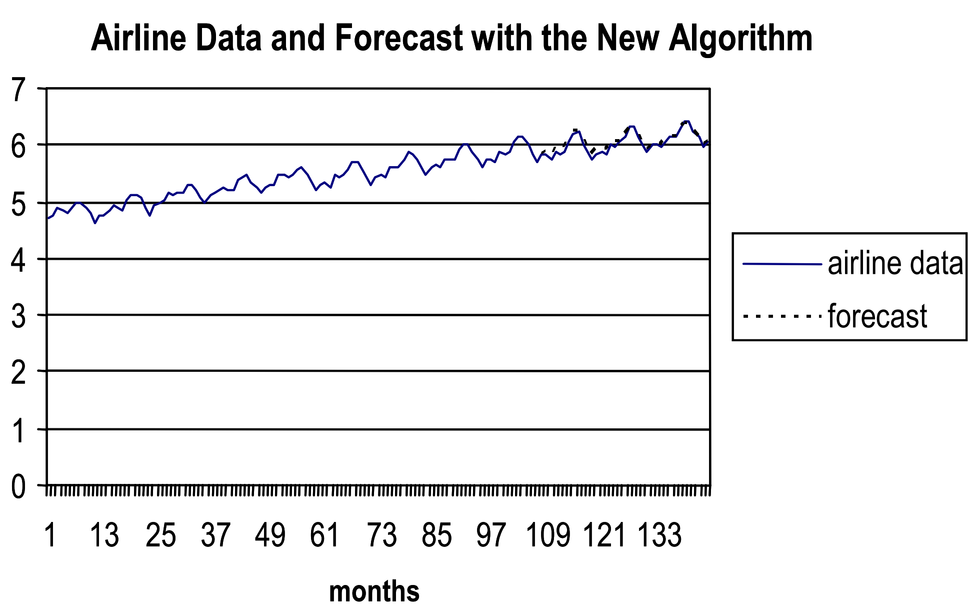

Figure 3. Actual airline data and the forecast with the new algorithm for the last three years

144, we use the following formula $\hat{x}_t = \sum_{i=108}^{t-1} \nabla \overline{x}_i + x_{108}$ where $\nabla \overline{x}_i$ is obtained recursively using model (21) for i = 108 to 143, and x_{108} is the initial condition. With the new algorithm and model (21), the MAPE of the forecasts for the last three years is 0.72%, and the forecasts are plotted in Figure 3. It can be seen that the new model has improved the forecast accuracy by about 40% compared to the Box-Jenkins model (20) which yields a MAPE of 1.26%.

6.2 Illustration 2. Retail Daily Demand Forecast

In the retail industry, it is often required to have the daily demand forecast for the entire coming week for workforce schedule purposes. The forecast accuracy has

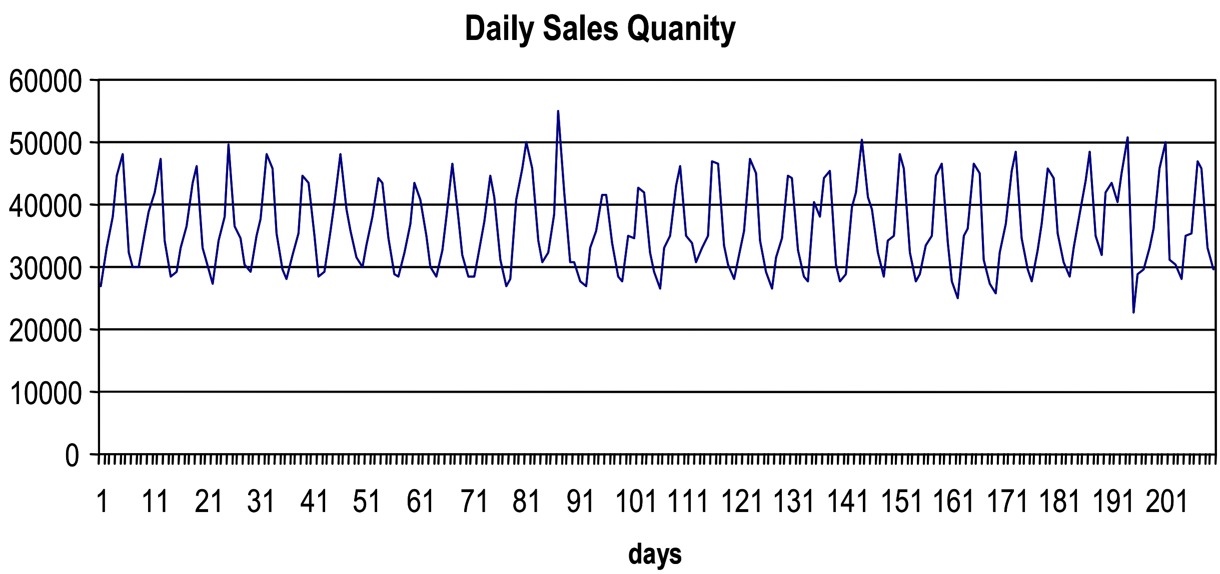

Figure 4. Plot of the actual daily sales data

Table 1. Retail Daily Sales Data (data is read cross the row from left to right)

26796	27472	29068	28590	32300	26596	26690	28604	25820	42014
33204	34278	35620	32762	38424	33038	31550	34080	32262	43594
38260	38054	40498	37944	55144	34918	34636	35150	37070	40490
44716	49580	48104	46678	41948	43002	44804	47914	45568	45026
47928	36680	39054	38100	30852	46014	44222	45714	48342	50686
32480	34770	35680	31886	30874	34876	32522	32392	34552	22518
30030	30536	31684	28548	27838	33888	28304	27808	29614	29018
29914	29394	29970	28272	26912	30610	27558	28960	27680	29682
35324	34886	33504	32086	33080	32976	40566	33304	32794	33238
38730	37758	38168	37134	35660	34902	38140	34842	37094	36302
42076	48258	44372	44666	41588	47068	44352	44432	45810	45622
47316	45884	43404	41230	41688	46562	45286	46350	44176	50018
34286	35198	34448	31118	33690	33520	30324	34028	35418	31144
28412	29714	28860	26816	28640	30398	27868	27614	30934	30378
29278	27892	28616	28224	27658	28192	28660	25126	28648	27950
33222	31926	32118	40600	34990	32882	39722	35064	32886	34964
36424	35486	36912	45822	34606	35888	41852	36262	38328	35460
43446	44456	43568	50126	42820	47132	50212	46348	43826	46902
46102	43628	40646	45626	42076	45034	41150	45090	48618	45650
33166	34900	35046	34062	32218	34128	39156	31100	34834	32996
29544	28540	29812	30818	29400	29374	32306	27322	31754	29698

some degree of impact on the workforce schedules. Overage in daily demand forecast could lead to over-schedules of employees and incurs labor cost to the store, and underage in forecast could cause under-schedules of employees, and in turn this may create customer dissatisfaction when not enough staffs are available. So, having good daily demand forecast is important. In this example, we will create a seasonal time series model using the new algorithm automatically, and calculate the daily forecast for an entire week.

The data of total daily sales are from a client for a period of 210 days, as shown in Table 1. Our task is to let the new algorithm automatically build a seasonal time series model using the first 203 days' data, and then we will apply the model to forecast the daily sales for the last week. As can be seen from Figure 4, no trends can be detected visually and this is verified by the algorithm. Then, the algorithm detects the periodicity of the data as 7. After this, the algorithm automatically builds the following GSARMA(1, 1) model,

$$x_t = 0.1824x_{t-1} + 0.814x_{t-7} + 0.212\varepsilon_t + 0.018\varepsilon_{t-7} \quad (22)$$

The autocorrelation function of the residuals of the model is shown in Figure 5 which indicates the properness of the model. The MAPE of the forecast over the last week is 5.25% which is a good forecast accuracy measure in retail forecasting. Figure 6 exhibits the actual and the forecast for the last week, and Table 2 lists the data.

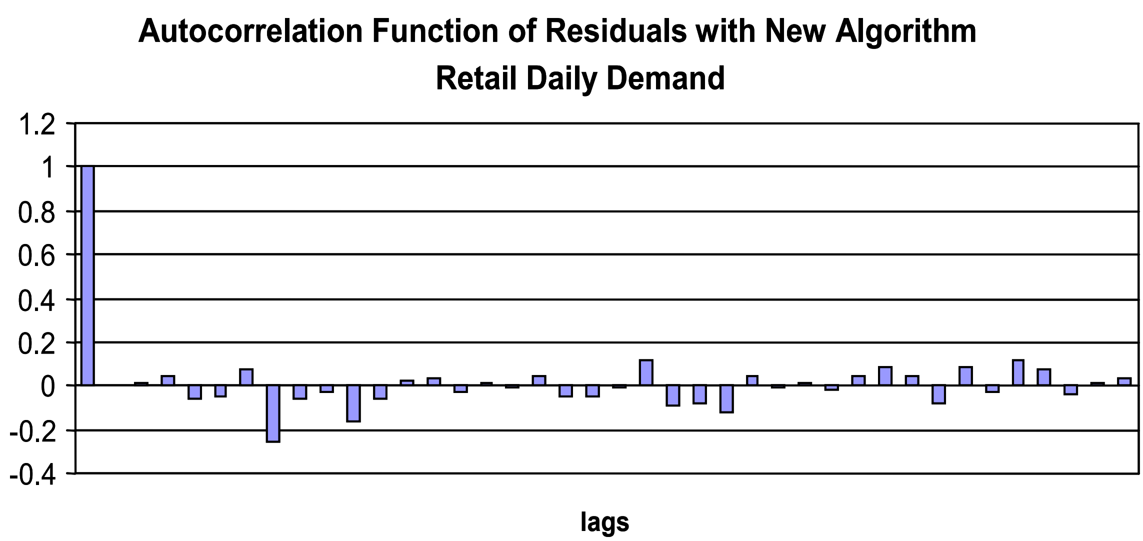

Figure 5. Estimated autocorrelation function of the residuals using the new algorithm for the retail daily demand data

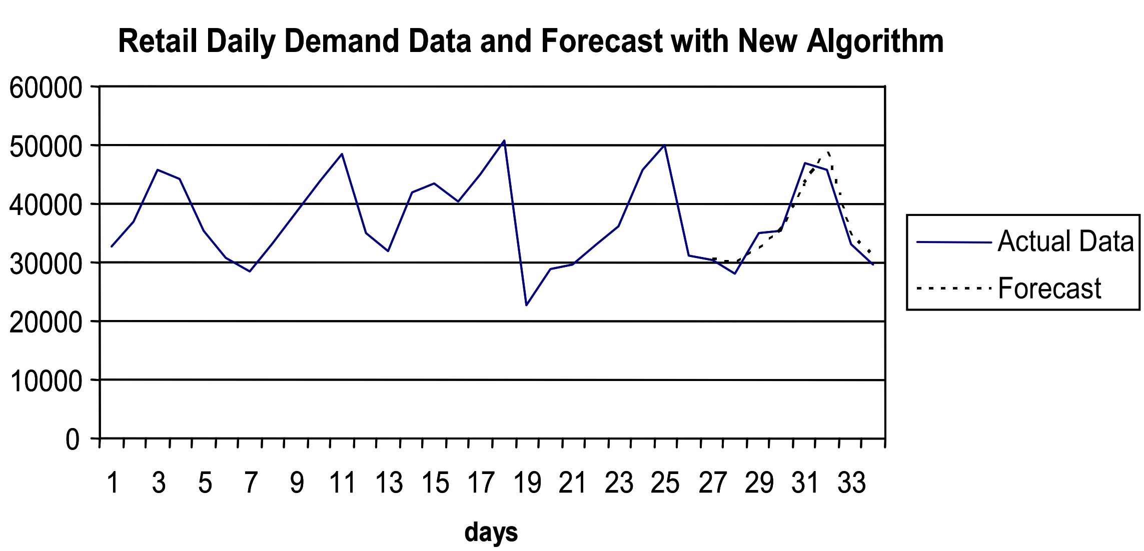

Figure 6. Actual and forecast of daily sales with the new algorithm

Table 2. Actual and Forecasted Daily Sales with 2 Different Models

Day	1	2	3	4	5	6	7	MAPE
Actual Data	27950	34964	35460	46902	45650	32996	29698	N/A
Seasonal Model	29856	32307	35328	43533	48682	34215	31021	5.25%
Periodic Models	29749	34754	37507	45391	46779	32800	29617	2.77%

6.3 Illustration 3. Monthly Accidental Deaths Data

This data set has been used in Brockwell and Davis (1996) as an illustrative example of modeling seasonal time series. We want to use the new algorithm to build a model for this data set and get the forecast so that comparisons can be made with the forecast from Brockwell and Davis (1996). This data set contains 78 data points over a period of 7 years. In Brockwell and Davis (1996), the first 72 data are used to build a seasonal time series model using the estimated autocorrelation function as the tool to select models, and using the maximum likelihood method to select the parameters. This leads to the following two models (Brockwell and Davis, 1996),

Model 1: $\nabla\nabla_{12}x_t = 28.831 + (1-0.478B)(1-0.588B^{12})\varepsilon_t$

Model 2: $\nabla\nabla_{12}x_t = 28.831 + \varepsilon_t - 0.596\varepsilon_{t-1} - 0.407\varepsilon_{t-6}$
$-0.685\varepsilon_{t-12} + 0.460\varepsilon_{t-13}$

Then, forecast is made for the last 6 months and comparisons of the forecast and the actual data are made. Table 3 lists the forecasts of the above two models with the MAPEs being 6.10% and 4.80% respectively. The new algorithm, after successfully detecting the periodicity of 12 and failing to find a significant trend in the data, automatically builds the following GSAR(1) model using the first 72 data points

$$x_t = 0.402x_{t-1} + 0.588x_{t-12} + \varepsilon_t \qquad (23)$$

Table 3. Actual Accident Death Data and Forecasts with 3 Different Models

Time	73	74	75	76	77	78	MAPE
Actual	7798	7406	8363	8460	9217	9316	N/A
Model 1	8441	7704	8549	8885	9843	10279	6.10%
Model 2	8345	7619	8356	8742	9795	10179	4.80%
New Model	8322	7398	7555	7817	8501	8964	5.94%

Note : Model 1 and Model 2 are from Brockwell and Davis, 1996.

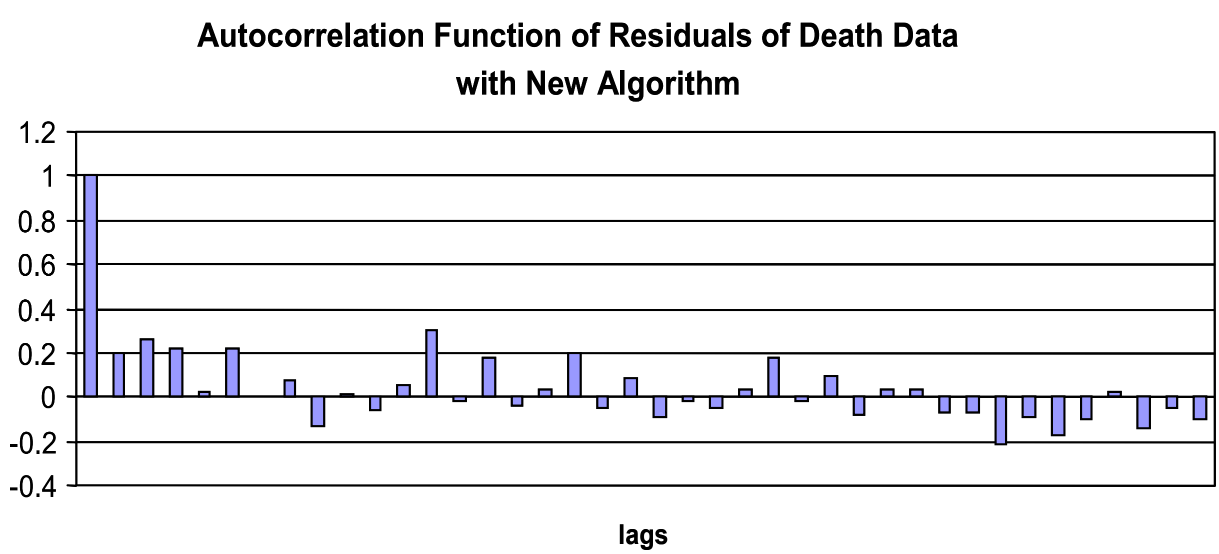

Figure 7. Estimated autocorrelation function of the residuals of the death data using the new algorithm at the significance level 0.9

Figure 7 shows the estimated autocorrelation function of the residuals which is treated by the algorithm as that of a white noise at the significance level of 0.9. The forecasts using this model for the last 6 months are listed in Table 3 where the MAPE is found to be 5.94%. Compared to the forecast results using the two models in Brockwell and Davis (1996), the new model generates forecasts slightly better than those of Model 1, but worse than those of Model 2. In Table 3, it is interesting to note that Model 1, which contains 3 parameters, has the highest MAPE of forecasts, the new model has 2 parameters and has a slightly lower MAPE, and Model 2 has 5 parameters with the best forecast results. It seems that there is a trade-off between model parsimony and forecasting accuracy. The structure of the model also has an important impact on forecasting accuracy.

6.4 Illustration 4. Periodic Models of Retail Daily Demand

In this example, we show how to apply the new algorithm to automatically build periodic models for the retail daily demand data used in Illustration 2. Our goal is to automatically build a PAR model for each season. In this illustration, we have 7 seasons each for a weekday of the week. The new algorithm builds the following seven GPAR(1) models of (18) automatically,

$$x_{(r,0)} = 0.767x_{(t,0)-1} + 0.218x_{(t,0)-7} + \varepsilon_{(r,0)}$$

$$x_{(r,1)} = 0.770x_{(t,1)-1} + 0.357x_{(t,1)-7} + \varepsilon_{(r,1)}$$

$$x_{(r,2)} = 0.751x_{(t,2)-1} + 0.314x_{(t,2)-7} + \varepsilon_{(r,2)}$$

$$x_{(r,3)} = 0.223x_{(t,3)-1} + 0.812x_{(t,3)-7} + \varepsilon_{(r,3)}$$

$$x_{(r,4)} = 0.498x_{(t,4)-1} + 0.484x_{(t,4)-7} + \varepsilon_{(r,4)}$$

$$x_{(r,5)} = 0.319x_{(t,5)-1} + 0.575x_{(t,5)-7} + \varepsilon_{(r,5)}$$

$$x_{(r,6)} = 0.490x_{(t,6)-1} + 0.446x_{(t,6)-7} + \varepsilon_{(r,6)}$$

It seems that no models are quite similar to each other for different seasons. Note that the data of the last week are not used in model building. Forecast is performed for the last week with a MAPE as 2.77% which is much better than that of using a seasonal model in Illustration 2. Table 2 lists the forecasts for the entire week using the GPAR(1) models. Significance level is again 90% in testing white noise of the modeling errors in the algorithm. For each season, only 29 error data points are obtained and used to estimate the autocorrelation coefficients. From the model diagnostics point of view, more data are needed. But, from the practical forecast point of view, the results are very satisfactory. It is worth to point out that some of the models identified above are not stationary, which can be verified by checking the parameters of the models. However, this should not be a serious problem here as the models are used to forecast for one data point only, and new models will be identified for a new data point. With respect to each season, we note that only 29 data points are used in modeling and this might be the cause for the non-stationarity of the models.

For convenience and comparison purpose, we provide Table 4 which summarizes the forecasting performances of all the models and data sets in the examples.

Table 4. Summary of Forecasting Performance of Different Models

Data Sets	Models	MAPE of Forecast
Airline Data Set	Box-Jenkins Model	1.26%
	New Model	0.72%
Retail Daily Sales Data	Seasonal	5.25%
	Periodic	2.77%
Death Data Set	Model 1	6.10%
	Model 2	4.80%
	New Model	5.94%

Note : Model 1 and Model 2 are from Brockwell and Davis, 1996.

7. DISCUSSION AND CONCLUSION

In this paper, new automated algorithms are proposed for creating both seasonal and periodic models for nonstationary seasonal time series. This includes detecting trends, estimating seasonalities, identifying model types and orders, and estimating model parameters. Algorithms are outlined for two different cases and models, and numeric examples are provided. The numeric examples indicate that the algorithms are practical and effective. More importantly, in all the illustration examples, the resultant models are built automatically without any human interventions, which is the goal of this paper.

The seasonal time series models of (11) to (15) can be seen as a generalization of the seasonal time series models in Box, Jenkins and Reinsel (1994) as we adopt more general forms for the polynomials of B and B^S in the model whereas in the seasonal models of Box, Jenkins and Reinsel (1994), specific forms of the polynomial are employed, leading to, for example, various multiplicative seasonal models. However, it is possible that in the generalized seasonal models, more parameters may be included.

Periodic time series models have been studied extensively in the literature. Models (18) and (19) can be seen as a generalization of the popular models in the literature (Franses and Paap, 2004; McLeod, 1992). In the generalized models, observations of the same season are explicitly included whereas in the literature these observations may not be included in the model at all if the order of the model is smaller than the periodicity of the time series. Additionally, in the literature, model diagnostics depends on the autocorrelations given by (17) whose calculations are interdependent for different seasons. In this paper, the generalized periodic time series model diagnostics is carried out by using the model residuals of the same season. The numeric results seem to indicate no contradictions to such a strategy. Although these results are very satisfactory, we propose to conduct in the future more numeric explorations to verify the effectiveness of this strategy.

ACKNOWLEDGMENT

The authors want to express their gratitude to the anonymous referees for their instructive comments and suggestions regarding how to revise this paper.

REFERENCES

Adams, H. and Goodwin, G. C. (1995), Parameter estimation for periodic ARMA models, *Journal of Time Series Analysis*, **16**, 127-145.

Akaike, H. (1974), A new look at the statistical model identification, IEEE Trans. On Automatic Control, **AC-19**, 716-723.

Åström, K. J. and Wittenmark, B. (1995), Adaptive Control, 2nd Ed., Addison-Wesley.

Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (1994), Time Series Analysis: Forecasting and Control, 3rd Ed., Prentice-Hall, Inc., New Jersey.

Brockwell, P. J. and Davis, R. A. (1996), Introduction to Time Series and Forecasting, Springer-Verlag, New York.

Dickey, D. A. and Fuller, W. A. (1979), Distribution of the estimators for autoregressive time series with a unit root, *Journal of the American Statistical Association*, **74**, 427-431.

Franses, P. H. and Paap, R. (2004), Periodic Time Series Models, Oxford University Press.

Franses, P. H. and Koehler, A. B. (1998), A model selection strategy for time series with increasing seasonal variation, *International Journal of Forecasting*, **14**, 405-414.

Franses, P. H. and Paap, R. (1994), Model selection in periodic autoregressions, *Oxford Bulletin of Economics and Statistics*, **56**, 421-440.

Franses, P. H. and Taylor, A. M. R. (2000), Determining the order of differencing in seasonal time series processes, *Econometrics Journal*, **3**, 250-264.

Hurd, H. L. and Gerr, N. L. (1991), Graphical methods for determining the presence of periodic correlation, *Journal of Time Series Analysis*, **12**, 337-350.

Liu, L. M. (2006), Time Series Analysis and Forecasting, 2nd Ed., Scientific Computing Associates.

McLeod, A. I. (1978), On the distribution of residual autocorrelation in Box-Jenkins models, *J. R. Statistics. Soc*, **B 40**, 296-302.

McLeod, A. I. (1992), Parsimony, model adequacy and periodic correlation in time series forecasting, *International Statistical Review*, **61**, 387-393.

McLeod, A. I. (1994), Diagnostic checking of periodic autoregression models with application, *Journal of Time Series Analysis*, **15**, 221-233.

Neter, J., Wasserman, W., and Kutner, M. H. (1989), Applied Linear Regression Models, 2nd Ed., Irwin, Inc.

Reilly, D. P. (1980), Experiences with an automatic Box-Jenkins Modeling Algorithm, Time Series Analysis, ed. O. D. Anderson, Amsterdam: North-Holland, 493-508.

Reilly, D. P. (1987), Experiences with an automatic transfer function algorithm, Computer Science and Statistics Proceedings of the 19th Symposium on the Interface, ed. R. M. Heiberger, Alexandria, VI: American Statistical Association, 128-135.

Sakai, H. (1982), Circular lattice filtering using Pagano's method, *IEEE Trans. Acoust. Speech Signal Process.*, **30**, 279-287.

Schwarz, G. (1978), Estimating the dimension of a model, *Annals of Statistics*, **6**, 461-464.

Song, Q. and Esogbue, A. O. (2006), A New Algorithm for Automated Box-Jenkins ARMA Time Series Modeling Using Residual Autocorrelation/Partial Autocorrelation Functions, *International Journal of Industrial Engineering and Management Systems*, **5**(2), 116-125.

Talluri, K. T. and van Ryzin, G. J. (2004), The Theory and Practice of Revenue Management, Kluwer Academic Publishers, Boston.

Tefsaye, T. G., Meerschaert, M. M., and Anderson P. L. (2006), Identification of periodic autoregressive moving average models and their application to the modeling of river flows, *Water Resources Research*, **42**.

Ula, T. A. and Smadi, A. A. (2003), Identification of periodic moving average models, *Communications in Statistics: Theory and Methods*, **32**(12), 2465–2475.

Wang, W., Van Gelder, P. H. A. J. M., Vrijling, J. K., and Ma, J. (2005), Testing and modeling autoregressive conditional heteroskedasticity of stream flow processes, *Nonlinear Processes in Geophysics*, **12**, 55-66.

Wu, S. M. and Pandit, S. M. (1979), Time Series and Systems Analysis, Modeling and Applications, Wisconsin University.