

# A Plane Moving Average Algorithm for Short-Term Traffic Flow Prediction

Lei Lv<sup>1</sup>, Meng Chen<sup>1</sup>, Yang Liu<sup>1</sup>, and Xiaohui Yu<sup>1,2</sup>(✉)

<sup>1</sup> School of Computer Science and Technology, Shandong University,  
Jinan 250101, China

llsdu13@gmail.com, chenmeng114@hotmail.com, {yliu,xyu}@sdu.edu.cn

<sup>2</sup> School of Information Technology, York University, Toronto,  
ON M3J 1P3, Canada

**Abstract.** In this paper, a plane moving average algorithm is proposed for solving the urban road flow forecasting problem. This new approach assembles information from relevant traffic time series and has the following advantages: (1) it integrates both individual and similar flow patterns in making prediction, (2) the training data set does not need to be large, (3) it has more generalization capabilities in predicting unpredictable and much complex urban traffic flow than previously used methods. To assess the new model, we have performed extensive experiments on a real data set, and the results give evidence of its superiority over existing methods.

**Keywords:** Plane moving average(PMA) · Time series · Flow pattern · Traffic flow prediction

## 1 Introduction

Traffic flow forecasting is a vital component of transportation planning, traffic control, intelligent transportation systems, and forecasting accurate traffic flow conditions has long been considered as an active approach to regional traffic control [16]. The approach can be broadly classified under: i) short-term and ii) long-term traffic flow forecasting [1]. Long-term forecasting provides monthly or yearly traffic flow forecasting conditions and is commonly used for long-term planning of transportation or construction. Short-term forecasting focuses on making predictions about traffic flow changes in the short-term, typically within one hour. In particular, short-term traffic volume forecasts support proactive dynamic traffic control. As a result, prediction technologies have gotten the attention of traffic engineers and researchers. A wide variety of techniques have been applied in the context of short-term traffic flow forecasting, depending upon the type of data that are available and the potential end use of the forecast. These techniques include moving average methods [17], k-nearest-neighbor methods [8], autoregressive MA (ARIMA) model or seasonal ARIMA (SARIMA) [19,20], neural networks (NNs) [9], and combining technologies such as DA approach [18].

Development of traffic flow forecasting models relies mainly on historical and current traffic flow data. The problem of traffic flow forecasting is a standard time series prediction task and the goal is to approximate the function that relates future values of traffic flow to previous and current observations of traffic flow, and the short-term traffic flow predictor represents a multi-input-single-output system, which relates the past traffic flow conditions to the future traffic flow conditions. Recent literatures have been concentrating on three aspects: i). presenting new or enhanced approaches [2, 11, 15], ii). merging efficient forecasting results [3, 7, 18], and iii). proposing effective pre-processing technologies [4, 5, 10].

However, it becomes clear that there are a few major problems with the existing methods. Firstly, those methods use data collected from motorways and freeways, where the change of traffic flow seems quite stable. Secondly, they use data collected from single point sources, which are limited in a more complicated situation. Thirdly, existing forecasting methods do not consider the temporal data as complex interactions in densely populated urban road networks and fresh comprehensive enough as well. Precisely, short-term traffic forecasting at urban arterials forms a more complex problem than freeway predictions due to constraints such as signalization, and urban flow collections contain lots of data points that previous models cannot handle their relations. The irrelevant traffic junctions may get the similar change rule and present the same flow pattern. The traditional methods do not consider the similar flow patterns and mine these similarities as well, they are often narrow in the angle of the data source and some of them are modeled so complicated, these reasons make them not appear to get practical application.

To solve those problems, we propose a Plane Moving Average (PMA) algorithm. The main idea of PMA is that it uses the closest flow data to make predictions through a new perspective. The PMA model is simple but effective and builds upon two models: the Individual Model and the Similarity Model. The first one, focuses on individual traffic junction using its own past vehicle volumes, the second one utilizes all traffic junctions' vehicle flow data based on the customized WWL (Where We Like) method which is a kind of Top-K technology and both of two sub-models use PMA algorithm to make predictions. Finally, We combine these two forecasting results by neural network to produce a more accurate predictor.

The contributions of this paper can be summarized as follows.

- We propose a Plane Moving Average to predict the short-term traffic flow. It is much simple but effective and uses a new angle of flow patterns in making prediction.
- To the best of our knowledge, PMA is the first model that takes similarity flow patterns into account and integrates both individual and similarity outcomes.
- We perform extensive experiments using a real data set and the results demonstrate the effectiveness of PMA.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 provides a brief description of preliminary preparation. Section 4

designs and describes the PMA Modeling of the short-term traffic flow predictor. In Section 5, the performance is discussed. Finally, the discussion of the conclusions and the future visions, regarding short-term traffic flow predictor design using the PMA with WWL algorithm, is presented in Section 6.

## 2 Related Work

For the past two decades, a considerable effort has made to develop efficient traffic prediction methods, which is backed by a huge number of literatures in this field. In what follows, we will describe three categories of studies that are most closely related to us.

*Long-term prediction:* Long-term prediction is researched in [13,14] and provides monthly or yearly flow forecasting conditions that is commonly used for long-term planning. Papagiannaki explores the properties of the network traffic, and propose a methodology that can be applied to forecast network traffic volume months in the future.

*Short-term prediction:* Short-term prediction has been widely investigated [2, 3,7,11,15,18–20], which is concerned with the prediction of only the nearly next period of time. Some of these methods make prediction with only the individual models [19,20], while others use the combination models [3,7,18]. Williams uses the well-known time series model ARIMA and SARIMA to make vehicular traffic flow prediction [19,20]. Tan et al. proposes an aggregation model (DA) that is elaborated using different fitting functions: the moving average (MA), exponential smoothing (ES), ARIMA and simple neural network for evaluation of prediction values [18]. Chan et al. provides a neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg? Cmarquardt algorithm [3]. Davarynejad describes a multi-phase time series architecture to solve the motorway flow forecasting problem [7]. [2,11,15] incorporate multiple factors more or less, such as travel speed, weather conditions and geographical features of a road. Given the travel speed and traffic volume of a road segment, more accurate predicting model can be built.

*Preprocessing techniques:* Several studies adopt data preprocessing technique to improve the final forecasting results [4,5,10]. Chan et al. proposes a simple but effective training method to pre-process traffic flow data before training purposes and the pre-processing approach intends to aid the back-propagation algorithm to develop more accurate neural networks [4,5]. Gao et al. decomposes the original data into burst data traffic and non-burst data and be predicted, respectively [10]. These techniques indicate that the flow predictor based on these pre-processed data outperform those that are developed based on original data.

## 3 Preliminary Preparation

In this section, we will define a few terms that are required for the subsequent discussion and describe our modeling strategy for the vehicle traffic flow forecasting.

**Definition 1 (Sampling Flow).** *The traffic flow data are collected all the time, each sampling flow refers to a total number of passed vehicles during a certain period of time. Unaffectedly, we choose  $\Delta t$  minutes per period that the traffic flow is collected within the time interval  $(t - \Delta t, t]$ ,  $t$  is an integer.  $q(t)$  is also an integer and stands for the traffic volume in the No.  $t$  periods.*

**Definition 2 (Flow Pattern).** *For each sampling flow,  $q(t)$  is the source time series. By analyzing the observed traffic flow data, it can be found that the traffic flow pattern is almost cyclical every day. Thus, two relevant time series are constructed as the period series  $s_a(t)$  and the daily series  $s_b(t)$ , denoted as follows: I.  $s_a(t)$  is a set that includes the previous  $k_a$  periods traffic flow data before  $q(t)$ .*

$$s_a(t) = \{q(t - k_a\Delta t), q(t - (k_a - 1)\Delta t), \dots, q(t - \Delta t), \hat{q}(t)\}.$$

II.  $s_b(t)$  is a set that includes the previous traffic flow record within the same time interval on  $k_b$  days before  $q(t)$ .

$$s_b(t) = \{q(t - 1440k_b), q(t - 1440(k_b - 1)), \dots, q(t - 1440), \hat{q}(t)\}.$$

### 4 PMA Modeling

We use a new PMA model to solve the short-term traffic flow prediction problem. PMA is an improvement algorithm of moving average(MA). Specifically, PMA runs on the plane flow pattern (a 2-dimensional data structure).

$S_b(t-k_a\Delta t)$		$\dots$	$S_b(t)$	
$q(t-1440k_b-k_a\Delta t)$	$q(t-1440k_b-(k_a-1)\Delta t)$	$\dots$	$q(t-1440k_b)$	$S_a(t-1440k_b)$
$q(t-1440(k_b-1)-k_a\Delta t)$	$q(t-1440(k_b-1)-(k_a-1)\Delta t)$	$\dots$	$q(t-1440(k_b-1))$	
$q(t-1440(k_b-2)-k_a\Delta t)$	$q(t-1440(k_b-2)-(k_a-1)\Delta t)$	$\dots$	$q(t-1440(k_b-2))$	
$\dots$	$\dots$	$\dots$	$\dots$	$\vdots$
$q(t-1440-k_a\Delta t)$	$q(t-1440-(k_a-1)\Delta t)$	$\dots$	$q(t-1440)$	
$q(t-k_a\Delta t)$	$q(t-(k_a-1)\Delta t)$	$\dots$	$\hat{q}(t)$	$S_a(t)$

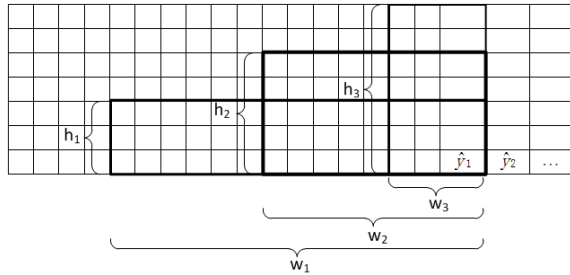
$PF_{individual}(t)$

**Fig. 1.** Plane flow pattern of individual model.  $\hat{q}(t)$  is the predicted value.

In order to take into account both individual and similar flow patterns to make the prediction, we propose two models, an Individual Model to model the individual flow patterns by using its own past traffic flow data and a Similarity Model to model the similar flow patterns by using alike historical traffic flow data. They are combined by using neural network to generate a predictor. What calls for special attention is that these two models share the PMA algorithm. Next we will present data construction methods about similarity model, individual model, the PMA training approach and the model integration.

### 4.1 Individual Model

We call  $s_a(t)$  the horizontal flow pattern and call  $s_b(t)$  vertical flow pattern, and propose a new perspective to build a plane flow pattern (PFP) (see Figure 1). There are  $s_a(t)$ ,  $s_b(t)$ ,  $k_a$  periods before  $s_b(t - k_a\Delta t)$  and  $k_b$  days before  $s_a(t - 1440k_b)$  combined together to build plane flow pattern (PFP) and each two adjacent value  $q$ 's timestamp differs either of  $\Delta t$  minutes and a day. We may find out it is possible that  $k_a\Delta t$  probably equals to 1440 which means that a typical sampling flow will appear more than once in PFP and get reutilization.



**Fig. 2.** Different size of PFP during the PMA training process and  $\hat{y}$  is the forecasting value

PMA algorithm is used to produce the forecasting results. For convenience, we will use  $y(i, j)$  to refer the  $i^{th}$  line of the  $j^{th}$  column's value of the plane flow pattern. In order to describe PMA algorithm, we must define two factors in the first place.

**Pattern Weight.** Each one of flow pattern could be benefited to the forecasting accuracy, so we define a kind of pattern weight  $\rho$  to describe different pattern's contribution to the target predictions.

$$\rho(m - i) = \frac{1}{L} \sum_{k_a=1}^L y(m, \delta) / y(m - i, \delta - k_a) \quad 0 < i < m \tag{1}$$

where  $L$  is the length of the PFP (see Figure 2) and  $\delta$  refers to  $\hat{q}(t)$ 's column,  $m$  is the target day number of the data set.

**PMA Forecasting Value.** The computing method is similar to MA, a kind of simple smoothing technique.

$$PMA(\delta) = \frac{1}{W} \sum_{k_b=1}^W y(m - k_b, \delta) \times \rho(m - k_b) \tag{2}$$

where  $W$  is the width of the PFP (see Figure 2). Each PMA forecasting value corresponds to a specific PMA size  $L_i \times W_i$ .

For example, consider this PFP 

3	4	5	10
2	3	4	8
1	2	3	[4]

 that consists of three kinds of  $s_a$

patterns and four kinds of  $s_b$  patterns, updown adjacent patterns are one day apart, the target value is 4. If we simply use flow pattern  $\{1, 2, 3\}$  and  $\{10, 8\}$  to make the average forecasting value, we receive value 2 and 9 but none of them is closed to 4. Using pattern weight we will get two weight 0.478 and 0.639, and the final forecasting value would change to  $(10 \times 0.478 + 8 \times 0.639)/2 = 4.946$ . Obviously, 4.946 is the nearest value.

---

**Algorithm 1.** PMA Algorithm

---

- 1: Firstly, for the PFP, work out the pattern weight  $\rho$  by (1);
  - 2:  $\langle X, Y \rangle$  is the size of the training set;
  - 3: **for**  $L = 1 \dots X$  **do**
  - 4:     **for**  $W = 1 \dots Y$  **do**
  - 5:         To each  $\langle L, W \rangle$ , compute PMA forecasting value of all training samples by (2);
  - 6:         Calculate the average RMSE between the each pair of forecasting value and the actual value;
  - 7:         Update the optimal PMA size according to the minimum RMSE;
  - 8:     **end for**
  - 9: **end for**
  - 10: Use the optimal PMA size to make predictions.
- 

In designing PMA training approach, the PMA size in the training set is an important feature that needs to be carefully chosen. The PMA algorithm is detailed in Algorithm 1. To obtain a PMA size that is capable of generalizing and performing well with new cases, data samples are usually subdivided into two sets: 1) a training set, 2) a testing set.

**4.2 Similarity Model**

In most cases, time series in the same category show the similar change rule, especially in a certain field (e.g., same area). For example, a flow that over 1000 vehicles /h must not happen in a small town’s main road. Therefore, we put forward a hybrid distance calculation criterion which combined Euclidean Distance and Pearson Correlation Coefficient to evaluate time series similarity, and select top-N similar time series to make predictions. Notice that the comparable time series should be at least **one** day before and if not, no known value could be available.

**WWL (Where We Like) Algorithm.** Between each two time series(equilong), we come up with a hybrid distance calculation which combined Euclidean Distance and Pearson Correlation Coefficient as the distance metric to define their

similarities. Minimizing these two metrics can both ensure a highly similar level about traffic volume and landmark as well. Firstly, filter time series by calculating Pearson Correlation Coefficient  $pcc(s^i, s^j) = \frac{\sum_{t=1}^m (s^i(t) - \bar{s}^i)(s^j(t) - \bar{s}^j)}{\sqrt{\sum_{t=1}^m (s^i(t) - \bar{s}^i)^2 \sum_{t=1}^m (s^j(t) - \bar{s}^j)^2}}$  that pcc value is over 0.8 to obtain the highly closed shape. Secondly, calculate Euclidean Distance by equation  $ed(s^i, s^j) = \|s^i - s^j\|$  to make sure that the traffic volume is as much as close and order them in the following way.

$$PFP_{similarity}(t) = \{s(t), s^1(t), \dots, s^i(t), \dots, s^d(t)\}. \tag{3}$$

where  $s(t)$  is the target flow pattern,  $s^i(t)$  stands for the  $No.i$  similar flow pattern to  $s(t)$  and the number  $d$  is the parameter of WWL algorithm. Remember that each  $s^i(t)$  should be at least **one** day before.

Empirically, in terms of flow data, the more similar the shape, the more useful the flow pattern. Hence, for ease of running PMA, we put them bottom-top in a straight line according to the order of similarity by WWL algorithm just the same as  $PFP_{individual}(t)$  does in individual model and PMA algorithm is used to produce its forecasting results.

### 4.3 Integration

Choosing a proper model to merge these two forecasting results is the primary task. There are many popular ways can be applied to do combination, such as Linear Regression, Logistic Regression, Neural Network, etc. For the effectiveness and simplicity, we choose neural network to integrate these two models that we have proposed.

As a result, the similarity model and individual model can get a series of predicting values,  $\mathbf{P}^w = (p_1^w, p_2^w, \dots, p_m^w)$  ( $w = 1$  for similarity model and  $w = 2$  for individual model), where  $m$  is the number of the time-ordered sampling results, and  $p_i^w$  is the forecasting value of time  $m$ . Certainly, we have a series of real traffic flow,  $\mathbf{R} = (r_1, r_2, \dots, r_m)$  as the predicting target. Then we can build a data set like  $(\mathbf{P}^1, \mathbf{P}^2, \mathbf{R})$ .

In designing NN models, the activation functions in the hidden layer and the output layer are *tansig* and *purelin*, respectively. We use the Levenberg-Marquardt BP algorithm as the learning algorithm because LM algorithm has the quickest convergence and it is the best learning rule in this case.

$$\begin{aligned}
 h_i &= \text{tansig}\left(\sum_{j=1}^2 w_{ij}p^j - \theta_1\right) \\
 net &= \text{purelin}\left(\sum_{i=1}^n w_i h_i - \theta_2\right)
 \end{aligned}
 \tag{4}$$

where  $n$  is the number of neurons in hidden layer and  $h_i$  is the hidden layer output and  $net$  is the final output. Although there is no precise rule on the optimum size of the training set and testing set, it is recommended that the training set should be larger.

## 5 Performance Evaluation

We have made extensive experiments to evaluate the performance of the proposed PMA model by using a real traffic flow data set. In this section, we will first explain the dataset and experimental settings, followed by the evaluation metrics to measure the performance and then, show the experimental results.

### 5.1 Study Area

The dataset used in the experiments consists of real vehicle passage records from April 1, 2013 to May 5, 2013 that collected from the traffic surveillance system of a major metropolitan area [6]. The dataset contains 140,440,933 real vehicle passage records, involves totally 308 camera locations on the main roads and we divide it into two parts, training set and testing set. The training set for the above models covers the data from April 1, 2013 to May 2, 2013 and the testing set is the traffic records from May 3 to May 5, 2013. We average the experimental results of all camera locations. The traffic flow data were aggregated and averaged into  $\Delta t$  minutes per period.

### 5.2 Pre-processing

We pre-process the dataset to form the flow patterns, counting the number of vehicles in each period of time, [18] suggests setting time interval  $\Delta t$  as 60. If a sampling flow is missing (always be 0 in this case) due to the camera's break or any other reason, we fix it by averaging the adjacent values. After the pre-processing, we get a total of 10780 horizontal flow patterns and 7392 vertical flow patterns and we predict traffic flow of future one period in the experiments.

### 5.3 Goodness-of-Fit Statistics

We use two widely employed evaluation statistics to assess the forecast accuracy of the results.

1) The **Root Mean Squared Error (RMSE)** is a way to measure the average error about the forecasting results and is calculated as

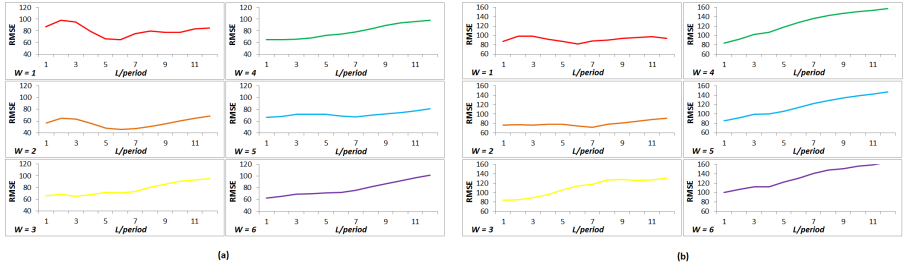
$$RMSE = \sqrt{\frac{1}{N} \sum_{n=1}^N (y_n - \hat{y}_n)^2} \quad (5)$$

2) The **Mean Absolute Percentage Error (MPAE)** is a way to measure the proportional error about forecasting results and is calculated as

$$MPAE = \frac{1}{N} \sum_{n=1}^N \frac{|y_n - \hat{y}_n|}{y_n} \times 100\% \quad (6)$$

Here,  $y_n$  and  $\hat{y}_n$  are the observed and the forecast values of observation  $n$ , respectively, and  $N$  is the total number of observations.



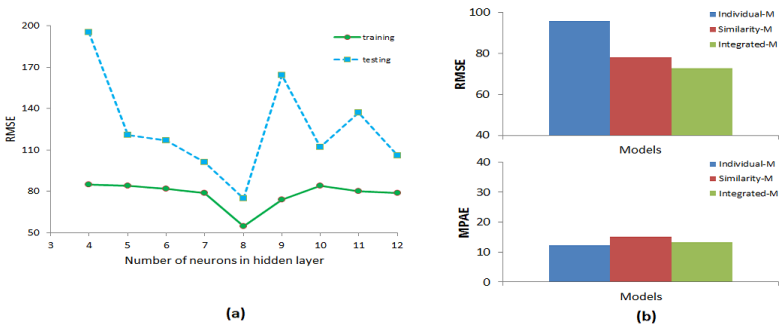


**Fig. 3.** The training RMSEs of the individual models and similarity models with different values of  $\langle L, W \rangle$ , respectively

### 5.4 Evaluation of PMA

We evaluate performance of PMA algorithm in these two sub-models, individual model and similarity model, and their integrations. For each experiment, we make predictions for all camera locations and report the average of the evaluation statistics.

Firstly, We vary the parameters  $\langle L, W \rangle$  (mentioned before, is the size of plane series) in (1) and (2) of individual model and similarity model. By reason of the RMSE could reflect the absolute error in traffic volume, we choose it to help training our models. Figure 3(a) and (b) show the RMSEs of the individual models and similarity models with different values of  $\langle L, W \rangle$ , in which the RMSE is calculated based on 24 observations (60 mins, one day). Based on the results showed in the picture, we choose  $L = 6, W = 2$  for individual model and  $L = 6, W = 2$  for similarity model as the optimal PMA size, when the RMSE is the smallest, respectively.



**Fig. 4.** Hidden-node numbers versus the RMSE on the training set and testing sets, and performance comparison among sub-models and their integration

Secondly, for the NN integration model, a series of NNs with different numbers of neurons in the hidden layer are trained. The number of neurons varies from 4 to 12, and the RMSEs are calculated for both the training set and the

testing set. According to its generalization ability on the testing set, the lower the value of the RMSE is, the better the network model is. Figure 4(a) shows the curve of the RMSE versus the number of hidden-layer neurons. In Figure 4(a), we find that the best number of hidden-layer neurons is 8. Therefore, a 2-8-1 NN model is selected for the further predictions. Figure 4(b) shows performance comparison among two sub-models and their integration. We can see that the RMSEs are significantly reduced and MPAEs are modestly improved.

### 5.5 Comparison of Results

Several single-source models including the naïve, MA, ARIMA, and NN model, and an aggregation model are applied to time series  $s_a(t)$  and  $s_b(t)$ . We compare their predictions on the testing sets with the PMA model.

1) **Naïve (or no-change) model** for traffic flow forecasting has the simplest form

$$\hat{q}_{Na}(t) = q(t - 1)$$

where  $\hat{q}_{Na}(t)$  is the forecast value at time interval  $t$ .

2) **MA Model**: An MA of order  $k$  is computed by

$$\hat{y}_{t+1} = \frac{y_t + y_{t-1} + t_{t-2} + \dots + t_{t-k+1}}{k}$$

where  $k$  is the number of terms in the MA [12]. The MA technique deals only with the latest  $k$  periods of known data; the number of data points in each average dose not change as time continues.

3) **ARIMA Model**: A general ARIMA model of order  $(r, d, s)$ , where  $d$  is the order of differencing, and orders  $r$  and  $s$  are the AR and MA operators.

4) **Artificial NN**: It is a single-source model that used for comparison with the PMA. The NN model is used to fit the nonlinear relationship

$$\hat{q}_{NN}(t) = f_1(q(t - 1), q(t - 2), \dots, q(t - l))$$

The inputs of NN model are the traffic flow records at previous  $l$  successive time intervals, and its output is the prediction of traffic flow at time interval  $t$ . The number of inputs  $l$  and the number of hidden neurons in the NN model are also optimized by experimentation.

5) **Aggregation Model (DA)**: It integrates three submodels (MA, ES and ARIMA) using simple neural network to combine the forecasting results.

Note that different models (naïve, MA, ARIMA, NN, DA) may need a different length of historical data. Hence, the sample size should be properly chosen for each model. For example, the length of time of the training data for the naïve model is no more than one hour before the forecasting time, while that for the training data for the ARIMA model at least covers two days before. For each model, we choose the parameters by observing the best fitting or forecasting and compare their forecasts on the same test sets.

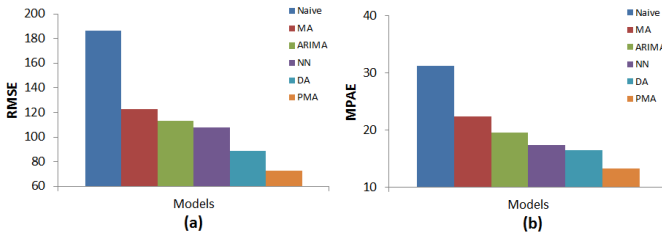


Fig. 5. RMSE and MPAE of different models in comparison

We generate forecasts for the future one period. Figure 5 show the RMSEs and MAPEs of testing set with the different forecasting models. For the Naïve, we simply use the last value as the forecasting value. As to the MA model, we set  $k = 3$  which is the most optimal parameter. For the ARIMA model, we set parameter as  $(1, 1, 0)$  where RMSE is the smallest. We set  $\alpha = 0.1$  and  $\gamma = 0.1$  in the ES model to build DA approach. For the last one, we select a 3-12-1 NN model for comparison. Figure 5 (a) and (b) show the predictions that resulted from the PMA model are better than the predictions that resulted from the other five models.

## 6 Conclusion

In this paper, aiming at the repeatable pattern of the similar traffic flow time series that previously methods did not consider, PFP has been constructed, a PMA algorithm and an aggregation strategy have been proposed to build a short-term traffic flow predictor. The PFP can be easily constructed from the source time series and the forecasting value of the PMA model can be automatically obtained by a computer program once the parameters are specified. Therefore, this new proposed approach is not a time-consuming but, rather a feasible job.

The PFP make full use of the information in the source time series which is collected on individual model as well as similarity model. By analyzing the forecasting performance of the naïve, MA, ARIMA, NN, DA and PMA, we have shown that the PMA can provide more accurate results than those of the other models.

For the further work, with the development of big data, data mining and machine learning, not limited to digital correlation, it is possible that we can consider the geographic correlation and even each flow movements to describe more potential characteristics and relationships of traffic flow. This problem deserves further study.

**Acknowledgments.** This work was supported in part by the 973 Program (2015CB-352 500), the National Natural Science Foundation of China Grant (6127 2092), the Shandong Provincial Natural Science Foundation Grant (ZR2012 FZ004), the Science and Technology Development Program of Shandong Province (2014GGE27178), the Independent Innovation Foundation of Shandong University (2012ZD012), the Taishan Scholars Program, and NSERC Discovery Grants.

## References

1. Abdulhai, B., Porwal, H., Recker, W.: Short-term traffic flow prediction using neuro-genetic algorithms. *ITS Journal-Intelligent Transportation Systems Journal* **7**(1), 3–41 (2002)
2. Chan, K.Y., Dillon, T., Chang, E., Singh, J.: Prediction of short-term traffic variables using intelligent swarm-based neural networks. *IEEE Transactions on Control Systems Technology*, **21**(1), 263–274 (2013)
3. Chan, K.Y., Dillon, T.S., Singh, J., Chang, E.: Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg-marquardt algorithm. *IEEE Transactions on Intelligent Transportation Systems* **13**(2), 644–654 (2012)
4. Chan, K.Y., Dillon, T.S., Singh, J., Chang, E.: Traffic flow forecasting neural networks based on exponential smoothing method. In: 2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 376–381. IEEE (2011)
5. Chan, K.Y., Yiu, C.K.: Development of neural network based traffic flow predictors using pre-processed data. In: *Optimization and Control Methods in Industrial Engineering and Construction*, pp. 125–138. Springer (2014)
6. Chen, M., Liu, Y., Yu, X.: NLPMM: A Next Location Predictor with Markov Modeling. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) *PAKDD 2014, Part II. LNCS*, vol. 8444, pp. 186–197. Springer, Heidelberg (2014)
7. Davarynejad, M., Wang, Y., Vrancken, J., van den Berg, J.: Multi-phase time series models for motorway flow forecasting. In: 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 2033–2038. IEEE (2011)
8. Davis, G.A., Nihan, N.L.: Nonparametric regression and short-term freeway traffic forecasting. *Journal of Transportation Engineering* **117**(2), 178–188 (1991)
9. Dia, H.: An object-oriented neural network approach to short-term traffic forecasting. *European Journal of Operational Research* **131**(2), 253–261 (2001)
10. Gao, Q., Li, G.: A traffic prediction method based on ann and adaptive template matching (2011)
11. Guo, H., Xiao, X., Tang, Y.: Short-Term Traffic Flow Forecasting Based on Grey Delay Model. In: Lei, J., Wang, F.L., Deng, H., Miao, D. (eds.) *AICI 2012. LNCS*, vol. 7530, pp. 357–364. Springer, Heidelberg (2012)
12. Hanke, J.E., Reitsch, A.G., Wichern, D.W.: *Business forecasting*. Prentice Hall Upper Saddle River, NJ (2001)
13. Papagiannaki, K., Taft, N., Zhang, Z.-L., Diot, C.: Long-term forecasting of internet backbone traffic: Observations and initial models. In: *INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2, pp. 1178–1188. IEEE (2003)
14. Papagiannaki, K., Taft, N., Zhang, Z.-L., Diot, C.: Long-term forecasting of internet backbone traffic. *IEEE Transactions on Neural Networks* **16**(5), 1110–1124 (2005)
15. Shang, J., Zheng, Y., Tong, W., Chang, E., Yu, Y.: Inferring gas consumption and pollution emission of vehicles throughout a city. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1027–1036. ACM (2014)
16. Smith, B.L., Demetsky, M.J.: Traffic flow forecasting: comparison of modeling approaches. *Journal of Transportation Engineering* **123**(4), 261–266 (1997)

17. Smith, B.L., Williams, B.M., Oswald, R.K.: Comparison of parametric and non-parametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies* **10**(4), 303–321 (2002)
18. Tan, M.-C., Wong, S.C., Xu, J.-M., Guan, Z.-R., Zhang, P.: An aggregation approach to short-term traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems* **10**(1), 60–69 (2009)
19. Williams, B.M.: Multivariate vehicular traffic flow prediction: Evaluation of arima modeling. *Transportation Research Record: Journal of the Transportation Research Board* **1776**(1), 194–200 (2001)
20. Williams, B.M., Hoel, L.A.: Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of Transportation Engineering* **129**(6), 664–672 (2003)