LI LINCHAO, Ph.D. Candidate[1, 2]
E-mail: linlinchao123@163.com
TOMISLAV FRATROVIĆ, Ph.D.[3]
E-mail: tomislav.fratrovic@fpz.hr
ZHANG JIAN, Ph.D.[1, 2]
(Corresponding Author)
E-mail: jianzhang@seu.edu.cn
RAN BIN, Ph.D. [1, 2]
[1] School of Transportation, Southeast University
  Si Pai Lou #2, Nanjing 210096, China
[2] Research Center for Internet of Mobility, Southeast
  University, Si Pai Lou #2, Nanjing 210096, China
[3] University of Zagreb, Faculty of Transport and Traffic
  Sciences, Vukelićeva 4, 10000 Zagreb, Croatia

TRAFFIC SPEED PREDICTION FOR HIGHWAY OPERATIONS BASED ON A SYMBOLIC REGRESSION ALGORITHM

ABSTRACT

Due to the increase of congestion on highways, providing real-time information about the traffic state has become a crucial issue. Hence, it is the aim of this research to build an accurate traffic speed prediction model using symbolic regression to generate significant information for travellers. It is built based on genetic programming using Pareto front technique. With real world data from microwave sensor, the performance of the proposed model is compared with two other widely used models. The results indicate that the symbolic regression is the most accurate among these models. Especially, after an incident occurs, the performance of the proposed model is still the best which means it is robust and suitable to predict traffic state of highway under different conditions.

KEY WORDS

highway congestion; traffic state; sensor data; speed prediction; incident; symbolic regression; genetic programming;

1. INTRODUCTION

According to the Urban Mobility Scorecard Annual Report in 2015 by INRIX, traffic congestion continues to be one of the biggest challenges in many major cities worldwide. For example, congestion caused each commuter in Washington DC to travel 82 hours more and burn 35 gallons of fuel extra which amounted to an average value of $1,834 per traveller. In order to reduce the congestion, in the last decades, a large number of roads were built. However, this approach did not give the expected results and did not entirely solve the problem. A better solution would be to fully use the existing roads and provide real-time traffic information to the travellers. With the useful information, travellers could choose when to start their trips and which routes to take the least times. For traffic information to have

beneficial effects, short-term traffic speed prediction is important, because in the Advanced Traveler Information System (ATIS) the traffic state is represented by the speed. In fact, current traveller information activities are being hampered by the lack of capacity to predict future traffic state [1].

Formally, given a sequence of values $\{X_1, X_2, ..., X_t\}$ of some observed traffic parameters (such as speed or occupancy) from sensors in the transportation network, the prediction problem is defined to predict the future values X_{t+i} (i=1,2,...n) and the objective is to build a prediction model that can minimize the gaps between prediction values and actual values. Throughout the years, many researchers have developed a wide variety of traffic prediction models from different perspectives. In general, the models can be grouped into statistics-based and computational intelligence-based models [2].

In statistics-based models, the linear regression basically assumes a linear combination of covariates and has a simple structure that can be found in [3-5]. In these studies, some techniques were implemented to enhance the accuracy such as stepwise method and varying coefficients. On the other hand, time-series approaches were widely used based on the works of Box and Jenkins who popularized a specific class of the method. They comprise combinations and generalizations of autoregressive moving average (ARMA) models [6]. Starting from this work, different types of models originated including autoregressive integrated moving average (ARIMA) [1, 7], seasonal ARIMA (SARIMA) [8, 9], vector ARMA (VARMA) [10, 11], space-time ARIMA (STARIMA) [12], etc. Kalman filter is another statistical model that has been used to predict traffic parameters by various researchers [13-15]. This approach uses time series methods predicting the future

traffic parameter by continuously updating the selected state variables. With the transition equation, the selected state vector is developed based on the past and current observations along with an optimal estimation state vector [16]. From the previous studies, it can be summarized that the statistics-based models have solid and widely accepted mathematical foundations that give them ability to provide more insight in the relationship among variables. However, its disadvantages are also obvious. The main drawback of this approach are the strong assumptions which are not practically relevant to the realistic traffic flow [17].

In recent years, the applications of computational intelligence-based models were the focal point in literature including Neural and Bayesian Networks [18, 19], Support Vector Regression (SVR) [20], *k*-nearest neighbours [21, 22], regression trees [23, 24], etc. This type of models was considered as inevitable, particularly as most classical approaches have been shown inadequate under unstable traffic conditions and complex road settings when and where it is needed [25]. Research has proven that computational intelligence-based models are statistically superior to linear statistical models because they are more flexible, more robust and more appropriate for capturing the complexity and uncertainty of traffic flow than traditional statistics-based models [17]. However, despite the numerous research papers in traffic prediction that utilize computational intelligence-based models, researchers often used them blindly, ignoring some of their shortcomings. First of all, this type of models are "black-box" because they cannot provide specific algorithms and the relationship between variables is difficult to interpret. In addition, the over-fitting problem is quite common when applying computational intelligence-based models with complex structure to training set with relatively small sample size. Getting a nearly perfect performance for the training set can lead to a large performance drop for the testing set [26].

Symbolic regression (SR) uses a function discovery approach for modelling relationships in the dataset. Unlike traditional regression methods in statistics that fit parameters to an equation with a predefined form, symbolic regression tries to develop the algorithm by simultaneously searching both the parameters and the forms. As a result, the output form of a symbolic regression is a formula that is simple and interpretable. Moreover, it is not necessary to give a prior assumption between the explanatory variables and the response variable as well as on the model structure. Additionally, it is not limited by the sample size and the important variables are automatically selected into the model. In recent years, a variety of evolution algorithms have been used to identify symbolic regression including genetic programming (GP), evolution strat-

egies, colony optimization, differential evolution, and many others [27]. Among these techniques, genetic programming developed by Koza is the most popular because it is easy to understand and uses probabilistic selection rules, not the deterministic ones [28].

There are various successful applications of symbolic regression in prediction. For example, it was applied to predict wind energy production and got a reliable prediction for the renewable energy [29]. It was also used to predict the oil production and validated it on both synthetic and real data, and made reliable results [30]. A novel symbolic regression was introduced to predict elastic modulus of self-compacting concrete [27]. In [31] it was used to predict syngas yield production. After comparison the results showed that multi-gene genetic programming is better than single-gene genetic programming. In the area of transportation, GP was used to build models for real-time crash prediction on highways [32]. It was used to input the missing value in the intelligent transportation system [33]. Also, GP was applied to evaluate the performance of the pavement, where it was used to develop models to predict pavement rutting [34].

To reduce the congestion caused by accidents, one can look for the influencing factors [35] and the severity of the damages. In this paper a different approach is taken. A task is set to build some models that can effectively utilize the huge amount of data and accurately predict the traffic state. We propose a novel symbolic regression method for traffic speed prediction which is built based on a popular evolution algorithm, genetic programming [28]. The contributions of this study are: (1) the proposed model does not only make up for the disadvantages of both statistics-based models and computational intelligence-based models, but also makes full use of their advantages; (2) the proposed model is evaluated with the data from the real world and the sensitivity of the input is analysed.

2. METHODOLOGY

2.1 General description of SR

Let $\mathbf{y}$ be the vector of true values and $\hat{\mathbf{y}}$ the vector of prediction values for the traffic data. The structure of the symbolic regression can be presented as in *Figure 1*: where for each $i \in \{1,...,G\}$: b_0 and b_i are the parameters that can be estimated by the least squares method; $\mathbf{t}_i$ is the ($N{\cdot}1$) vector of outputs from the i-th GP tree; N is the sample size.

The symbolic regression shown in *Figure 1* can be applied through the following four simple steps: (1) generate the initial random population; (2) evaluate the fitness of individuals; (3) conduct genetic operators; (4) select the best model among outputs. The detailed explanations of each procedure are given in the following:
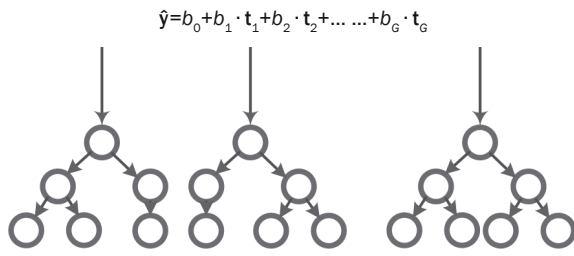
$$\hat{y} = b_0 + b_1 \cdot t_1 + b_2 \cdot t_2 + \ldots \ldots + b_G \cdot t_G$$

Figure 1 – The structure of the symbolic regression

1) Generate the initial random population

A population of random individuals is generated using the input variables as well as a specified function set such as sin, tan, cos, log, or exp, and so on. The parameters are randomly generated from the predefined bounded interval [-10, 10] [36]. In this stage, node functions, size of population, maximum depth of tree and maximum number of genes need to be set to control the generation.

2) Evaluate the fitness of individuals

Then each generation individual is evaluated by calculating the correlation coefficient R of observed values and predicted values and the bigger R corresponds to the better individual.

3) Conduct genetic operators

Once the fitness of individual is obtained, the genetic operations are conducted. The operators including crossover and mutation are shown in *Figure 2*. Next, the new individuals are sent to step 2. In this stage, the total number of the cycles needs to be set.

4) Select the best model among outputs

Finally, it returns a large number of models which differently describe the specifics of the datasets. In the last stage, it is a difficult task to select the best model while balancing a trade-off between performance and complexity. In this paper, we convert this task into a multi-objective optimization problem and solve it using Pareto front. The two objectives of the optimization problem are to minimize the complexity $f_1(\mathbf{x})$ and the fitness $f_2(\mathbf{x})$ of the expression $\mathbf{x} \in \Omega$, where Ω is a feasible region, i.e. a set of all feasible solutions.

2.2 Complexity

The complexity of an expression is decided by its structure such as tree depth, tree nodes, component function nonlinearity and number of variables. F. Smits proposed using the sum of the complexity of the tree structure and all sub trees as a metric that can reflect the characteristic of the model [37]. The criterion is defined as the total number of nodes in the structure and each sub tree. The less complex the expression is, of course, the better. *Figure 3* shows the computing processes of two different simple structures with three nodes.
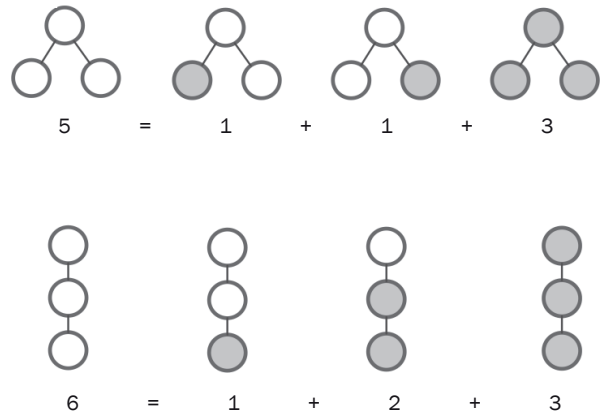

Figure 3 – Example of the computing processes

The complexity of expression $\mathbf{x} \in \Omega$ is computed by

$$f_1(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x}) \tag{1}$$

where $p_j(\mathbf{x})$ is the total number of nodes in the *j*-th sub tree of expression $\mathbf{x}$, *j*=1,...,*m*. In *Figure 3*, the second structure is more complex with $f_1(\mathbf{x})$=6.

2.3 Fitness

The fitness evaluates how good a model is. Basically, two types of measures, root mean squared error (RMSE) and the R^2, are usually used during building the model. *R* is the correlation coefficient of observed values and predicted values for expression $\mathbf{x}$:
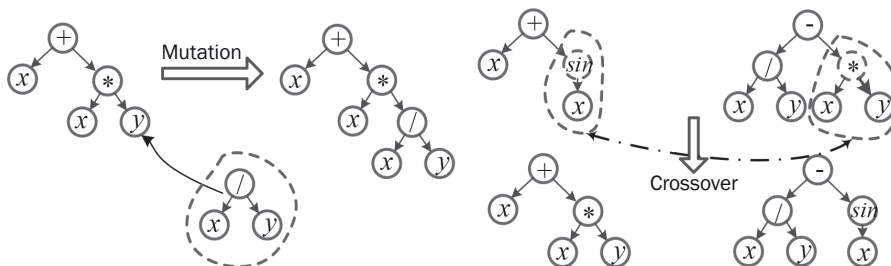

Figure 2 – Mutation and crossover of GP

$$R = R(\mathbf{x}) = \frac{\sum_{i=1}^{N}(\widehat{y}_i - \ddot{y})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N}(\widehat{y}_i - \ddot{y})^2}\sqrt{\sum_{i=1}^{N}(y_i - \overline{y})^2}} \qquad (2)$$

where y_i and $\hat{y}_i$ are the i-th prediction value and measured value. $\ddot{y}$ and $\overline{y}$ are the averages of prediction values and measured values, respectively.

When the measured value is low, RMSE always becomes sensitive, and therefore R^2 is selected in this study and the fitness is defined as $f_2(\mathbf{x})=1-R^2$ for expression $\mathbf{x} \in \Omega$, to qualify the accuracy of the model.

2.4 Pareto front

To find the balance of complexity and fitness, Pareto front is widely used in the multi-objective optimization. In the last stage of the symbolic regression, a large number of models is obtained. The Pareto front considers all models to be equally important and the aim is to identify the non-dominated ones which can be conducted after some definitions.

Definition of Pareto dominance

We define a binary relation of Pareto dominance and write $\mathbf{x}_1 \succ \mathbf{x}_2$ saying that the solution (expression) $\mathbf{x}_1$ dominates the solution (expression) $\mathbf{x}_2$ if both following conditions are true:

$$\begin{cases} \forall i \in \{1,2\} f_i(\mathbf{x}_1) \le f_i(\mathbf{x}_2) \\ \exists j \in \{1,2\} f_j(\mathbf{x}_1) < fj(\mathbf{x}_2) \end{cases} \qquad (3)$$

Definition of Pareto optimality

If no other solution dominates $\mathbf{x}^*$, it is said to be Pareto optimal. Notice that it is not necessarily unique. In the mathematical form

$$\nexists \mathbf{x} \in \Omega \quad \mathbf{x} \succ \mathbf{x}^* \qquad (4)$$

where Ω is a feasible region.

Definition of Pareto front

For a given multi-objective optimization problem, a Pareto front (frontier, boundary) P is a set consisting of all Pareto optimal vectors:

$$P = \{\mathbf{x} \in \Omega \mid \nexists \mathbf{x}' \in \Omega \quad \mathbf{x}' \succ \mathbf{x}\} \qquad (5)$$

The Pareto front is a set of non-dominated points. In other words, there are no other points better than the points of the Pareto front in both complexity and fitness. In particular, the point in the lower left of the Pareto front (high accuracy and low complexity) is the best solution to the multi-objective optimization problem. An example of the technique is presented in *Figure 4*. All models are plotted with $1-R^2$ as the vertical axis, and the expressional complexity as the horizontal axis. The Pareto front is highlighted in green and the point with a red circle is the solution to the problem.
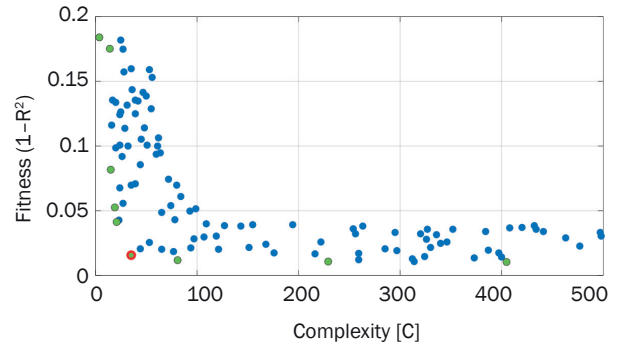

Figure 4 – Example of the Pareto front set

3. PERFORMANCE ANALYSIS

3.1 Data and evaluation criteria

The data used in this paper are extracted from the monitoring system of Xi-Cheng freeway, Jiangsu, China which is 22 km and has three lanes in each direction. The point in *Figure 5* represents the position of a microwave sensor which is selected as the object of this study.
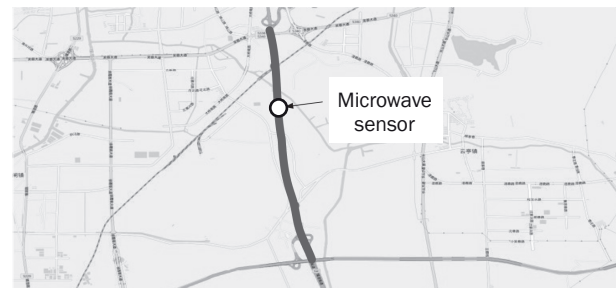

Figure 5 – Location of the microwave sensor (Source: Openstreetmap)

In the monitoring system, the speed is measured every five minutes in each lane. To obtain the speed v of a road section, an ensemble average algorithm is presented as follows:

$$v = \frac{v_1 V_1 + v_2 V_2 + v_3 V_3}{V_1 + V_2 + V_2} \qquad (6)$$

where V_1, V_2, V_3 is speed in lane 1, lane 2 and lane 3, respectively; V_1, V_2, V_3 is volume in lane 1, lane 2 and lane 3, respectively.

The final dataset includes 5 weekdays from June 8th 2015 to June 12th 2015. However, in the dataset, 30 records (2.08%) are missing. To minimize the effect of the missing values, they are filled by the average of values measured at the same time of the day from other weekdays. The summarization of the data are shown in *Table 1*.

Full dataset is divided into two parts: a training dataset and a testing dataset. The training dataset, including the first four days, is used to build the

Table 1 – Summarization of the initial data

Date	Minimum	1st quarter	Median	Mean	3rd quarter	Maximum
1	50.56	82.19	88.99	89.65	97.54	101.65
2	35.09	83.63	86.71	92.36	94.25	99.50
3	50.76	86.25	85.61	89.85	93.13	99.81
4	47.73	81.85	86.16	90.42	92.87	99.55
5	30.95	81.43	86.46	84.07	90.48	100.74

prediction models. The last day presents a testing dataset to evaluate the performance of the models with root mean square error (RMSE) and mean absolute percentage error (MAPE). RMSE and MAPE are accuracy measurements commonly used to evaluate traffic prediction models. They are defined in the following expressions:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\widetilde{v}_i - v_i\right)^2} \qquad (7)$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|\widetilde{v}_i - v_i\right|}{v_i} \cdot 100\% \qquad (8)$$

where v_i is the measured value of the i-th time interval and $\tilde{v}_i$ is the predicted value of the i-th time interval; N is the total number of the testing intervals.

3.2 Experiment design

Input is the data before the time when prediction starts and the length must be known in advance (as is shown in *Figure 6*). For ARIMA, its strength is to capture the temporal characteristic of the traffic speed, and the length is determined by parameters *p* and *q*. For SR and SVR, the input is set by the researchers. The length of input vector significantly affects the prediction accuracy. It is necessary to analyse the influence of the length of the input data. In the first experiment, SR and SVR are built with different vectors having 1, 2..., 10 lagged elements to study the relation between length of input and prediction accuracy.

In the second experiment, three different models are compared. ARIMA model is executed in R with the package 'forecast'. SVR is executed in R with the package 'e1071'. The code for the presented model SR is written in Matlab using the GPTIPS 2 tool box, which is available online [36]. These two models are used
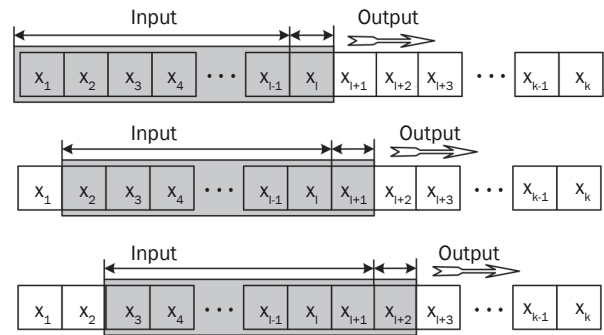

Figure 6 – Input of the models

to compare against SR for the following two reasons. First, ARIMA is a statistics-based model and SVR is a computational intelligence-based model. They can represent traffic prediction models from the main perspectives. Second, ARIMA and SVR were proven to be more accurate than other models among the above two types of models in previous studies. The SR model is detailed in section 3 because it is the focus of this paper. The formulations of ARIMA and SVR were not given because they were only used for comparison. We added references in which the readers can find the detail procedures of the ARIMA and SVR models used in this study [17].

3.3 Parameter selection

The parameters of ARIMA model are *p*, *d*, *q* - *p* is the order of autoregression; *d* is the order of differentiation, and *q* represents the order of the moving average. They can be obtained by the autocorrelation function and partial autocorrelation (PACF) plots. In the SVR, radial basis function is selected as kernel and two parameters γ, *C* are set with grid search

Table 2 – Parameters of the symbolic regression

SR	ARIMA	SVR
Number of individuals in the generation=300	*p*=2	$\gamma=1.95\cdot10^{-3}$
Maximum amount of time to run=180 s	*d*=1	*C*=8
Multi-gene option=true	*q*=3	
Maximum number of genes=4		
Maximum depth of tree=5		
Node functions={'times', 'minus', 'plus', 'square', 'sin', 'cos'}		

method. Coefficient *C* governs the relative importance of the penalty function; γ controls the width of the kernel function. Initially, parameters are set in a large interval and then, with *k*-fold cross-validation, the interval can be narrowed. In several cases, the best parameters are obtained. The parameters of SR are set considering the requirement of the prediction interval and to capture the non-linear pattern. Parameters of the algorithms are shown in *Table 2*.

3.4 Experiment 1: Analysis of the input length

The results of experiment 1 are presented in *Figure 7*. When the length of the input vector equals 6, the MAPE of SR is the lowest. The SVR gets its lowest MAPE as the length of the input vector is 5. At the beginning (length of input vector is less than 3), the MAPE of SR is higher than SVR. However, as the length increases after 3, the MAPE of SR becomes lower than the MAPE of SVR and both fluctuate in a narrow range.

From the time-consuming aspect, the training time and the testing time will increase with the increase of the input vector length. In practice, the prediction model of intelligent transportation system has strict requirements for its real-time. Accordingly, to consider the time-accuracy trade-off of the models and make a fair comparison, in the following section two models are trained with the same vector having 3 elements as input.
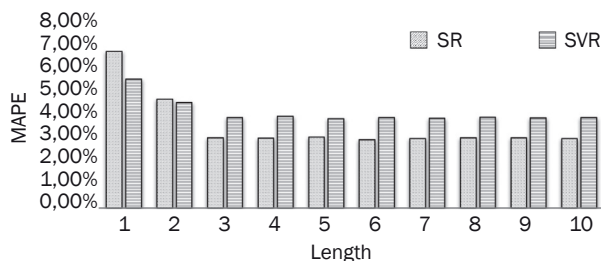

Figure 7 – The influence of the length of input vector

3.5 Experiment 2: Comparison of three models

The results of the three different models are shown in *Figure 8* together with MAE (Mean Absolute Error). The oval shows the performance of the model when an incident happens. It results in a congestion and the speed dramatically changes. The fluctuation of the

prediction values of three models indicates that they can capture the general tendency of the real data.

However, in some specific points the error of the three models is different. When the speed fluctuates within a small range but dramatically, the performance of the ARIMA appears to be worse shown by the green oval (*Figure 8 a*). SVR outperforms ARIMA and it seems to capture the sudden change of the speed (*Figure 8 b*). Obviously, SR cannot only capture the pattern in normal period but also performs well under incident (*Figure 8 c*).

To evaluate the performance of the models under different traffic conditions, the speed is divided into four regimes: 20-40 km/h, 40-60 km/h, 60-80 km/h and 80-100 km/h. For each regime, VAPE and RMSE are calculated (*Table 3*). The lowest RMSE and MAPE reflect that SR outperforms the ARIMA and SVR in all regimes, especially in 20-40 km/h and 40-60 km/h. In *Figure 8* SR line of the predicted values follows the measured values while two other models show more discrepancy and higher MAE after an incident occurs.

4. CONCLUSION

In this study, a simple and interpretable symbolic regression based on genetic programming is proposed to predict the traffic speed. Using real world dataset from Xi-Cheng freeway, Jiangsu, China, the model is compared with the traditional models SVR and ARIMA. The results show SR is the most accurate. It cannot only take advantage of the computational intelligence-based models but also it is interpretable. To evaluate the performance of the models in abnormal condition, the test day including an incident was used. The result shows that the presented model is good in predicting the traffic speed during all traffic states. In summary, the proposed traffic speed prediction model outperforms SVR and ARIMA.

In fact, traffic information retrieved by just one sensor, as considered in this paper, is not enough to accurately represent traffic conditions at an entire freeway segment, unless such segment is small enough to ensure homogenous characteristics of traffic and infrastructure. In the future, we will obtain data from lots of sensors and the spatial information of the traffic speed will be taken into consideration to enhance the accuracy.

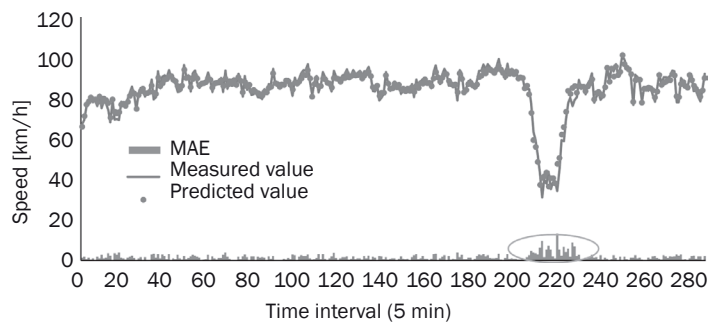*Table 3 – Evaluation criteria of the three models*

Regimes	RMSE			MAPE [%]		
	SVR	SR	ARIMA	SVR	SR	ARIMA
20-40	11.20	6.20	19.15	5.06	2.30	7.19
40-60	7.22	4.32	17.46	4.18	2.48	10.36
60-80	5.38	2.53	7.74	4.55	2.16	6.42
80-100	1.89	1.55	2.91	2.06	1.66	3.18
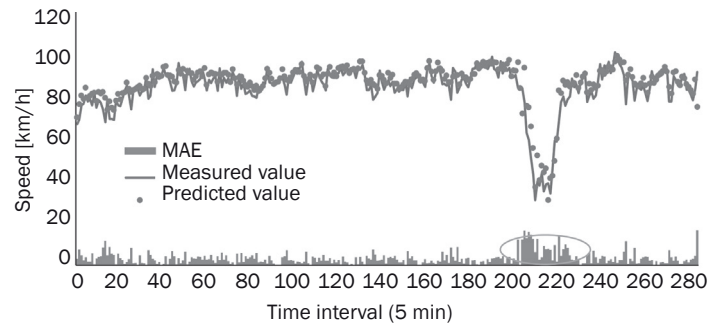
a) Result of ARIMA
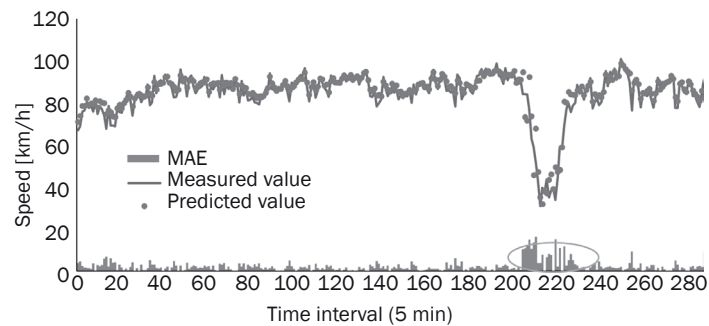
b) Result of SVR

c) Result of SR

Figure 8 - Comparison of the three different models

李林超，博士研究生[1,2]
电子邮箱：linlinchao123@163.com
TOMISLAV FRATROVIĆ，博士[3]

张健，博士（通讯作者）[1,2]
电子邮箱: jianzhang@seu.edu.cn
冉斌，博士[1,2]
[1] 东南大学交通学院
  中国江苏省南京市四牌楼2号
[2] 东南大学物联网应用研究中心
  中国江苏省南京市四牌楼2号
[3] University of Zagreb,
  Faculty of Transport and Traffic Sciences
  Vukelićeva 4, 10000 Zagreb, Croatia

基于符号回归算法的高速公路速度预测研究
摘要
随着高速公路交通拥堵的日益严峻，提供实时交通状态信息有助于缓解该问题，因而受到了广泛的关注。本论文利用符号回归算法构建高速公路速度预

测模型，旨在向出行者提供更加准确的交通信息。模型以遗传算法为基础构建回归簇，以模型复杂度和精度为优化目标，利用帕累托前沿理论，求解出最优回归。通过微波检测器获取的数据，将本文提出的模型与传统的预测模型进行对比，结果表明：本文提出的模型不仅在正常交通状态下具有较高的精度，在交通事件发生时同样可以获取较好的预测结果。表明，本文提出的模型具有较好的稳定性和抗干扰性，可以应用于实践。

关键字

交通拥堵；交通状态；检测器数据；速度预测；事件；符号回归；基因算法；

REFERENCES

[1] Smith BL, Williams BM, Oswald RK. Comparison of parametric and nonparametric models for traffic flow forecasting. Transportation Research Part C: Emerging Technologies. 2002;10(4):303-21.

[2] Pan T, Sumalee A, Zhong R-X, Indra-Payoong N. Short-term traffic state prediction based on temporal–spatial correlation. IEEE Transactions on Intelligent Transportation Systems. 2013;14(3):1242-54.

[3] Sun H, Liu H, Xiao H, He R, Ran B. Use of local linear regression model for short-term traffic forecasting. Transportation Research Record: Journal of the Transportation Research Board. 2003(1836):143-50.

[4] Zhang X, Rice JA. Short-term travel time prediction. Transportation Research Part C: Emerging Technologies. 2003;11(3):187-210.

[5] Kwon J, Coifman B, Bickel P. Day-to-day travel-time trends and travel-time prediction from loop-detector data. Transportation Research Record: Journal of the Transportation Research Board. 2000(1717):120-9.

[6] Lippi M, Bertini M, Frasconi P. Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning. IEEE Transactions on Intelligent Transportation Systems. 2013;14(2):871-82.

[7] Lee S, Fambro D. Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. Transportation Research Record: Journal of the Transportation Research Board. 1999;(1678):179-88.

[8] Williams BM, Hoel LA. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. Journal of transportation engineering. 2003;129(6):664-72.

[9] Szeto W, Ghosh B, Basu B, O'Mahony M. Multivariate traffic forecasting technique using cell transmission model and SARIMA model. Journal of Transportation Engineering. 2009;135(9):658-67.

[10] Mai T, Ghosh B, Wilson S, editors. Multivariate short-term traffic flow forecasting using Bayesian vector autoregressive moving average model. Transportation Research Board 91st Annual Meeting; 2012.

[11] Ghosh B, Basu B, O'Mahony M. Multivariate short-term traffic flow forecasting using time-series analysis. IEEE Transactions on Intelligent Transportation Systems. 2009;10(2):246-54.

[12] Kamarianakis Y, Prastacos P. Space–time mod-eling of traffic flow. Computers & Geosciences. 2005;31(2):119-33.

[13] Okutani I, Stephanedes YJ. Dynamic prediction of traffic volume through Kalman filtering theory. Transportation Research Part B: Methodological. 1984;18(1):1-11.

[14] Guo J, Huang W, Williams BM. Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. Transportation Research Part C: Emerging Technologies. 2014;43:50-64.

[15] van Hinsbergen CP, Schreiter T, Zuurbier FS, Van Lint J, van Zuylen HJ. Localized extended kalman filter for scalable real-time traffic state estimation. IEEE transactions on intelligent transportation systems. 2012;13(1):385-94.

[16] Kalman RE. A new approach to linear filtering and prediction problems. Journal of basic Engineering. 1960;82(1):35-45.

[17] Dong C, Richards SH, Yang Q, Shao C. Combining the statistical model and heuristic model to predict flow rate. Journal of Transportation Engineering. 2014;140(7):04014023.

[18] Halawa K, Rusiecki A, Bazan M, Janiczek T, Ciskowski P, Kozaczewski P. Road Traffic Predictions Across Major City Intersections using Multilayer Perceptrons and Data from Multiple Intersections Located in Various Places. IET Intelligent Transport Systems. 2016.

[19] Yasin Çodur M, Tortum A. An Artificial Neural Network Model for Highway Accident Prediction: A Case Study of Erzurum, Turkey. Promet - Traffic & Transportation. 2015;27(3):217-25.

[20] Li L, He S, Zhang J, Ran B. Short-term highway traffic flow prediction based on a hybrid strategy considering temporal-spatial information. Journal of Advanced Transportation. 2016;50(8):2029-2040.

[21] Cai P, Wang Y, Lu G, Chen P, Ding C, Sun J. A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. Transportation Research Part C: Emerging Technologies. 2016;62:21-34.

[22] Yu B, Song X, Guan F, Yang Z, Yao B. k-Nearest Neighbor Model for Multiple-Time-Step Prediction of Short-Term Traffic Condition. Journal of Transportation Engineering. 2016;142(6):04016018.

[23] Nikovski D, Nishiuma N, Goto Y, Kumazawa H, editors. Univariate short-term prediction of road travel times. Proceedings 2005 IEEE Intelligent Transportation Systems, 2005; 2005: IEEE.

[24] Kindzerske M, Ni D. Composite nearest neighbor nonparametric regression to improve traffic prediction. Transportation Research Record: Journal of the Transportation Research Board. 2007;(1993):30-5.

[25] Vlahogianni EI, Karlaftis MG, Golias JC. Short-term traffic forecasting: Where we are and where we're going. Transportation Research Part C: Emerging Technologies. 2014;43:3-19.

[26] Jin J, Ran B. Automatic freeway incident detection based on fundamental diagrams of traffic flow. Transportation Research Record: Journal of the Transportation Research Board. 2009;(2099):65-75.

[27] Golafshani EM, Ashour A. Prediction of self-compacting concrete elastic modulus using two symbolic

regression techniques. Automation in Construction. 2016;64:7-19.

[28] Koza JR. Genetic programming: on the programming of computers by means of natural selection. MIT press; 1992.

[29] Vladislavleva E, Friedrich T, Neumann F, Wagner M. Predicting the energy output of wind farms based on weather data: Important variables and their correlation. Renewable energy. 2013;50:236-43.

[30] Yang G, Li X, Wang J, Lian L, Ma T. Modeling oil production based on symbolic regression. Energy Policy. 2015;82:48-61.

[31] Pandey DS, Pan I, Das S, Leahy JJ, Kwapinski W. Multi-gene genetic programming based predictive models for municipal solid waste gasification in a fluidized bed gasifier. Bioresource technology. 2015;179:524-33.

[32] Xu C, Wang W, Liu P. A genetic programming model for real-time crash prediction on freeways. IEEE Transactions on Intelligent Transportation Systems. 2013;14(2):574-86.

[33] Tang J, Zhang G, Wang Y, Wang H, Liu F. A hybrid approach to integrate fuzzy C-means based imputation method with genetic algorithm for missing traffic volume data estimation. Transportation Research Part C: Emerging Technologies. 2015;51:29-40.

[34] Yu B, Gu X, Ni F, Guo R. Multi-objective optimization for asphalt pavement maintenance plans at project level: Integrating performance, cost and environment. Transportation Research Part D: Transport and Environment. 2015;41:64-74.

[35] Linchao L, Fratrović T. Analysis of Factors Influencing the Vehicle Damage Level in Fatal Truck-Related Accidents and Differences in Rural and Urban Areas. Promet - Traffic & Transportation. 2016;28(4):331-40.

[36] Searson DP. GPTIPS 2: an open-source software platform for symbolic data mining. Handbook of genetic programming applications. Springer; 2015. p. 551-73.

[37] Smits GF, Kotanchek M. Pareto-front exploitation in symbolic regression. Genetic programming theory and practice II. Springer; 2005. p. 283-99.