Research Article

Fuzzy Prediction for Traffic Flow Based on Delta Test

Yang Wang, Yanyan Chen, and Jianhui Lai

Beijing Key Lab of Traffic Engineering, Beijing University of Technology, 100 Pingleyuan, Chaoyang District, Beijing 100124, China

Correspondence should be addressed to Yanyan Chen; cdyan@bjut.edu.cn

This paper presents a novel approach to one-step-forward prediction of traffic flow based on fuzzy reasoning. The successful construction of a competent fuzzy inference system of Sugeno type largely relies on proper choice of input dimension and accurate estimation of structure parameters and rules. The first issue is addressed with a proposed method, based on δ-test, which can simultaneously determine input dimension and reduce noise level. In response to the second issue, two clustering techniques, based on nearest-neighbor clustering and Gaussian mixture models, are successively employed to determine the antecedent parameters and rules, and the estimation for the consequent parameters is achieved by the least square estimation technique. A number of experiments have been performed on the one-week data of traffic flow to evaluate the proposed approach in terms of denosing, prediction performances, overfitting, and so forth. The experimental results have demonstrated that the proposed prediction approach is effective in removing noise and constructing a competent and compact fuzzy inference system without significant overfitting.

1. Introduction

Since intelligent transportation systems (ITS) emerged, research efforts have been continuously devoted to traffic flow prediction, as live and accurate prediction on traffic flow is a premise for efficient traffic management and control. While a wide spectrum of prediction techniques, such as Kalman filter [1] and its extension [2], support vector machine [3], Bayesian networks [4], and hybrid approach [5–8], has been reported in literature, the methodology behind these techniques is based either on influencing factors or on historical data of traffic flow. The influencing factors (e.g., weather) are used as inputs for the underlying traffic system which outputs traffic flows to reflect traffic states. The correlation between the inputs and outputs, established using either analytical model or data-derived approach, is used as the primary reasoning for future traffic flow. On the other hand, the methodology behind the historical data-based approaches is different from those based on influencing factors and the fundamental idea is that the future depends on its past. For a sequence of traffic flow observations with equal sampling interval, called traffic flow time series, the dependent relation between the traffic flow to be predicted at t and its past $x_{t-1}, x_{t-2}, \ldots, x_{t-d}$ can be formulated as follows:

$$x_t = f\left(x_{t-1}, x_{t-2}, \ldots, x_{t-d}\right) + r_t, \tag{1}$$

where the term r_t, called residual in this paper, is the part that cannot be accounted by the model f, due to either a lack of functional determination or real noise. Therefore, the term *residual* in this paper is defined to include modeling error and real noise.

The prediction performance by the methods based on influencing factors is considerably dependent on the quality of data collected for different factors which may not be guaranteed always. Therefore, in our ongoing work, the methodology based on historical data is adopted. Also, as only one type of data (i.e., traffic flow) is required, there is no need to manipulate units for different types of data. This paper reports our work on the prediction approach for one step ahead (i.e., short-term).

To model the dependent behavior, a number of techniques have been proposed, ranging from Kalman filter [1, 2], artificial neural network [9], and nonparametric regression

[10] to hybrid approach [11]. Our research has been focused on the modeling approach based on fuzzy inference system (FIS), in that it is tolerant to noise, resistant to uncertainty, and easy to incorporate expert and field knowledge [12]. However, research activity in this area is relatively silent, reflected by a few notable contributions reported in literature over the last decade. Zhang and Ye proposed a prediction methodology by using fuzzy logic system to fuse the outputs of two methods out of autoregressive integrated moving average, backpropagation neural networks, exponential smoothing method, and Kalman filter, resulting in four different combinations [6]. Similar idea has been adopted in [7], but the two methods mixed by a fuzzy logical model are history mean and artificial neural network models. Paper [8] describes a hybrid methodology that two fuzzy rule-based systems are constructed, one providing the next flow estimation based on the current flow only and the other predicting the one-step-ahead flow based on the current flow at the current location and the upstream location. A genetic algorithm is used to tune the parameters for the fuzzy rule-based systems by minimizing the mean absolute relative error between the estimated and the observed values. Paper [13] presents a prediction approach for short-term traffic flow prediction primarily based on a Sugeno fuzzy system (also known as TSK fuzzy system). The initial structure is formed by partitioning the input vector space by the mean shift clustering algorithm and subsequently optimized to eliminate redundant structure by the mean firing technique, and finally the other parameters are determined by particle swarm optimization with the aim of minimizing root mean squared error (RMSE).

A multiple-inputs (or single input vector) and single-output (MISO) FIS captures the conditional dependence between inputs and output, $P(x_t \mid x_{t-1}, x_{t-2}, \ldots, x_{t-d})$, using the following rules with fuzzified inputs:

$$R_i: \quad \text{if } x_{t-1} \text{ is } A_{i1} \text{ and } x_{t-2} \text{ is } A_{i2} \text{ and } \cdots x_{t-d} \text{ is } A_{id} \\ \text{then } x_t \text{ is } B_i \text{ or } f_i(\mathbf{x}). \tag{2}$$

Whether the consequent part in fuzzy rules (as described above) is based on fuzzy sets or linear functions of input vector yields two common types of FISs, namely, Mamdani and Sugeno FISs. Although the prediction method proposed in this paper adopts the Sugeno approach, it is interesting to investigate the predictability of Mamdani FIS as well.

To create a competent Sugeno FIS, careful decisions should be made on the following steps: (1) the input dimension: although all traffic flow recordings in the past should in principle be taken into account when predicting the future flow, it is not practical to consider the whole set of historical data as the computational complexity grows exponentially with the number of input dimension and curse of dimensionality [14]; (2) parameters and rules: a large number of membership functions can reduce model error, but the model suffers from the computational complexity, again. To compromise the complexity and accuracy, a proper choice should be made on the number of membership functions and the parameters associated should be determined in an optimal or near-optimal fashion. As a Sugeno FIS is adopted
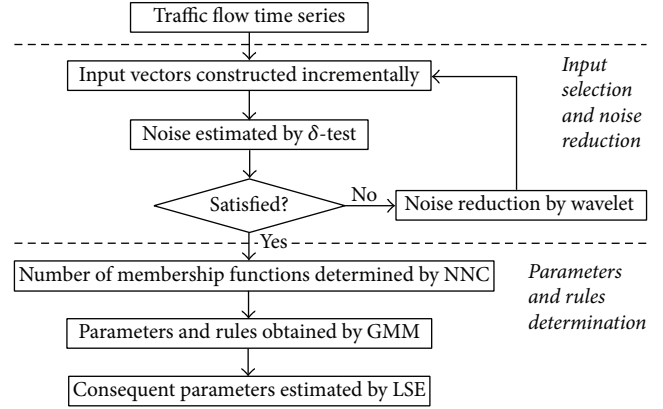

FIGURE 1: Framework of the prediction approach proposed.

as the basis for the prediction, the set of parameters for the consequent part should be optimized to improve modeling performance.

This paper proposes a novel approach to one-step-ahead prediction (e.g., but multistep-ahead prediction can be realized simply by recursive prediction) for traffic flow based on a Sugeno FIS, with a particular emphasis on the above-mentioned issues. Firstly, a nonparametric residual variance estimation method, called δ-test [15], is used to measure the noise level reduced using a wavelet-based denoising technique and the input dimension is determined when the noise reaches a reasonable level. To address the second issue, a clustering is performed using the nearest-neighbor clustering (NNC) method [16] to obtain the number of membership functions and a Gaussian mixture model (GMM) [17] is subsequently applied to determine the parameters associated with membership functions. Finally, the consequent parameters are obtained using the least square estimation (LSE) technique [18].

2. Prediction Approach

2.1. Approach Outline. The prediction approach presented in this paper uses a first-order Sugeno FIS as a basis of the predictor. Such choice has been made due to the fact that Sugeno FIS has been proved to be a universal approximator [19–21]. Additionally, FIS has been reported to be tolerant to noise and resistant to uncertainty [12]. To achieve a concrete implementation for the FIS, we follow the commonly adopted model structure: T-norm for conjunction operations, Gaussian membership functions, linear function for consequent part, and product inference of rules.

Figure 1 schematically shows the approach framework used to build the FIS based on the historical observations of traffic flow. Note that it is assumed that an appropriate preprocess has been taken to exclude any outliers and make up all missing recordings for the traffic flow time series. The algorithm firstly generates a set of input vectors by incrementing the dimension which has been bounded to the maximum dimension specified by users. For the set of input vectors generated, the δ-test is used to estimate the

residual variance for each input vector without explicitly building a model. Based on the estimated residual variances, the algorithm subsequently evaluates whether any input vectors satisfy the requirement set by users. If the evaluation indicates none of them can meet the requirement, which implies the noise level still remains high, the time series is processed by a wavelet-based technique to reduce the noise level. Such process is iteratively performed until the residual error can be reduced to meet the requirement. During the second step, two clustering algorithms (NNC and GMM) are successively employed to determine the number of membership functions and the associated parameters, respectively. Also, the clustering results obtained from the GMM algorithm help determine the rules for the FIS to be constructed. In our approach, the LSE technique is used to determine a set of parameters for consequent part that minimizes the mean square error (MSE) between the model output and training samples. Those steps and how they are specifically implemented are discussed in detail in the following subsections.

2.2. Input Selection and Noise Reduction. A proper selection of input dimension can alleviate the curse of dimensionality problem, while all the mostly related historical states can still be utilized for the prediction. Our approach employs the δ-test to determine the input dimension from a set of candidate input vectors that have been generated using recent recordings and the maximum number of input dimensions is predefined.

The δ-test was first introduced in [22], improved in [23], simplified in [24], and used in [15, 25]. The idea behind all versions of the δ-test is simple and intuitive. As expressed in (1), if the underlying relationship of the system can be completely determined by a smooth function f, r should be real noise which is assumed to be independent to input vector $\mathbf{x}$ (i.e., $(x_{t-1}, x_{t-2}, \ldots, x_{t-d})$) and have zero mean and bounded variance. For a given input vector $\mathbf{x}$ and its nearest neighbor $\mathbf{x}'$, the continuity of regression function implies the outputs $f(\mathbf{x})$ and $f(\mathbf{x}')$ should be close enough in the output space (note that it is not necessary to be the closest pair for the outputs). In other words, if a deterministic (time-invariant) system is fed with the same values at different time instants, the same values should be outputted from the system, apart from the noise. Alternatively, the corresponding outputs are different due to the presence of noise.

This work uses the simplified version of the δ-test, which first searches for the nearest neighbor (its index is denoted as NN) from N points for a given point in the input space according to a distance metric (Euclidean distance is used in this context):

$$NN(i) := \operatorname*{argmin}_{j \neq i} \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2. \tag{3}$$

Then, the variance of r, var$[r]$, can be estimated by

$$\operatorname{var}[r] \approx \delta = \frac{1}{2N} \sum_{i=1}^{N} \left(f(\mathbf{x}_i) - f(\mathbf{x}_{NN(i)}) \right)^2. \tag{4}$$

The result obtained from the δ-test provides a criterion for input selection. The algorithm starts by including only the current observation as the initial dimension of the candidate input vector and the residual variance for such origination is calculated. Then, the dimension increments by considering the next recent recording to form a new candidate input vector which is subsequently examined by the δ-test. Such iterative process terminates when the input dimension exceeds the maximum number predefined.

In contrast to the model-based approaches, it is not necessary to build a model for input dimension determination by using the δ-test to select a competent input vector within the maximum dimension which may require domain knowledge or assistance of a model. However, the result from the δ-test cannot implicitly tell us to what extent an optimal model can fit a noisy time series, as the estimated residual variance encompasses modeling error and real noise.

The data-driven approach is often used to construct a FIS based on a set of training data, whenever a prior knowledge about the system is unavailable or incomplete. The goodness of the model is usually measured by the MSE between the training data and the outputs from the model. If the training data has been corrupted by some noise, there is a risk of overfitting to the noisy training data if the objective is to minimize the MSE and the model will consequently perform poorly on the previously unseen inputs. This is the primary reason that we designed the denoising substage, as the consequent parameters estimated are subject to the minimization of the MSE (described in the next section).

To this end, the well-known denoising technique based on wavelet transform [26] is employed and one-dimensional denoising process is performed directly on the collected time series of traffic flow. In general, the denoising procedure follows the three steps: (1) decomposition: compute the wavelet decomposition of the noisy data by applying a wavelet ("db3" in our case) to it at the specified level; (2) detail coefficients thresholding: select appropriate threshold for each level and apply thresholding method (soft thresholding in our implementation) to remove the noises; (3) reconstruction: inverse wavelet transform of the thresholded wavelet coefficients to generate clean data. Further information on wavelet-based denoising technique can be obtained from the papers [26–28].

The decomposition level is one of the critical factors, which has significant impact on the denoising performance (the discussions on the others, such as wavelet base, are beyond the scope of this paper) [29]. The simplest method is to choose a fixed decomposition level of wavelet transform based on experience before denoising, but it will be problematic when the noise level embedded is not what is expected. Therefore, a number of adaptive approaches have been proposed [30, 31]. Those approaches determine the decomposition level by iteratively examining the denoising performance for the incremented level until the performance meets some stop criteria which are set normally based on either signal-to-noise ratio (SNR) [32] or singular spectrum [30]. However, our method is different from those reported

in literature in that it is based on the residual variance estimated by the δ-test and the input vector can be determined simultaneously.

As shown in Figure 1, a number of candidate input vectors are generated by incrementing the dimension in turn and analyzed via the δ-test. If the minimum residual variance obtained from the set of candidate input vectors does not satisfy the following criterion (called variance ratio R in this paper),

$$R = \frac{\text{var}(s)}{\text{min}(\boldsymbol{\delta})} < T, \tag{5}$$

where s is the original time series or the reconstructed time series (if denoising is performed), $\boldsymbol{\delta}$ is a set of residual variance values corresponding to the set of candidate input vectors, and T is a constant which needs to be predetermined, then the wavelet-based denoising technique is applied to the time series to reduce noise level and the decomposition starts from one level. The denoised time series generates another set of candidate input vectors which are subsequently examined again by the δ-test. If the noise level is still high, the decomposition level increments and the denoising process is performed again. Such iterative process terminates until the stop criterion is met and the input vector that has produced the minimum residual variance will be selected as the input for the FIS.

2.3. Parameters and Rules Determination. The NNC [16] is a simple but effective clustering technique that can be used to determine the number of membership functions and provide initial cluster centers for the subsequent tuning process. The clustering process by the NNC algorithm is as follows: (1) choose the first data from the set of training data as the first cluster center; (2) group the data into the first cluster if its distance to the cluster center is less than the predefined threshold value (0.8 for all experiments presented in this paper); (3) otherwise, set the data as the second cluster center; (4) examine the next data by calculating its distances to all existing cluster centers; (5) group the next data into the cluster whose center is closest to it and their distance is less than the threshold; (6) otherwise, generate a new cluster by setting it as the center. Such process terminates if all data have been checked and the final number of clusters obtained is used as the number of membership functions.

During the tuning process, a GMM is used to group the training data into a set of clusters whose number and initial centers have been determined by the preceding process.

A mixture of K component Gaussians can be expressed as follows:

$$f(\mathbf{x}) = \sum_{k=1}^{K} w_k \cdot f(\mathbf{x}; \mathbf{v}_k, \boldsymbol{\Sigma}_k) = \sum_{k=1}^{K} w_k$$
$$\cdot \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}_k|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{v}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \mathbf{v}_k)\right], \tag{6}$$

where $f(\mathbf{x}; \mathbf{v}_k, \boldsymbol{\Sigma}_k)$ is the probability density function of kth component Gaussian with mean vector $\mathbf{v}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$, and its weight w_k satisfies

$$\sum_{k=1}^{K} w_k = 1, \quad \forall i : w_k \geq 0. \tag{7}$$

To determine the parameter vector $\boldsymbol{\theta}$ (i.e., w_k, $\mathbf{v}_k$, and $\boldsymbol{\Sigma}_k$ for each component) for a Gaussian mixture model, the Expectation Maximization (EM) algorithm is applied to the set of training data. An unobserved latent variable $\mathbf{y}$ is added and this latent variable determines the component from which the observed data $\mathbf{x}$ originates. Thus, the log-likelihood function can be formulated as follows:

$$\ln L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} \ln f(\mathbf{x}_i, \mathbf{y}_i; \boldsymbol{\theta}). \tag{8}$$

The EM algorithm [33] fits a mixture model to a set of training data by alternately performing between expectation step and maximization step. Suppose the current step is at t step. *Expectation Step.* Determine the expectation of the log-likelihood function, with respect to the conditional distribution of $\mathbf{y}$ given $\mathbf{x}$ under the current estimate of the parameters. *Maximization Step.* Update the parameters that maximize the expectation of the log-likelihood function obtained in expectation step. The process terminates either if a predefined number of iterations is exceeded or if the parameters improvement between two consecutive iterations is less than the minimum amount of improvement predefined.

As Gaussian membership functions are adopted for the premises of the FIS, the result from the clustering by GMM can be directly used to determine the parameters for the membership functions. Meanwhile, the rules can be extracted from the clustering results and the number of rules for the FIS is the number of clusters.

The consequent part is a linear function of the input components, and, therefore, we use a well-known method, least square estimation [18], to determine the parameters for the linear function.

3. Experiments

The proposed approach has been implemented in MATLAB and evaluated through a series of experiments on the traffic flow data which was collected and aggregated at intervals of 2 minutes' duration for a road in Beijing for one week, from November 20 (Monday) to November 26 (Sunday), 2006.

Before examining the proposed prediction approach, the performance of the δ-test should be evaluated as it has been employed to determine the decomposition level and input dimension. The evaluations on the δ-test from various aspects have been reported in [22–25], apart from its sensitiveness on the choice of the starting point. Therefore, this work is concentrated to a sensitive analysis to evaluate the dependence of the δ-test on the starting point. The one-week traffic flow data with 100 different starting points have been used by the δ-test to estimate the residual variance and the coefficients of variation have been calculated and listed in

TABLE 1: Coefficients of variation for the residual variances estimated for the traffic flow with 100 different initial points.

	Day						
	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Coefficient of variation	1.8983	2.1455	2.4925	3.2683	3.2022	1.4212	2.4154

Table 1 for the traffic flow of each day. From Table 1, it can be seen that all the coefficients of variation are bounded below 3.3, implying that the results obtained by the δ-test does not vary considerably with different starting points.

When determining the input dimension, the maximum number of input dimensions has to be specified by users. To this end, users may take advantage of their experience and knowledge on the traffic flow field. Here, we provide a simple method to determine the maximum number of input dimensions based on the calculation of the partial correlation coefficients [34] for the traffic flow data. Figure 2 shows the partial correlation coefficients obtained for the traffic flow data of each day with the maximum lag of 100 steps. This experiment implies the traffic flow with lag greater than 10 generally has much less significant association with the current flow, as most coefficients fall into 95% confidence intervals, as shown in Figure 2. Therefore, the maximum number of input dimensions has been selected to be 10 for the experiments presented in this paper.

The next experiment has been designed to evaluate the denoising and prediction performance of the proposed approach on the traffic flow data of one week. To evaluate the denoising performance for the proposed approach on the traffic flow, Kolmogorov-Smirnov test (K-S test) has been employed for each test to determine whether the denoised data and noise data estimated by the wavelet denoising technique follow the same distribution and the correlation coefficient between the denoised data and the estimated noise has been also calculated for each day to measure their dependency. On the other hand, Root Mean Square Error (RMSE) [13] was used as an index to evaluate the prediction performance of the proposed approach. Although it is not appropriate to use RMSE as an objective to build a model as the minimization of RMSE is likely to result in overfitting, it is suitable for performance evaluation particularly when comparison is carried out. For all the following experiments, the first 500 datasets of traffic flow was used to train a FIS and the last 200 datasets to test the prediction performance.

The obtained results are shown in Table 2 and Figures 3 and 4. The decomposition level and input dimension for the traffic flow on each day have been obtained when the variance ratio meets the required criterion. The number of rules (the same as the number of membership functions) has been determined by performing the NNC algorithm on the training data with the cluster radii set as 0.8.

Figure 3 shows the histograms for the denoised data and the noise data estimated by the wavelet denosing method with the decomposition levels determined by the proposed approach and listed in Table 2. It is obvious that the denoised data and the noise data do not follow the same distribution which is also proved by the p values obtained with 95% confidence level through the T-S test. From Figure 3, it can be

seen that the estimated noise data generally follow a Gaussian distribution. Additionally, the correlation coefficients calculated for the denoised flow and the noise data also indicate there is a very weak correlation between them. Furthermore, the original traffic flow data and the denoised flow are also illustrated in Figure 4, with grey solid lines and black dotted lines, respectively. From Figure 4, it can be seen that the denoised flow data is able to retain the general flow trend without losing the mostly significant peaks and valleys, while a smooth traffic flow pattern is obtained.

Regarding the training and prediction performance, RMSEs have been calculated and listed in Table 2, together with the fitting data and the predicted data depicted against the original data and the denoised data in Figure 4. From Table 2, although the RMSEs for training are lower than those for prediction, the prediction in general performed well. The larger RMSE values obtained during the prediction stage may be caused by inefficient samples presented during the training stage. Furthermore, the discrepancies between the fitting flow or the predicted flow and the denoised data are hardly seen from Figure 4 for all days, reflecting the proposed approach can accurately predict the flow trend.

To evaluate the proposed algorithm for input dimension determination, the input dimension has been varied from 1 to 10 while the decomposition level and the number of rules is kept the same as those listed in Table 2. When varying the input dimension, it is difficult to generate the same number of clusters (i.e., the number of rules) by the NNC algorithm and thus it was set to be ineffective for this experiment. Consequently, the initial centers for the GMM under construction were set randomly. To eliminate any bias that might result from the random initialization, 100 independent runs have been performed and the averaged results are presented in Table 3. While the RMSEs for training (denoted as T) and prediction (denoted as P) are presented, the ratios of the RMSEs for prediction over those for training are listed as well in an attempt to indicate the degree of overfitting.

In general, the RMSEs for each day of the week decrease with the growth of the input dimension, indicating a trend that the modeling performance is gradually improved. However, the situation for prediction is quite complicated, as the pattern is not very evident, though it appears that a concave pattern generally holds for the most of days. The RMSE ratios (indicated by P/T), in the cells with bold font in Table 3, correspond to the input dimensions shown in Table 2 and the smallest ratio for each day of the week is also highlighted by italic font. For the Monday, although the RMSE ratio corresponding to the input dimension of 2 is the third smallest RMSE ratio, the overfitting problem is not significant. However, the RMSE ratio for Tuesday is quite high, but the RMSE values obtained during the training
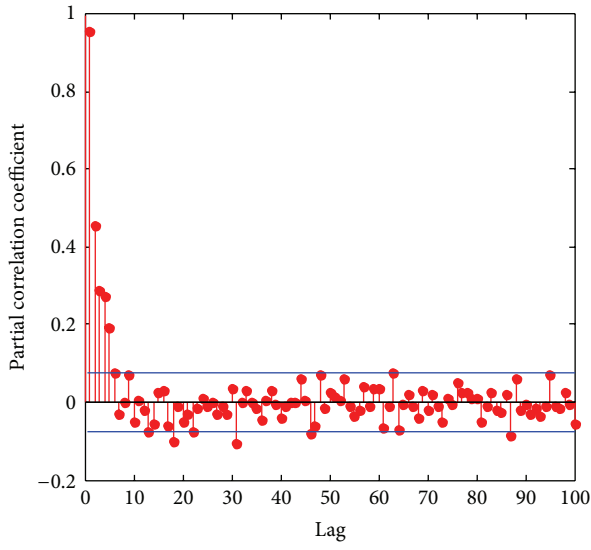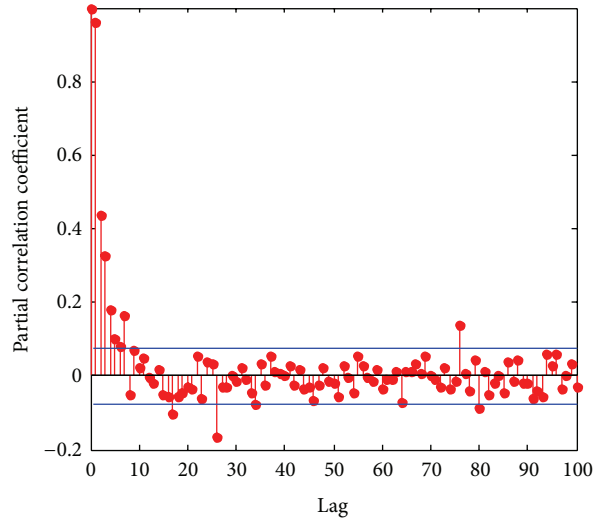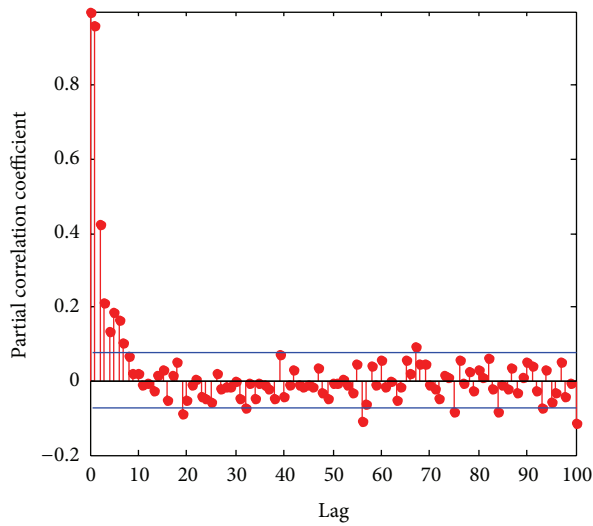
(a) November 20 (Monday)
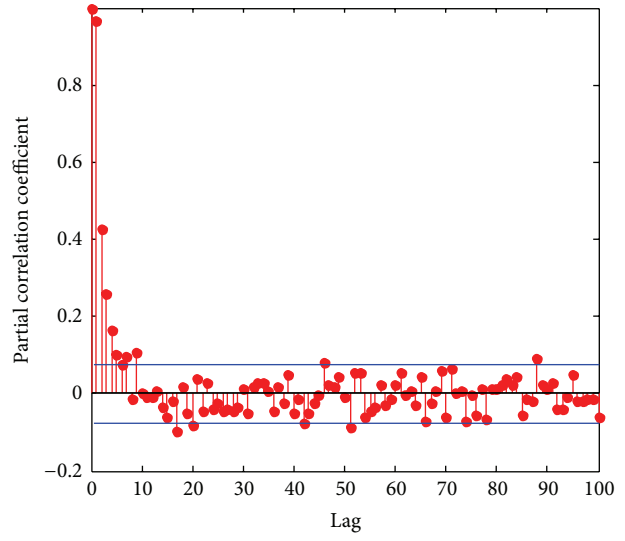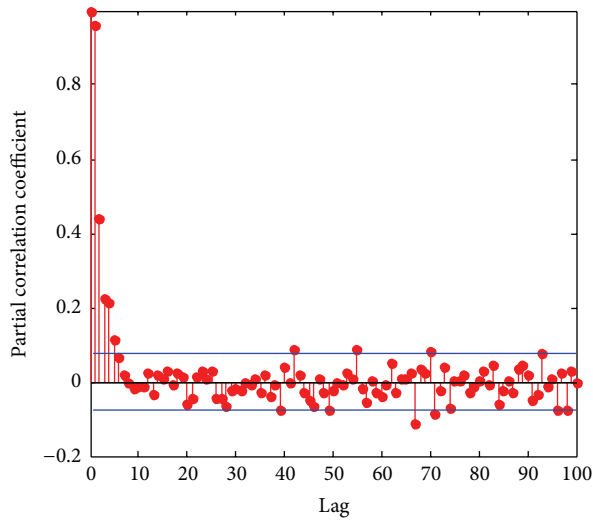
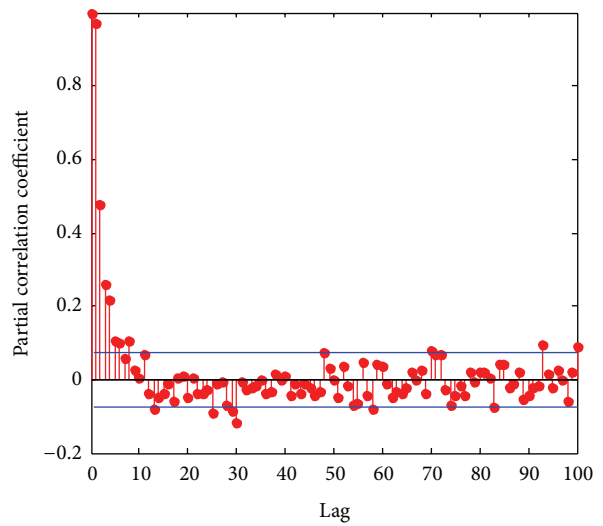(b) November 21 (Tuesday)

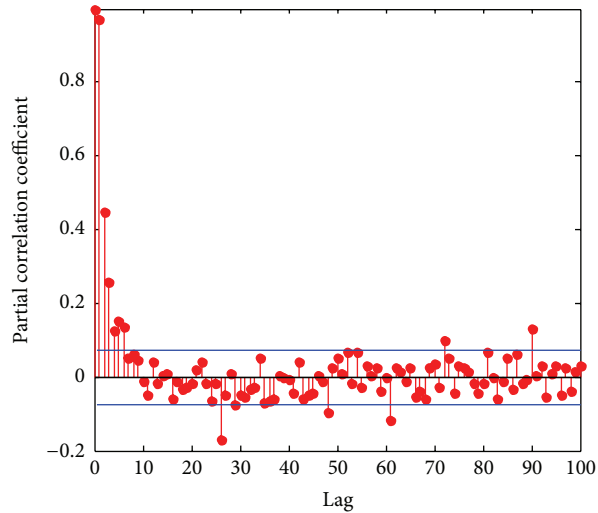(c) November 22 (Wednesday)

(d) November 23 (Thursday)

(e) November 24 (Friday)

(f) November 25 (Saturday)

Figure 2: Continued.

(g) November 26 (Sunday)

FIGURE 2: Partial correlation coefficients for the traffic flow data collected from November 20, 2006, to November 26, 2006, with 95% confidence interval indicated by the horizontal lines.

TABLE 2: Evaluation of the proposed approach on the one-week traffic flow.

	Day of week						
	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Decomposition level	3	4	4	4	4	3	3
p of K-S test (95%)	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Correlation coefficient	0.0042	0.0038	0.0039	0.0107	0.0041	0.0037	0.0007
Input dimension	2	4	4	2	2	2	2
Number of rules	2	2	2	2	2	2	2
Training (RMSE)	0.0107	0.0028	0.0054	0.0064	0.0054	0.0025	0.0052
Predicting (RMSE)	0.0181	0.0171	0.0124	0.0269	0.0095	0.0113	0.0058

and predicting phase are both smaller than those indicated with italic font, which gives the smallest RMSE ratio. A similar pattern is also held for Wednesday. For Thursday, the overfitting effect has been reduced to the minimum when the input dimension corresponds to that listed in Table 2. Although the RMSE ratios for Friday and Sunday are not the smallest, they are bounded within 2, implying overfittings are not considerable. For Saturday, the RMSE ratio is ranked as the second in terms of the smallest value, but the RMSE values for the training and prediction phase are all smaller than those with the smallest ratio (denoted as italic font). After all, the experimental results indicate that the overfitting problem can be generally restrained with the input dimension determined by the proposed approach which is not designed to explicitly deal with the overfitting problem.

In order to evaluate the NNC algorithm for the determination of the number of rules (clusters), we have performed another experiment by varying the number of clusters from 1 to 10 and the RMSE values and the RMSE ratios (i.e., P/T) obtained from the FISs created by the GMMs with random initialization are presented in Table 4. Note that the RMSEs and the RMSE ratios presented have been averaged over 100 independent runs as the initial center for each Gaussian component in a mixture model was initialized randomly.

The RMSEs listed in the column corresponding to 1 cluster are considerably large, as compared to those in other columns, and this is due to the fact that all training data have been grouped in one cluster. Apart from the first column, the performances are quite constant for the whole week, reflected by a general decreasing and increasing patterns possessed for modeling and prediction, respectively. In Table 4, the RMSE ratios in the second column have been highlighted with bold font to indicate they have been obtained when the number of rules (clusters) was determined as 2, as shown in Table 2. The RMSE ratios obtained with the choice of 2 rules for the days of the week except for Friday are all closer to 1 than those with the number of rules greater than 2. For Friday, the RMSE ratio (highlighted with the italic font in Table 2) mostly approaches 1 among those with the number of rules greater than 1, but the number of rules is just one more than that determined by the NNC algorithm. In summary, the choice of 2 rules (clusters) is proper as the overfitting problem can be considerably alleviated in general and consequently results in a compact structure for the FIS.
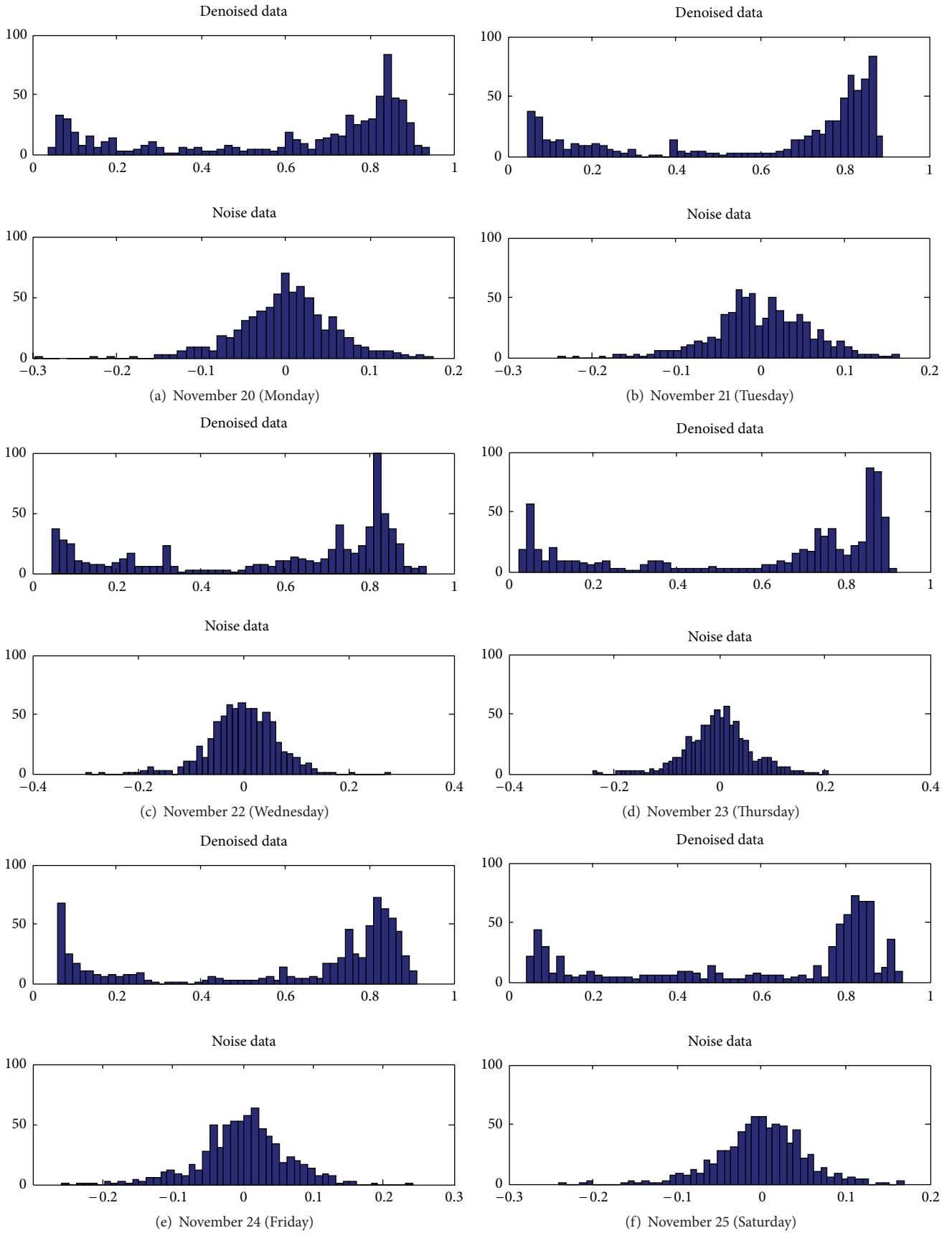
(a) November 20 (Monday)

(b) November 21 (Tuesday)

(c) November 22 (Wednesday)

(d) November 23 (Thursday)

(e) November 24 (Friday)

(f) November 25 (Saturday)

Figure 3: Continued.
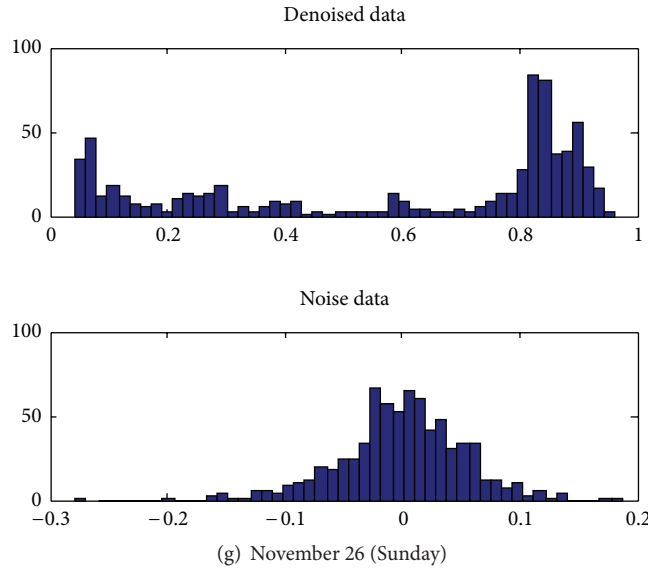
(g) November 26 (Sunday)

FIGURE 3: Histograms of the denoised data and the noise data in the week, November 20 to 26, 2006.

TABLE 3: Evaluation of the impact of the input dimension on the prediction by varying the dimension from 1 to 10 for the one-week traffic flow (RMSEs averaged over 100 independent runs are presented, T and P denote the RMSE values obtained from training and prediction, resp., and P/T presents the RMSEs for prediction over those for training).

	Inputs									
	1	2	3	4	5	6	7	8	9	10
Mon										
T	0.0115	**0.0108**	0.0105	*0.0108*	0.0095	0.0096	0.0092	0.0091	0.0093	0.0098
P	0.0191	**0.0154**	0.0147	*0.0128*	0.0207	0.0189	0.0186	0.0239	0.0267	0.0193
P/T	1.6597	**1.4265**	1.3996	*1.1863*	2.1734	1.9692	2.0206	2.6174	2.8844	1.9665
Tue										
T	*0.0059*	0.0029	0.0029	**0.0029**	0.0028	0.0028	0.0028	0.0028	0.0028	0.0027
P	*0.0187*	0.0181	0.0175	**0.0174**	0.0179	0.0183	0.0194	0.0180	0.0204	0.0202
P/T	*3.1700*	6.1422	6.0858	**6.1060**	6.3589	6.6108	6.9882	6.5280	7.3970	7.4519
Wed										
T	*0.0090*	0.0056	0.0053	**0.0053**	0.0053	0.0053	0.0052	0.0051	0.0050	0.0045
P	*0.0149*	0.0147	0.0130	**0.0130**	0.0129	0.0128	0.0127	0.0126	0.0121	0.0136
P/T	*1.6659*	2.6353	2.4358	**2.4358**	2.4476	2.4290	2.4454	2.4803	2.4069	3.0425
Thu										
T	0.0071	**0.0068**	0.0063	0.0056	0.0056	0.0056	0.0056	0.0056	0.0057	0.0056
P	0.0318	**0.0288**	0.0307	0.0289	0.0268	0.0269	0.0267	0.0292	0.0334	0.0469
P/T	4.4555	**4.2649**	4.8475	5.1203	4.8073	4.7819	4.7918	5.2005	5.8887	8.4189
Fri										
T	0.0062	**0.0059**	0.0058	0.0054	0.0053	0.0052	*0.0052*	0.0052	0.0050	0.0050
P	0.0171	**0.0110**	0.0115	0.0110	0.0100	0.0096	*0.0095*	0.0096	0.0100	0.0102
P/T	2.7758	**1.8723**	1.9914	2.0230	1.8928	1.8430	*1.8382*	1.8701	1.9925	2.0338
Sat										
T	*0.0055*	**0.0025**	0.0024	0.0024	0.0023	0.0023	0.0023	0.0022	0.0022	0.0019
P	*0.0135*	**0.0113**	0.0124	0.0128	0.0135	0.0146	0.0157	0.0149	0.0149	0.0162
P/T	*2.4502*	**4.5961**	5.1182	5.3653	5.7938	6.3996	6.9174	6.7054	6.7307	8.6916
Sun										
T	0.0093	**0.0052**	0.0052	0.0051	0.0049	0.0049	*0.0048*	0.0045	0.0045	0.0037
P	0.0124	**0.0055**	0.0052	0.0053	0.0052	0.0049	*0.0047*	0.0049	0.0048	0.0041
P/T	1.3314	**1.0533**	1.0068	1.0310	1.0636	0.9970	*0.9910*	1.0982	1.0878	1.1193

Table 4: Evaluation of the impact of the number of clusters on the prediction by varying it from 1 to 10 for the one-week traffic flow (RMSEs averaged over 100 independent runs are presented, and T and P denote the RMSE values obtained from training and prediction, resp.).

	Cluster									
	1	2	3	4	5	6	7	8	9	10
Mon										
T	260.31	**0.0108**	0.0103	0.0102	0.0101	0.0100	0.0099	0.0097	0.0097	0.0096
P	265.27	**0.0154**	0.0180	0.0194	0.0187	0.0196	0.0201	0.0210	0.0212	0.0224
P/T	1.0191	**1.4259**	1.7476	1.9020	1.8515	1.9600	2.0303	2.1649	2.1856	2.3333
Tue										
T	226.74	**0.0029**	0.0026	0.0026	0.0025	0.0025	0.0025	0.0024	0.0024	0.0024
P	253.02	**0.0174**	0.0224	0.0214	0.0211	0.0208	0.0204	0.0200	0.0202	0.0202
P/T	1.1159	**6.0000**	8.6154	8.2308	8.4400	8.3200	8.1600	8.3333	8.4167	8.4167
Wed										
T	241.15	**0.0053**	0.0049	0.0049	0.0048	0.0047	0.0046	0.0044	0.0044	0.0043
P	205.69	**0.0129**	0.0205	0.0212	0.0210	0.0218	0.0227	0.0236	0.0236	0.0257
P/T	0.8530	**2.4340**	4.1837	4.3265	4.3750	4.6383	4.9348	5.3636	5.3636	5.9767
Thu										
T	262.33	**0.0067**	0.0063	0.0050	0.0050	0.0046	0.0041	0.0035	0.0034	0.0032
P	280.59	**0.0287**	0.0308	0.0350	0.0378	0.0405	0.0426	0.0454	0.0436	0.0458
P/T	1.0696	**4.2836**	4.8889	7.0000	7.5600	8.8043	10.3902	12.9714	12.8235	14.3125
Fri										
T	258.21	**0.0059**	0.0063	0.0051	0.0050	0.0049	0.0048	0.0048	0.0047	0.0046
P	289.71	**0.0110**	0.0086	0.0133	0.0133	0.0129	0.0125	0.0134	0.0139	0.0143
P/T	1.1220	**1.8644**	*1.3651*	2.6078	2.6600	2.6327	2.6042	2.7917	2.9574	3.1087
Sat										
T	269.68	**0.0025**	0.0025	0.0024	0.0024	0.0024	0.0024	0.0024	0.0024	0.0024
P	328.19	**0.0113**	0.0115	0.0112	0.0114	0.0113	0.0111	0.0112	0.0112	0.0112
P/T	1.2170	**4.5200**	4.6000	4.6667	4.7500	4.7083	4.6250	4.6667	4.6667	4.6667
Sun										
T	274.15	**0.0052**	0.0053	0.0052	0.0052	0.0051	0.0051	0.0050	0.0049	0.0049
P	316.86	**0.0055**	0.0059	0.0065	0.0061	0.0061	0.0062	0.0061	0.0068	0.0072
P/T	1.1558	**1.0577**	1.1132	1.2500	1.1731	1.1961	1.2157	1.2200	1.3878	1.4694

4. Conclusions

The approach proposed for one-step-ahead prediction in this paper consists of a number of novel elements that have been developed to deal with the issues relating to the input dimension determination and parameters and rules estimation. A series of experiments have been performed based on the one-week traffic flow to evaluate the proposed prediction approach. The first experiment demonstrates a method introduced to determine the maximum number of input dimensions, a parameter required for the determination of the input dimension, by calculating partial correlation coefficient. By analyzing the distribution discrepancy between the denoised flow and the estimated noise data, the denoising performance has been evaluated and the experimental results indicate the denoising procedure is effective in the noise removal. The prediction performance has been evaluated by calculating the RMSE values for the training phase and the predicting phase. Further experiments have been performed to evaluate the input dimension and the number of rules determined by the

proposed approach in terms of overfitting. Although the prediction approach has not been designed with the particular emphasis on the overfitting issue, the experimental results have shown that no significant overfitting is caused from the proposed prediction approach. Based on the experimental analyses, it can be concluded that the proposed approach is compact and competent, but further improvements are possible, by refining the stopping criterion for denosing and input dimension determination process, for example.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

(a) November 20 (Monday)

(b) November 21 (Tuesday)

(c) November 22 (Wednesday)

(d) November 23 (Thursday)

(e) November 24 (Friday)

(f) November 25 (Saturday)

Figure 4: Continued.

Training phase

Prediction phase

(g) November 26 (Sunday)
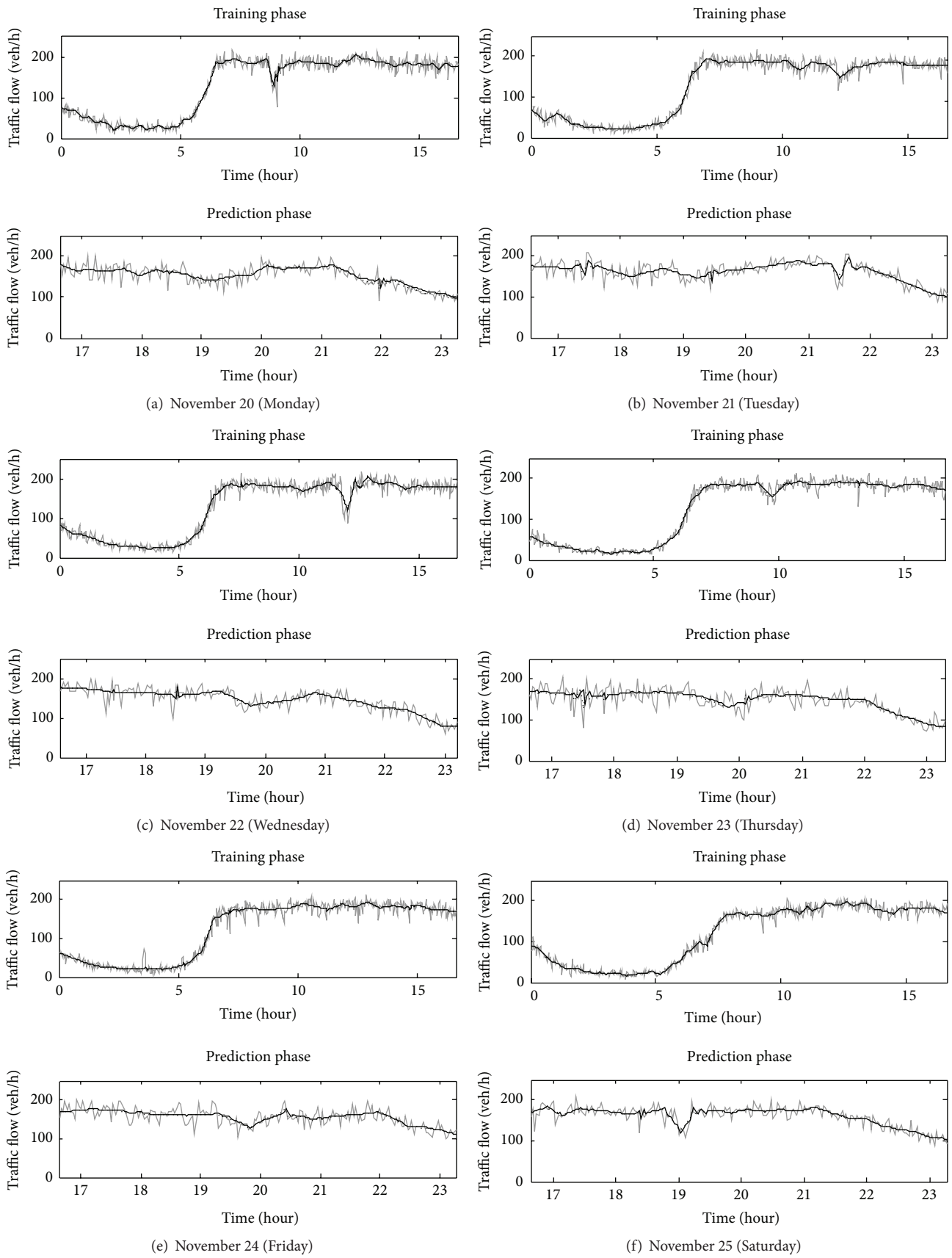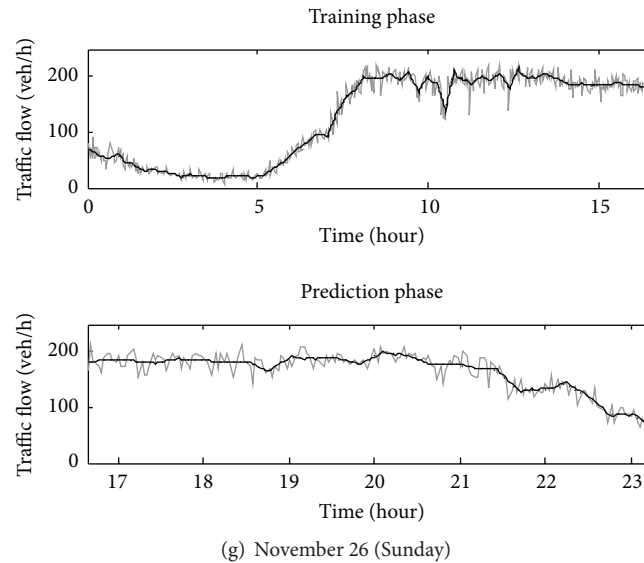
FIGURE 4: Evaluation of the proposed prediction approach for the training and prediction phases on the traffic flow data in the week, November 20 to 26, 2006 (the original flow data and the denoised flow data are represented by the grey solid lines and the black dotted lines, resp., while the black solid lines indicate the fitting and predicted traffic flow).

References

[1] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research Part B*, vol. 18, no. 1, pp. 1–11, 1984.

[2] D. Gazis and C. Liu, "Kalman filtering estimation of traffic counts for two network links in tandem," *Transportation Research Part B: Methodological*, vol. 37, no. 8, pp. 737–745, 2003.

[3] N. Zhang, Y. Zhang, and H. Lu, "Seasonal autoregressive integrated moving average and support vector machine models: prediction of short-term traffic flow on freeways," *Transportation Research Record*, vol. 2215, pp. 85–92, 2011.

[4] E. Castillo, J. M. Menéndez, and S. Sánchez-Cambronero, "Predicting traffic flow using Bayesian networks," *Transportation Research Part B: Methodological*, vol. 42, no. 5, pp. 482–509, 2008.

[5] K. Kuhn and A. Nicholson, "Traffic flow forecasting and spatial data aggregation," *Transportation Research Record*, no. 2260, pp. 16–23, 2011.

[6] Y. L. Zhang and Z. R. Ye, "Short-term traffic flow forecasting using fuzzy logic system methods," *Journal of Intelligent Transportation Systems*, vol. 12, no. 3, pp. 102–112, 2008.

[7] G.-J. Shen, "An intelligent hybrid forecasting model for short-term traffic flow," in *Proceedings of the 8th World Congress on Intelligent Control and Automation (WCICA '10)*, pp. 486–491, Jinan, China, July 2010.

[8] L. Dimitriou, T. Tsekeris, and A. Stathopoulos, "Adaptive hybrid fuzzy rule-based system approach for modeling and predicting urban traffic flow," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 5, pp. 554–573, 2008.

[9] C. M. Zhu, C. P. Yan, X. L. Xu, and G. X. Wu, "Research on the application of the prediction of the expressway traffic flow based on the neural network with genetic algorithm," *Advanced Materials Research*, vol. 189–193, pp. 4400–4404, 2011.

[10] B. L. Smith, B. M. Williams, and R. Keith Oswald, "Comparison of parametric and nonparametric models for traffic flow forecasting," *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 4, pp. 303–321, 2002.

[11] Y. S. Gu, Y. C. Li, J. C. Xu, and Y. P. Liu, "An aggregation approach to short-term traffic flow prediction, novel model based on wavelet transform and GA-Fuzzy neural network applied to short time traffic flow prediction," *Information Technology Journal*, vol. 10, pp. 2105–2111, 2011.

[12] Y. Wang, P. Beullens, H. H. Liu, D. J. Brown, T. Thornton, and R. Proud, "A practical intelligent navigation system based on travel speed prediction," in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC '08)*, pp. 470–475, IEEE, Beijing, China, October 2008.

[13] L. Zhao and F.-Y. Wang, "Short-term fuzzy traffic flow prediction using self-organizing TSK-type fuzzy neural network," in *Proceedings of the IEEE International Conference on Vehicular Electronics and Safety (ICVES '07)*, Beijing, China, December 2007.

[14] Y. C. Jin, "Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 2, pp. 212–221, 2000.

[15] F. M. Pouzols, A. Lendasse, and A. B. Barros, "Autoregressive time series prediction by means of fuzzy inference systems using nonparametric residual variance estimation," *Fuzzy Sets and Systems*, vol. 161, no. 4, pp. 471–497, 2010.

[16] L. X. Wang, *Adaptive Fuzzy Systems and Control—Design and Stability Analysis*, Prentice Hall, Englewood Cliffs, NJ, USA, 1994.

[17] Z. Ju and H. Liu, "Fuzzy Gaussian mixture models," *Pattern Recognition*, vol. 45, no. 3, pp. 1146–1158, 2012.

[18] G. Tsekouras, H. Sarimveis, E. Kavakli, and G. Bafas, "A hierarchical fuzzy-clustering approach to fuzzy modeling," *Fuzzy Sets and Systems*, vol. 150, no. 2, pp. 245–266, 2005.

[19] H. Ying, "General Takagi-Sugeno fuzzy systems with simplified linear rule consequent are universal controllers, models and filters," *Information Sciences*, vol. 108, no. 1–4, pp. 91–107, 1998.

[20] H. Ying, "Sufficient conditions on uniform approximation of multivariate functions by general Takagi-Sugeno fuzzy systems with linear rule consequent," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, vol. 28, no. 4, pp. 515–520, 1998.

[21] K. Zeng, N.-Y. Zhang, and W.-L. Xu, "A comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 6, pp. 773–780, 2000.

[22] H. Pi and C. Peterson, "Finding the embedding dimension and variable dependencies in time series," *Neural Computation*, vol. 6, pp. 509–520, 1994.

[23] A. J. Jones, "New tools in non-linear modeling and prediction," *Computational Management Science*, vol. 1, pp. 109–149, 2004.

[24] A. Lendasse, F. Corona, J. Hao, N. Reyhani, and M. Verleysen, "Determination of the mahalanobis matrix using nonparametric noise estimations," in *Proceedings of the European Symposium on Artificial Neural Networks*, Bruges, Belgium, April 2006.

[25] F. Mateo, D. Sovilj, and R. Gadea, "Approximate k-NN delta test minimization method using genetic algorithms: application to time series," *Neurocomputing*, vol. 73, no. 10–12, pp. 2017–2029, 2010.

[26] M. Misiti, Y. Misiti, G. Oppenheim, and J. M. Poggi, *Wavelets and Their Applications*, ISTE Publishing Knowledge, London, UK, 2007.

[27] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[28] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, "Wavelet shrinkage: asymptopia?" *Journal of the Royal Statistical Society, Series B: Methodological*, vol. 57, no. 2, pp. 301–369, 1995.

[29] Y. F. Ren and X. Z. Ke, "Selection of wavelet decomposition level in multi-scale sensor data fusion of MEMS gyroscope," *International Journal of Digital Content Technology and its Applications*, vol. 4, pp. 209–215, 2010.

[30] T. Cai and J. Zhu, "Wavelet de-noising of speech using singular spectrum analysis for decomposition level selection," *Journal of Shanghai Jiaotong University (Science)*, vol. 12, no. 2, pp. 190–196, 2007.

[31] Q. Zhang, W. Chen, and Y. Tang, "Method of choosing the adaptive level of discrete wavelet decomposition to eliminate zero component," *Optics Communications*, vol. 282, no. 5, pp. 778–785, 2009.

[32] S. R. Messer, J. Agzarian, and D. Abbott, "Optimal wavelet denoising for phonocardiograms," *Microelectronics Journal*, vol. 32, no. 12, pp. 931–941, 2001.

[33] C. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, Wiley-Interscience, New York, NY, USA, 2nd edition, 2008.

[34] W. Sun and M. Liu, "Wind speed forecasting using FEEMD echo state networks with RELM in Hebei, China," *Energy Conversion and Management*, vol. 114, pp. 197–208, 2016.