Adaptive Seasonal Time Series Models for Forecasting Short-Term Traffic Flow

Article *in* Transportation Research Record Journal of the Transportation Research Board · December 2008

DOI: 10.3141/2024-14

2 authors, including:

Billy M Williams
North Carolina State University

Adaptive Seasonal Time Series Models for Forecasting Short-Term Traffic Flow

Shashank Shekhar and Billy M. Williams

Conventionally, most traffic forecasting models have been applied in a static framework in which new observations are not used to update model parameters automatically. The need to perform periodic parameter reestimation at each forecast location is a major disadvantage of such models. From a practical standpoint, the usefulness of any model depends not only on its accuracy but also on its ease of implementation and maintenance. This paper presents an adaptive parameter estimation methodology for univariate traffic condition forecasting through use of three well-known filtering techniques: the Kalman filter, recursive least squares, and least mean squares. Results show that forecasts obtained from recursive adaptive filtering methods are comparable with those from maximum likelihood estimated models. The adaptive methods deliver this performance at a significantly lower computational cost. As recursive, self-tuning predictors, the adaptive filters offer plug-and-play capability ideal for implementation in real-time management and control systems. The investigation presented in this paper also demonstrates the robustness and stability of the seasonal time series model underlying the adaptive filtering techniques.

Traffic condition forecasting plays a key role in many intelligent transportation systems (ITS) applications. Increased availability of roadway sensor data has made it possible to monitor traffic states in real time. The ability to correctly anticipate future traffic states lies at the heart of ITS applications such as incident detection, real-time traveler information, route guidance, and ramp metering. Time series analysis provides an important class of models which can be used for traffic state prediction.

Time series models belong to the family of parametric models. Before parametric models can be used for forecasting, an initial data set is required in order to estimate the parameters. Conventionally, most parametric models have been implemented statically. The term “static” in this case refers to the fact that these models do not use information from the incoming data stream to update the model parameters. The estimation step provides a fixed set of parameter values used for the forecasting process. In this framework, parameters are only updated during periodic reestimation. For ITS network application, parameter estimation is required for each location where data are collected and forecasts are desired. From a system implementation viewpoint, the need to perform periodic parameter reestimation at each traffic-monitoring location is a major drawback for parametric models. Systemwide use of parametric models will be greatly facilitated if such models can be implemented in an adaptive framework.

Typically, parameter estimation is done using the maximum likelihood (ML) method. In general, the ML method provides parameters with the greatest predictive power. Therefore, step-by-step ML-based parameter reestimation could be considered a candidate approach for providing model adaptability. However, the computational burden of such an approach would be very high, and the computational burden increases exponentially as the number of parameters and size of the training data set increases. Therefore, recalculating ML-based estimates after each time step would be computationally impractical.

To provide computationally tractable adaptability, this paper applies common techniques from the field of adaptive filtering in order to automate parameter estimation and updating. The three techniques explored here are the Kalman filter (KF), recursive least squares (RLS), and least mean squares (LMS). Using an adaptive approach empowers the prediction models with plug-and-play capability and provides accurate forecasts with minimum human oversight and supervision. Therefore, the key question is whether such techniques can deliver acceptable performance.

This paper applies adaptive filters to a specific parametric model: the seasonal autoregressive integrated moving average model. Before the details are presented, it is worthwhile to describe the problem at hand and define its scope. The traffic state variable used in this study is the 15-min flow rate. This is simply the number of vehicles crossing a point along a network link in 15 min expressed in terms of vehicles per hour.

The 15-min interval is a common time interval used in many traffic engineering analyses. Furthermore, a 15-min flow rate has a fine enough resolution to track dynamic changes in the traffic state to a degree sufficient for many applications, while at the same time providing a degree of smoothing sufficient to filter out noise caused by high-frequency traffic stream fluctuations. The model uses 15-min flow data as the input and provides forecasts for the next one or multiple 15-min intervals into the future. Even though the research presented in this paper is based on a 15-min aggregation interval, the results should be applicable to other aggregation intervals. The range of aggregation interval applicability is a topic for further research.

The time series model used in this paper is univariate, meaning that each location on a network link where data are collected is modeled as a separate series. In order to forecast flow at such a location, only past data from that location are used. In other words, no data are used from neighboring locations. The data used in this research come from three- and four-lane urban and rural motorway sections in the United Kingdom.

The remainder of the paper is organized as follows. A description of the time series model used for forecasting is presented first, followed

S. Shekhar, Parsons Brinckerhoff, 909 Aviation Parkway, Suite 1500, Morrisville, NC 27560. B. M. Williams, Department of Civil, Construction, and Environmental Engineering, North Carolina State University, 424B Mann Hall, 2501 Stinson Drive, Raleigh, NC 27695. Corresponding author: B. M. Williams, billy_williams@ncsu.edu.

Transportation Research Record: Journal of the Transportation Research Board, No. 2024, Transportation Research Board of the National Academies, Washington, D.C., 2007, pp. 116–125.
DOI: 10.3141/2024-14

by a brief description of the three adaptive filters. Next, the data used for this study are discussed. The final two sections present the study's findings and conclusions.

SEASONAL TIME SERIES MODEL

The two fundamental building blocks of a univariate time series model for a stationary data stream are the autoregressive (AR) component and the moving average (MA) component. In a purely AR model, the forecast is a function of its past observations only out to some finite autoregressive order. For a purely MA model, the forecast could be thought of as a function only of past forecast errors or process innovations. However, MA processes can also be thought of as recursive smoothing filters with exponentially decaying dependencies on past observations. For example, simple exponential smoothing can be expressed in MA form. The more general autoregressive moving average (ARMA) model contains both AR and MA terms. In mathematical form, these models are expressed as

Autoregressive model of order *p*, AR (*p*)

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \epsilon_t \tag{1}$$

Moving average model of order *q*, MA (*q*)

$$x_t = \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q} \tag{2}$$

Autoregressive moving average model of order (*p*, *q*), ARMA (*p*, *q*)

$$\begin{aligned} x_t = {} & \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} \\ & - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q} \end{aligned} \tag{3}$$

where x and ϵ denote observed variable and the error series, respectively, and ϕ and θ denote AR and MA parameters.

The order of such models is given by the highest lagged term present. Using the backshift operator, B (defined as $Bx_t = x_{t-1}$ and $B^S x_t = x_{t-S}$), the three equations above can be rewritten as follows:

$$\left(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p\right) x_t = \epsilon_t$$

or

$$\phi(B) x_t = \epsilon_t \tag{4}$$

$$x_t = \left(1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q\right) \epsilon_t$$

or

$$x_t = \theta(B) \epsilon_t \tag{5}$$

$$\left(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p\right) x_t = \left(1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q\right) \epsilon_t$$

or

$$\phi(B) x_t = \theta(B) \epsilon_t \tag{6}$$

An ARMA model can only be applied to a weakly stationary time series. A series is said to be weakly stationary if its mean is constant and covariance is dependent only on the lag between correlated series values and not on time (*1*). The most common method to induce stationarity is through the use of differencing. An ARMA model fitted to a differenced series is called an autoregressive integrated moving average (ARIMA) model. The order of differencing is denoted by *d*. Most real-world nonseasonal series can be made stationary by first- or second-order differencing. An ARIMA(p,d,q) model can be written as

$$\phi(B)(1 - B)^d x_t = \theta(B) \epsilon_t \tag{7}$$

Occasionally, a time series includes a repeating pattern recurring at a fixed period. This characteristic is referred to as seasonality. A seasonal time series can be modeled through the use of seasonal differencing and seasonal AR and MA parameters as needed. Such a model is called a seasonal autoregressive integrated moving average (SARIMA) model. A SARIMA $(p,d,q)(P,D,Q)_S$ model is defined as

$$\phi(B)\Phi\left(B^S\right)(1 - B)^d \left(1 - B^S\right)^D x_t = \theta(B)\Theta\left(B^S\right)\epsilon_t \tag{8}$$

where $\Phi(B^S)$, $(1 - B^S)^D$, and $\Theta(B^S)$ denote seasonal AR, seasonal differencing, and seasonal MA polynomials, respectively; and S denote seasonal interval.

Traffic flow data series typically exhibit a repetitive pattern week by week (*2*). This repetitive pattern must be acknowledged and exploited in traffic flow modeling and forecasting. In other words, general traffic condition data series must be modeled as seasonal time series with the seasonal period of 1 week. Previous research has established the model form SARIMA $(1,0,1)(0,1,1)_S$ as a robust parametric model for general freeway data series (see Williams [*2*], Smith et al. [*3*], and Williams and Hoel [*4*]). The SARIMA $(1,0,1)(0,1,1)_S$ model can be written as

$$(1 - \phi B)\left(1 - B^S\right) V_t = (1 - \theta B)\left(1 - \Theta B^S\right) \epsilon_t \tag{9}$$

where

- V_t = 15-min flow series,
- ϵ_t = error series (assumed to be white noise),
- S = seasonal factor (e.g., with a 15-min traffic flow series there are 672 15-min time periods in a week),
- ϕ = short-term AR parameter,
- θ = short-term MA parameter, and
- Θ = seasonal MA parameter.

The model has a simple and elegant physical interpretation. It consists of two components: a seasonal component and a nonseasonal or short-term component. In decomposed form, the seasonal filter generates a series of deviations from a smoothed nominal seasonal value, and the nonseasonal filter models these short-term deviations. Mathematically these constituent components can be expressed as

seasonal component, $\text{IMA}(1,1)_S$

$$\left(1 - B^S\right) x_t = \left(1 - \Theta B^S\right) a_t \tag{10}$$

short-term component, ARMA(1,1)

$$(1 - \phi B) a_t = (1 - \theta B) \epsilon_t \tag{11}$$

where a_t is the short-term deviation from the seasonal average.

Both of these components involve exponentially weighted smoothing (*5*). This is evident when the components are written in their π-weight form $[1 - \pi(B) = \phi(B)/\theta(B)]$ (*6*). The seasonal part, IMA $(0,1,1)_S$, is equivalent to simple exponential smoothing at a seasonal lag with the smoothing parameter, α, equal to 1 – Θ. Therefore, in π-weight form, the IMA $(0,1,1)_S$ model can be written as

$$\hat{x}_t = (1-\Theta)x_{t-S} + \Theta(1-\Theta)x_{t-2S} + \Theta^2(1-\Theta)x_{t-3S} + \cdots$$

or recursively as

$$\hat{x}_t = (1-\Theta)x_{t-S} + \Theta\hat{x}_{t-S} \tag{12}$$

Similarly, the short-term part, ARMA (1,1), can be written as

$$\hat{x}_t = (\phi-\theta)x_{t-1} + \theta(\phi-\theta)x_{t-2} + \theta^2(\phi-\theta)x_{t-3} + \cdots$$

or recursively as

$$\hat{x}_t = (\phi-\theta)x_{t-1} + \theta\hat{x}_{t-1} \tag{13}$$

where $\hat{x}_t$ denotes the expected value at time t of the respective series.

Thus, the seasonal component provides an exponentially smoothed series based on previous flow observations at 1-week lags. The short-term component provides an exponentially smoothed estimate of the deviations from the historically smoothed series. Furthermore, as ϕ approaches a value of 1, the short-term component approaches simple exponential smoothing and the model approaches the Box and Jenkins airline model SARIMA $(0,1,1)(0,1,1)_S$ (*7*).

Adaptive Seasonal ARIMA Model

ML estimation uses nonlinear optimization methods to maximize the likelihood function (or more precisely minimize the negative of the log-likelihood function) (*8*). As mentioned above, once parameters have been determined using a training data set, they remain static unless and until the ML estimation is repeated on new training data.

This paper proposes an adaptive parameter estimation method through the use of adaptive filters. The basic idea is to update parameters with time so that they are continuously improved through exploiting the information gained from new data. The investigation of adaptive prediction methods was motivated by the following anticipated advantages:

- Parameters could be updated immediately after new data are obtained.
- Forecast accuracy would be comparable to that obtained through ML-based predictors.
- Missing data and outliers could be handled easily in an adaptive framework.
- No initial estimation data set would be necessary because the model starts from initial parameter values and quickly converges to near-optimal values.
- Implementation would be straightforward and plug-and-play, thereby obviating the need for manual parameter estimation.

State Space Representation of Seasonal ARIMA Model

In control systems terminology, the state of a system is the set of variables required to describe that system (*9*). For seasonal ARIMA model, the state is simply the parameter set of the model. Before adaptive filters can be applied, the model must be represented in state space form. A model is in state space form if it can be described by the two state space equations: observation equation and state transition equation. The observation equation relates the state of the system to its inputs and outputs. The transition equation describes how the state propagates with time.

The seasonal ARIMA $(1,0,1)(0,1,1)_S$ model with a nonzero mean term (denoted by c) can be written as

$$(1-\phi B)(1-B^S)V_t = c + (1-\theta B)(1-\Theta B^S)\epsilon_t \tag{14}$$

If y_t denotes the seasonally differenced flow series, $y_t = (1 - B^S)V_t$, the model can be rewritten as

$$y_t = c + \phi y_{t-1} - \theta\epsilon_{t-1} - \Theta\epsilon_{t-S} + \theta\Theta\epsilon_{t-S-1} + \epsilon_t \tag{15}$$

and in vector form as

$$y_t = [1 \quad y_{t-1} \quad -\epsilon_{t-1} \quad -\epsilon_{t-S}][c \quad \phi \quad \theta \quad \Theta]' + \theta\Theta\epsilon_{t-S-1} + \epsilon_t \tag{16}$$

For an adaptive model, the parameters become time varying as denoted by their subscripts, for example,

$$y_t = [1 \quad y_{t-1} \quad -\epsilon_{t-1} \quad -\epsilon_{t-S}][c_t \quad \phi_t \quad \theta_t \quad \Theta_t]' + \theta_t\Theta_t\epsilon_{t-S-1} + \epsilon_t \tag{17}$$

Thus the model has the following state space representation:

Observation equation

$$y_t = Z_t\alpha_t + d_t + \epsilon_t \tag{18}$$

State transition equation

$$\alpha_t = \alpha_{t-1} + \eta_t \tag{19}$$

where

state vector α_t = parameter set $[c_t \quad \phi_t \quad \theta_t \quad \Theta_t]$;
observation vector y_t = weekly differenced series $(1 - B^S)V_t$;
observation matrix $Z_t = [1 \quad y_{t-1} \quad -\epsilon_{t-1} \quad -\epsilon_{t-S}]$;
transition matrix = identity matrix (signifying random walk);
input vector d_t = nonlinear component $\theta_t\,\Theta_t\,\epsilon_{t-S-1}$;
observation error ϵ_t = iid(0, H);
transition error η_t = iid(0, Q);
observation error variance = H; and
transition error covariance matrix = Q.

Once the model has been translated into state space form, adaptive filtering methods can be applied to the parameter states. Three candidate adaptive filters are investigated using the above representation. The complete state space representation is only required for the KF. For the RLS filter, some simplifying assumptions can be made to the above state space representation. The LMS filter requires an even simpler representation. The adaptive filters and their application to the seasonal ARIMA model are described in the following section.

ADAPTIVE FILTERS

A filter functions as an estimator for a dynamic process. A filter can be generally defined as any data processing algorithm used to extract information about a quantity of interest from an error-corrupted process. For the present case, the quantity of interest is the parameter set for the seasonal ARIMA model. All filters optimize a cost function in order to get suitable parameter estimates. This cost function for the investigated filters is the sum of squared errors. However, unlike the ML method (which optimizes the likelihood function), adaptive filters provide a recursive parameter updating solution. For each of the adaptive filters described below, the SARIMA parameters are updated after every observation. At each 15-min time step, the filter equations use the current state and error to calculate new parameter estimates.

Before going into the details for the candidate filters, it is worthwhile to summarize the underlying estimation theory behind adaptive filtering, including the Weiner-Hopf equation (*10*). The Weiner-Hopf equation gives a solution to the quadratic sum-of-squared-error cost function. All filters utilize the Wiener-Hopf solution in one form or another. The Kalman and RLS filters are based on a second-order (gradient and Hessian) solution to this equation. In contrast, the LMS filter is based only on the first-order (stochastic gradient) method. Thus KF and RLS have faster rates of convergence than does LMS because they are based on a second-order solution. On the other hand, LMS estimation is computationally simpler than the second-order filters (*10*). The descriptions below begin with the most generalized and powerful linear filter, the KF, and then proceed to the two simpler ones, RLS and LMS.

Kalman Filter

The KF is an algorithm that provides an efficient recursive solution to the least-squares problem (*9*). KF is a second-order filter which retains information about both the first moment (mean) and the second moment (covariance) of the state vector based on all the observations up to time *t*. After each new observation, mean and covariance of the state vector are updated. The KF equations applicable to the problem at hand are presented below. Here, the notation $(t \mid t-1)$ denotes the value at time *t* given the observations up to time $t-1$.

The first three equations are the predictor equations (before observation at time *t* is available).

State mean prediction

$$a_{t|t-1} = a_{t-1} \tag{20}$$

State covariance prediction

$$P_{t|t-1} = P_{t-1} + Q \tag{21}$$

Kalman gain

$$K_t = \frac{P_{t|t-1}Z_t'}{Z_t P_{t|t-1} Z_t' + H} \tag{22}$$

where

a_t = state estimate at time *t*, that is, $E(\alpha)$;
P_t = state covariance estimate at time *t* (covariance matrix of the state vector); and
K_t = Kalman gain.

The Kalman gain can be thought of as the weight given to the most recent observation for updating mean and covariance of the state. The two corrector equations update mean vector and covariance matrix after observation at time *t* is available.

State update

$$a_t = a_{t|t-1} + K_t \epsilon_t \tag{23}$$

Covariance update

$$P_t = P_{t|t-1} - K_t Z_t P_{t|t-1} \tag{24}$$

RLS Filter

The RLS filter can be derived as a special case of the more general Kalman filter (*10*). The difference between the two filters lies in the treatment of the state process. Kalman filter treats state propagation as a stochastic process, whereas RLS filters model state as a deterministic process. Thus there is no transition error, η_t, and consequently no transition error covariance in the RLS equations. Therefore, the three predictor equations for RLS reduce to

State mean prediction

$$\alpha_{t|t-1} = \alpha_{t-1} \tag{25}$$

State covariance prediction

$$P_{t|t-1} = \frac{P_{t-1}}{\lambda} \tag{26}$$

Stepwise gain

$$K_t = \frac{P_{t|t-1}Z_t'}{Z_t P_{t|t-1} Z_t'} = \frac{P_{t-1}Z_t'}{Z_t P_{t-1} Z_t' + \lambda} \tag{27}$$

The forgetting factor, λ, determines how fast the weights decay. The inverse of $(1-\lambda)$ gives the memory of the filter. In simple terms, λ determines how many past observations to consider for doing parameter estimation. For $\lambda = 0.99$, memory is 100, which means that only the past 100 observations play a role in determination of parameters. If $\lambda = 1$, then the problem reduces to that of ordinary least squares where memory is infinite and all past observations are given equal weights while determining the estimates. In other words, the cost function minimized by the RLS filter is the exponentially weighted sum of squared errors.

The RLS algorithm can be completely described by the following two equations:

State update

$$\alpha_t = \alpha_{t-1} + \frac{P_{t-1}Z_t'}{Z_t P_{t-1} Z_t' + \lambda} \epsilon_t \tag{28}$$

Covariance update

$$P_t = \frac{P_{t-1} - \dfrac{P_{t-1}Z_t'Z_t P_{t-1}}{Z_t P_{t-1} Z_t' + \lambda}}{\lambda} \tag{29}$$

LMS Filter

The LMS algorithm belongs to the family of stochastic gradient filters. The algorithm uses an approximate estimate of the gradient at each time step to get nearer to the optimal solution of the mean-square-error (MSE) function (*10*). The quadratic MSE surface has optimal parameter values at its bottom-most point. The key feature of LMS algorithm is that it uses a rough approximation to the gradient and avoids calculation of true gradient (which requires considerably more computation). Once the function converges to its minimum, the parameters execute a random motion near the optimal point.

LMS consists of a single equation. Because it is a first-order filter, it updates only the state mean and ignores calculation of Hessian and state covariance, which are second-order characteristics. The state update equation is

$$\alpha_t = \alpha_{t-1} + \mu Z_{t-1}\epsilon_{t-1} \qquad (30)$$

The step-size factor, μ, determines the magnitude of change made at each step to the state vector. A large value of μ provides higher convergence but can give rise to instability. On the other hand, a small value ensures stability but makes convergence slower.

DATA

The data used in this research come from the Motorway Incident Detection and Automatic Signalling (MIDAS) system in United Kingdom. Archived data for the entire year of 2002 were provided by the United Kingdom Highway Agency. The archived observational data include lane-by-lane flow, speed, and occupancy at 1-min discrete intervals. The adaptive filter investigation used flow data for the entire year from eight different sites.

Data Aggregation

The 1-min flow data were aggregated to 15-min discrete flow rate series. In performing the aggregation, a protocol was required for dealing with missing 1-min observations. For the 15-min discrete series, a threshold of two-thirds nonmissing values was selected. In other words, if any 15-min interval had 10 or more nonmissing 1-min values (greater than or equal to two-thirds), then the average flow rate during the nonmissing intervals was taken as the 15-min flow rate. However, if the number of nonmissing values was less than 10, then the 15-min flow rate for that time interval was set as missing. For the MIDAS data used, there were no instances where observations were missing by lane. In other words, either data were available for all lanes or data were missing for all lanes.

Data Cleansing

Traffic data are prone to outliers and missing observations. These data anomalies may result from either measurement error (such as would result from malfunctions of the detection equipment or communications system) or from actual, though abnormal, traffic conditions (such as a capacity-reducing incident). Because this study is concerned with comparing the performance of the candidate adaptive methods for SARIMA model parameter estimation and because the presence of outliers will bias the parameter estimates, these outliers were identified and removed from the 15-min flow data. An iterative procedure was used for detecting and replacing outliers. This procedure was developed for previous large data set traffic flow forecasting research (*2*) and is based on the outlier detection and estimation work of Chen and Liu (*11*). The procedure iteratively flags possible outliers with a pulse indicator variable and reestimates the parameters while jointly estimating the magnitude of each outlier in the current candidate outlier set. At each iteration, the outlier set is updated by using separate, predefined statistical test levels for adding and removing observations from the candidate outlier set.

After applying this procedure to the 15-min traffic flow series, the outlying observations were adjusted by the final estimate of the outlier magnitude and the missing values were replaced by the final model forecasts (similar to interpolation but following the fitted process model). This provided a set of continuous series without missing values and (time series) outliers. These data sets will be referred to as the outlier-corrected data sets, and the uncorrected ones will be referred to as raw data sets. The data set attributes for the eight investigated sites are given in Table 1. All the investigated sites are on urban freeways in the London metropolitan area. M25 is the London orbital motorway, and M6 and M1 are radial motorways.

As Table 1 clearly illustrates, the percentages of missing and outlying observations were not high (less than 5% in most cases). If the adaptive filters developed in this research were used in an online system, missing values would of course add no new information, and therefore updating would not occur for time intervals with missing observations. For outliers, the parameter updating could either continue in the presence of possible outliers (likely a reasonable option if the rate of outlier occurrence were low), or the parameter updating could be suspended for intervals with observations flagged as possible outliers. Comparative evaluation of these online updating options will

TABLE 1 Data Set Attributes

Site	Motorway	Lanes	Series Length	Percentage Missing	Percentage Outliers	Total (%) (missing + outliers)
4762A	M25	4	35,040	3.07	2.85	5.92
4680B	M25	4	35,040	3.97	1.52	5.49
4565A	M25	4	35,040	3.10	1.50	4.60
6951A	M6	3	35,040	1.57	1.63	3.20
6954B	M6	3	35,040	1.69	1.62	3.30
2737A	M1	3	30,912	1.68	1.24	2.91
2808B	M1	3	30,912	1.08	1.89	2.97
4897A	M1	3	30,912	2.77	1.39	4.16

be important follow-on research. Furthermore, outliers in the data stream can either be actual innovations to the time series process, such as would result from a capacity-reducing incident, or a spurious observation (i.e., measurement error). Time series models such as those used in this study cannot distinguish between these two possibilities alone but can play a role in helping to identify and quantify incident effects and measurement error when combined with other traffic management system information.

RESULTS

Sensitivity of SARIMA Parameters

As discussed in the section on the time series model, the SARIMA $(0,1,1)(0,1,1)_S$ model provides a type of seasonal exponential smoothing. The three parameters ϕ, θ, and Θ determine the weighting pattern applied to past observations. In practice, the SARIMA $(0,1,1)(0,1,1)_S$ model is quite robust in that for a given traffic flow data stream there is a relatively wide range of parameter values where the model is stable and the root-mean-square error (RMSE) surface is relatively flat. For the MIDAS data used in this study, these ranges were observed to be 0.85 to 0.95 for ϕ, 0.1 to 0.4 for θ, and 0.8 to 0.95 for Θ (*6*). As long as parameters fall in these ranges, the RMSE results lie close to the minima.

The parameter ranges given above were determined through a detailed parameter sensitivity analysis at each analysis site. Parameter sensitivity at each site was investigated by varying two parameters at a time by 0.01 increments while holding the third parameter fixed at its RMSE optimal value. Forecasts and RMSE results for the entire data set were generated using all corresponding parameter value combinations. For Site 4565A, for example, the minimum RMSE was 226.67 with optimal parameter values at $\phi = 0.94$, $\theta = 0.28$, and $\Theta = 0.92$.

Filter Attributes

Each of the filters has its own attributes which need to be determined. RLS and LMS require only one attribute, the value of forgetting factor, λ, and the step-size parameter, μ, respectively. For the Kalman filter, reasonable values must be determined for the observation error covariance, H, and state transition error covariance matrix, Q. The filter attribute values (*6*) used are as follows: forgetting factor, $\lambda = 0.9998$; step-size parameter, $\mu = 3 \times 10^{-7}$; observation error variance, $H = 200^2$; and state transition error covariance matrix,

$$Q = \begin{bmatrix} 5\times10^{-4} & 0 & 0 & 0 \\ 0 & 3\times10^{-8} & 0 & 0 \\ 0 & 0 & 1\times10^{-7} & 0 \\ 0 & 0 & 0 & 1\times10^{-6} \end{bmatrix}$$

The KF parameter, H, was set based on the observed RMSE for the data series. The parameter, Q, was set based on multiple trial-and-error runs with the 15-min data sets. The matrix values given above were found to provide robust tuning of the rate of parameter change. No formal estimation procedure was used. The trial-and-error selection was guided generally by balancing the objectives of responsiveness and stability. Although these static values for H and Q provided good results as is demonstrated below, further research is needed to determine the benefit of allowing H and Q to be time varying.

Comparison Between Filter Estimates and ML Estimates

In order to compare the results of adaptive filters with those obtained from ML, a pseudo-adaptive moving-window approach was applied (*6*). In this approach, ML estimates were first obtained for initial 8-week data (window). This window was then moved one observation at a time through the final observation, and ML estimation was performed on each intermediate window. This brute-force moving-window method, although very time consuming, provides a good benchmark against which the adaptive filter results can be compared. Figures 1 through 3 illustrate the parameter comparison for all four methods for the 2002 data series from Site 4565A.

The statistical data analysis software SAS (Version 9.0) was used to implement all the models. The SAS ARIMA procedure provides a powerful routine for computing the ML parameter estimates. However, the proposed adaptive algorithms can be easily implemented using any standard programming or scripting language because the filters are comprised solely of recursive systems of simple linear equations.

The adaptive parameter estimates track well with the 8-week ML estimates in all cases, especially when considered in light of the parameter sensitivity analysis discussed above. In general, the KF estimates track the ML estimates more closely, especially with respect to the seasonal MA parameter (see Figure 3).

As a rigorous test of the plug-and-play capabilities of the adaptive SARIMA forecasting, the KF was initialized with all parameters set to zero rather than to informed reasonable initial values. Figure 4 shows the convergence of the SARIMA parameters using the Kalman filter for the initial 5 weeks of data at Site 4565A. Notice the rapid convergence of the short-term parameters to their optimal range. In fact, the seasonal MA parameter converges just as rapidly once observational data become available to the seasonal process. As a characteristic of the seasonal model form, approximately 2 weeks of observations are required before updating of the seasonal MA parameter can begin.

Comparison Between Filter Forecasts and ML Forecasts

The effectiveness and robustness of the adaptive filters are further reinforced by goodness-of-fit comparisons of the four adaptive methods with one-step forecasts from a model using ML-based parameters from the entire year. Note that this is a challenging test for the adaptive methods because the ML forecasts involve applying a forecast model back to the data series used to estimate the model parameters. The adaptive filters, however, generated forecasts at each interval based only on the information provided by the prior observations. Tables 2 and 3 provide the comparison of predictive performance. The table column headings indicate the following methods:

- Maximum likelihood estimates based on fitting to the entire data set (ML),
- Adaptive estimates obtained from KF,
- Adaptive estimates obtained from recursive least-squares filter (RLS), and
- Adaptive estimates obtained from least-mean-squares filter (LMS).

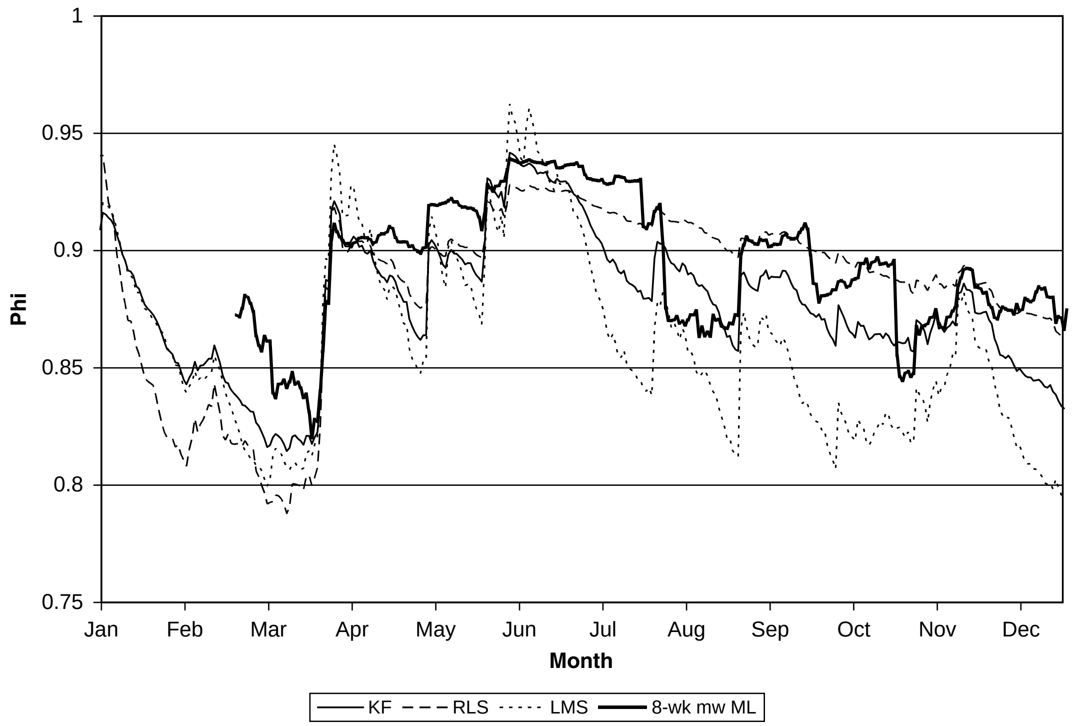

FIGURE 1 Estimates for short-term autoregressive parameter, ϕ.

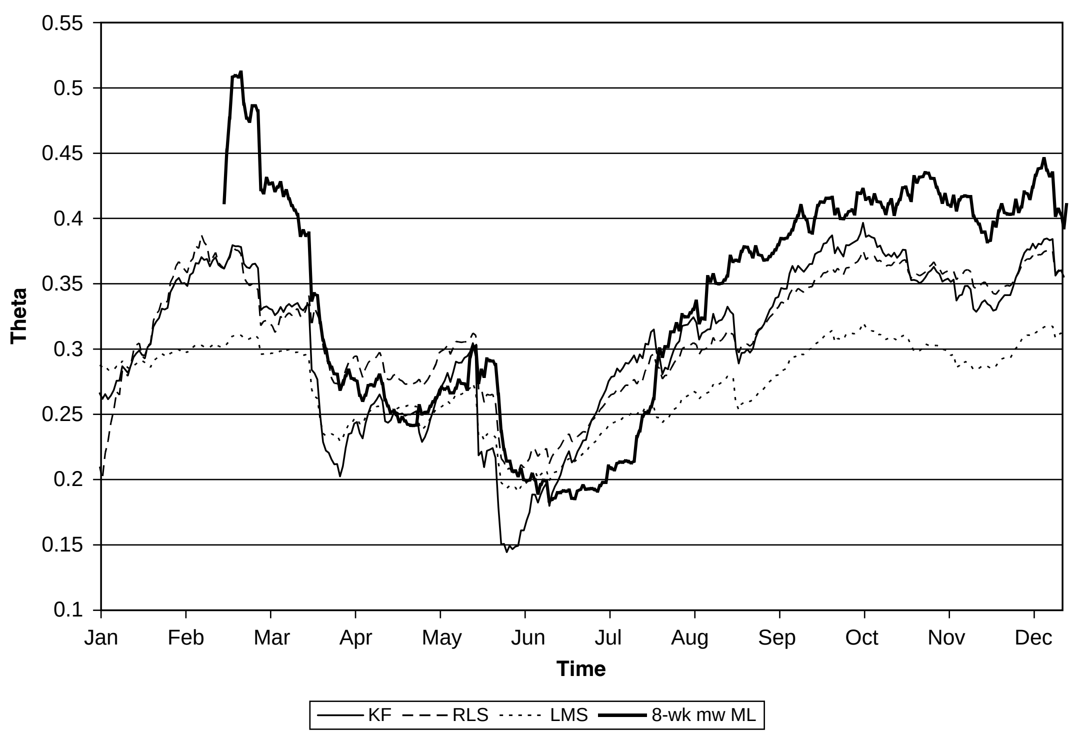

FIGURE 2 Estimates for short-term moving average parameter, θ.

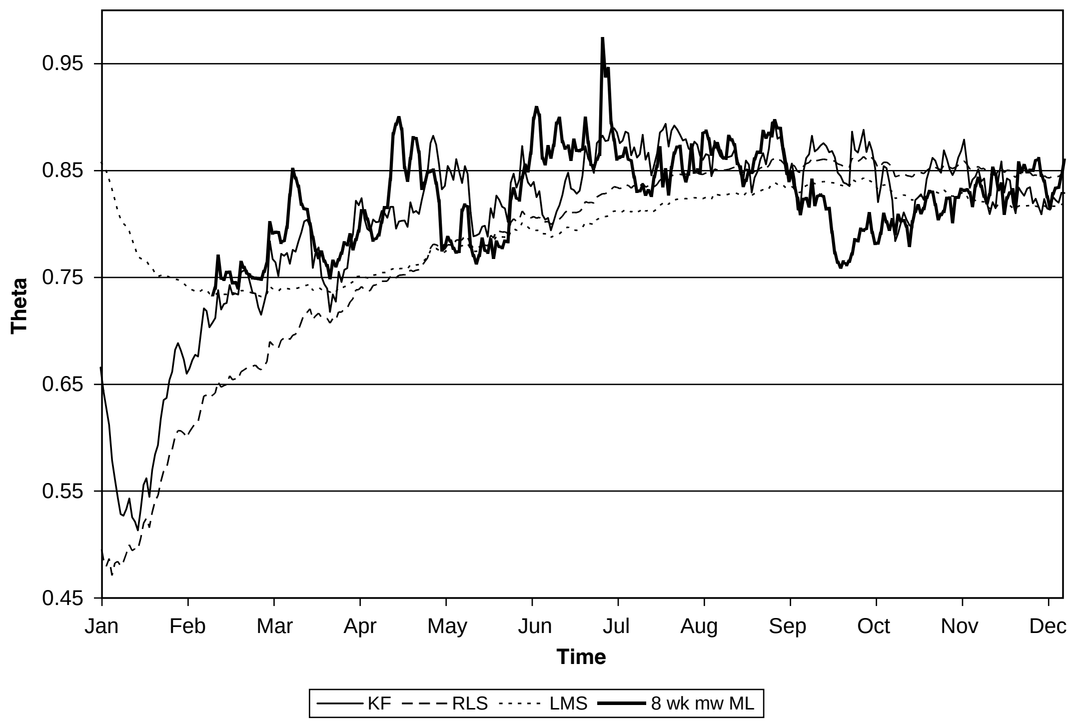

FIGURE 3 Estimates for seasonal moving average parameter, Θ.

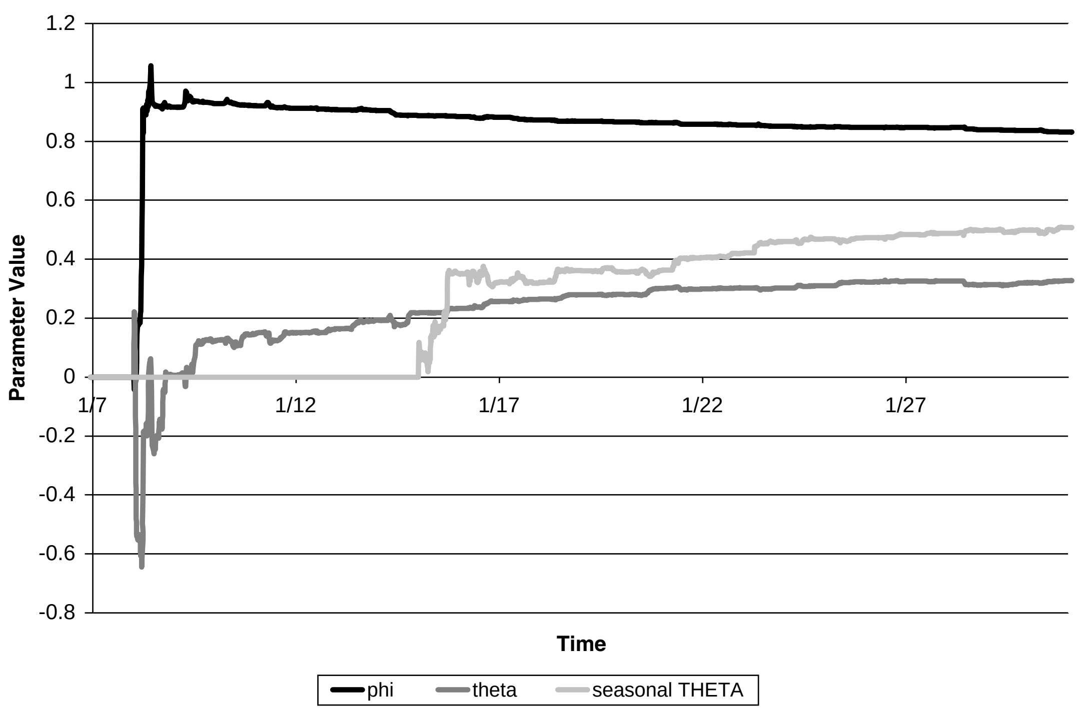

FIGURE 4 Convergence of SARIMA parameters with Kalman filtering.

TABLE 2 RMSE of Forecasts

Site	ML	KF	RLS	LMS
4762A	307.45	306.29	308.32	307.48
4680B	220.57	221.22	222.23	222.67
4565A	228.34	228.12	230.45	230.52
6951A	215.23	217.02	217.44	218.45
6954B	174.16	175.02	175.52	175.96
2737A	208.35	210.08	211.10	212.30
2808B	201.18	202.85	203.43	204.04
4897A	198.23	200.50	201.61	201.29

TABLE 3 MAPE of Forecasts

Site	ML (%)	KF (%)	RLS (%)	LMS (%)
4762A	7.19	7.31	7.32	7.23
4680B	6.36	6.49	6.42	6.38
4565A	7.15	7.23	7.23	7.18
6951A	8.36	8.62	8.57	8.47
6954B	7.84	7.92	7.93	7.90
2737A	7.26	7.39	7.43	7.39
2808B	8.19	8.34	8.38	8.35
4897A	7.2	7.34	7.37	7.31

Table 2 provides the RMSE results. The results for all the adaptive filtering methods compare favorably with the ML estimated model with the KF forecasts providing the best adaptive forecasts at all sites. The KF adaptive filter actually outperformed the ML estimated model on the two most volatile series. This result attests to the robustness provided by the KF method's ability to adapt to more dramatic shifts in the parameter state during the year.

Table 3 presents another measure, the mean absolute percentage error (MAPE), defined as

$$\frac{\sum_{i=1}^{n}\frac{\left|V_t-\hat{V}\right|_t}{V_t} * 100}{n} \tag{31}$$

where n denotes number of observations.

Since absolute percent error can be skewed due to low and near-zero values, observations below 100 vehicles per hour were excluded while calculating MAPE. The ML-based predictions are best in terms of MAPE for all sites. However, as with RMSE, the results are comparable in all cases.

Figure 5 shows the MAPE of forecast by week for the four methods. Overall, the MAPE lies near or below 8% except for the first and last week of the year, where traffic pattern changes drastically due to the holiday season.

CONCLUSIONS

The research presented in this paper employs dynamic filter theory to provide an adaptive seasonal time series model for forecasting short-term traffic flow. In general, this approach can be applied to other parametric models to achieve adaptability. The crucial step in applying this approach is to represent the underlying parametric model in an appropriate state space form.

Three well-known adaptive filters have been successfully applied to the SARIMA $(0,1,1)(0,1,1)_S$ traffic flow forecast model. All three filters—the Kalman filter, the recursive least-squares filter, and the least-mean-squares filter—are able to track the parameters correctly. Results show that forecast performance obtained from filters is close to that provided by ML estimated models even when

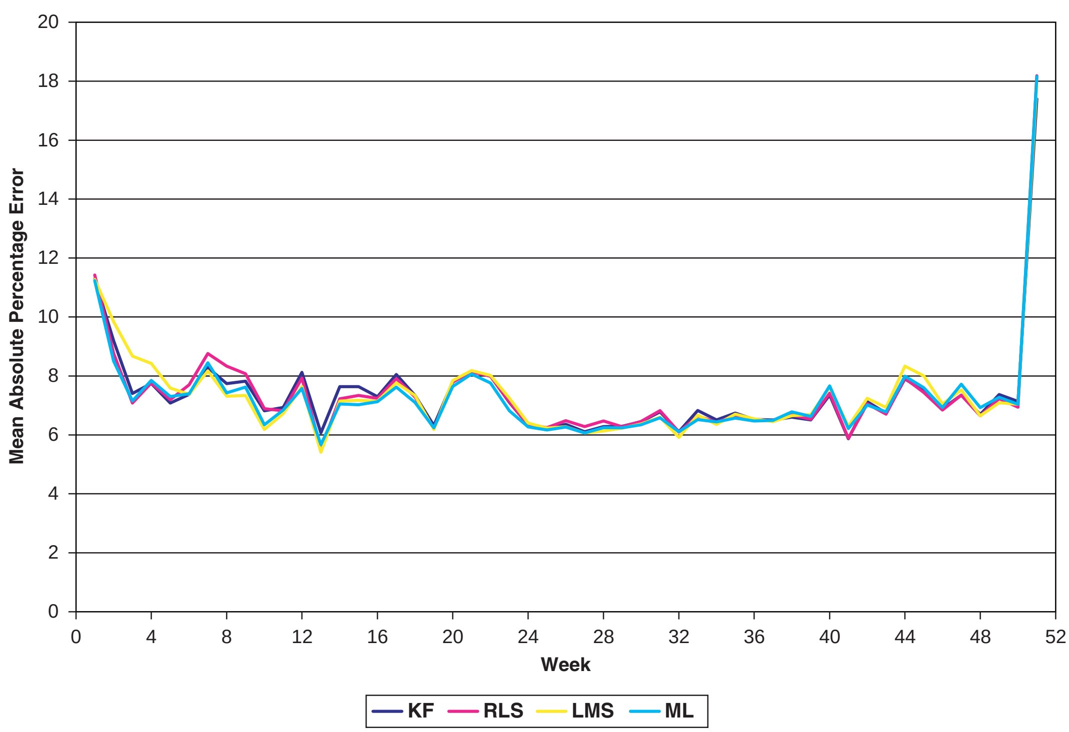

FIGURE 5 MAPE by week for Site 4565A.

the ML models are evaluated in terms of goodness of fit to the training data. The Kalman filter provides the most rich and accurate adaptive framework without a significant increase in computation burden and thus is recommended as the preferred adaptive filtering scheme.

This paper also establishes the robustness of the SARIMA model. Parameter sensitivity analysis shows that the model forecasts are stable for a relatively wide range of parameter values.

In summary, this research provides theoretically and empirically sound methods for performing adaptive SARIMA parameter estimation and forecasting. The proposed adaptive methods have the advantage of enabling plug-and-play implementation in a real-time system. Therefore, traffic management system software will now be able to exploit the superior predictive performance of SARIMA-based forecasts without having to perform site-by-site model fitting and without having to reestimate model parameters at periodic intervals. Given the results of this study and the findings of foregoing research on SARIMA (*2–4*), the expectation that the proposed adaptive methods will perform well in freeway management systems is high. More empirical study is needed to establish the efficacy of adaptive seasonal time series methods for arterial and surface street management systems.

ACKNOWLEDGMENTS

This work was supported by a grant from the National Science Foundation. The authors thank the United Kingdom Highway Agency for providing the data used in this research.

REFERENCES

1. Brockwell, P. J., and R. A. Davis. *Introduction to Time Series and Forecasting.* Springer-Verlag, Inc., New York, 1996.
2. Williams, B. M. *Modeling and Forecasting Vehicular Traffic Flow as a Seasonal Stochastic Time Series Process.* PhD dissertation. University of Virginia, Charlottesville, 1999.
3. Smith, B. L., B. M. Williams, and R. K. Oswald. Comparison of Parametric and Non-parametric Models for Traffic Flow Forecasting. *Transportation Research Part C,* Vol. 10, 2002, pp. 303–321.
4. Williams, B. M., and L. A. Hoel. Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. *Journal of Transportation Engineering,* Vol. 129, No. 6, 2003, pp. 664–672.
5. Williams, B. M., P. K. Durvasula, and D. E. Brown. Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models. In *Transportation Research Record 1644,* TRB, National Research Council, Washington, D.C., 1998, pp. 132–141.
6. Shekhar, S. *Recursive Methods for Forecasting Short-Term Traffic Flow Using Seasonal ARIMA Time Series Model.* MS thesis. North Carolina State University, Raleigh, 2004.
7. Box, G. E. P., and G. M. Jenkins. *Time Series Analysis: Forecasting and Control.* Holden-Day, San Francisco, Calif., 1976.
8. Harvey, A. C. *Time Series Models,* 2nd ed. The MIT Press, Cambridge, Mass., 1993.
9. Balakrishnan, A. V. *Kalman Filtering Theory,* 2nd ed. Optimization Software, Inc., New York, 1987.
10. Haykin, S. *Adaptive Filter Theory,* 4th ed. Prentice-Hall, Englewood Cliffs, N.J., 2002.
11. Chen, C., and L. Liu. Joint Estimation of Model Parameters and Outlier Effects in Time Series. *Journal of the American Statistical Association,* Vol. 88, 1993, pp. 284–297.

The Information Systems and Technology Committee sponsored publication of this paper.