sustainability

Article

A Novel Online Prediction Method for Vehicle Velocity and Road Gradient Based on a Flexible-Structure Auto-Regressive Integrated Moving Average Model

Bin Ma [1,2], Penghui Li [3,*], Xing Guo [4], Hongxue Zhao [5] and Yong Chen [1,2]

[1] School of Mechanical and Electrical Engineering, Beijing Information Science and Technology University, Beijing 100192, China; mbhbgc1985@bistu.edu.cn (B.M.); chenyong_jz@126.com (Y.C.)
[2] Beijing Laboratory for New Energy Vehicles, Beijing 100192, China
[3] Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China
[4] School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; gx074153690@163.com
[5] Research Institute of Highway Ministry of Transport, Beijing 100088, China; zhxue1989@126.com
* Correspondence: penghuil@bjtu.edu.cn

Abstract: The auto-regressive integrated moving average (ARIMA) model has shown promise in predicting vehicle velocity and road gradient (V–G) for the purpose of constructing power demands in predictive energy management strategies (PEMS) for electric vehicles (EVs). It offers flexibility, accuracy, and computational efficiency. However, the performance of a conventional ARIMA model with fixed structure parameters can be disappointing when the data fluctuate. To overcome this limitation, a novel and flexible-structure-based ARIMA (FS–ARIMA) is proposed in this paper to improve online prediction performance. First, the sliding window method was developed to produce fitting data in real time based on real local historical data, reducing the online computation time. Secondly, the influence of the sliding window sample size, differencing order, and lag in the model on the prediction accuracy was investigated. Based on this, an FS–ARIMA was proposed to improve the prediction accuracy, where an augmented Dickey–Fuller (ADF) test was developed to select the differencing order in real time and the Bayesian information criterion (BIC) was applied to update the model and determine its lag under an optimal sample size. Lastly, to validate the proposed FS–ARIMA, simulations were conducted using two typical driving cycles collected via experiments, as well as the following three typical driving cycles: the New European Driving Cycle (NEDC), the Urban Dynamometer Driving Schedule (UDDS), and the Worldwide Harmonized Light Vehicles Test Cycle (WLTC). The results demonstrated that FS–ARIMA improved prediction accuracy by approximately 41.63% and 42.19% for the velocity and gradient, respectively. The proposed FS–ARIMA prediction model has potential applications in predictive energy management strategies for EVs.

Keywords: ARIMA; velocity and road gradient; short-term prediction; flexible structure; BIC

check for updates

Citation: Ma, B.; Li, P.; Guo, X.; Zhao, H.; Chen, Y. A Novel Online Prediction Method for Vehicle Velocity and Road Gradient Based on a Flexible-Structure Auto-Regressive Integrated Moving Average Model. *Sustainability* **2023**, *15*, 15639. https://doi.org/10.3390/su152115639

Academic Editor: Aritra Ghosh

Received: 20 July 2023
Revised: 18 August 2023
Accepted: 3 September 2023
Published: 6 November 2023

1. Introduction

In electric vehicles (EVs), a predictive energy management strategy (PEMS) that considers future driving conditions is considered the most promising approach to achieve energy savings [1,2]. Among many driving condition items, the accuracy of a vehicle's velocity and the road gradient (V–G) are two main issues with constructing the power demand online [3] that significantly impact the performance of a PEMS. Without accurate V–G predictions, the performance of a PEMS can be disappointing [4]. However, in real-world driving situations, knowledge about V–G is often scarce. This is primarily due to the absence of vehicle–road communications and the lack of high-precision maps, especially in the early stages of intelligent transportation systems (ITS). Therefore, finding a solution to accurately predict V–G in the absence of real-time data and high-precision maps is of great importance for the effective implementation of a PEMS in an EV.

In recent years, numerous prediction strategies have been proposed to solve the V–G prediction issue. One such strategy is the autoregressive integrated moving average model (ARIMA), which offers flexibility, accuracy, and computational efficiency [5]. The key advantage of the ARIMA model is its ability to ensure online predictions by extracting internal correlations with local history information, which overcomes the challenge of the limited availability of typical data for offline training. Consequently, the ARIMA is suitable for online V–G prediction based on local historical data along a driving route. However, a conventional ARIMA has lags and fixed differencing orders and, therefore, offers less adaptability in random conditions, especially when the V–G experience rapid fluctuations. Therefore, improving an ARIMA model's prediction accuracy to a certain extent is meaningful, especially in these early stage of ITS, where there is a lack of typical data [6,7].

1.1. Literature Review

A PEMS that considered future V–G information was introduced into EVs several years ago, and has proven to be beneficial for prolonging battery life and minimizing fuel consumption [8]. In recent years, various strategies have been proposed to predict future V–G, aiming to improve the efficiency of a PEMS. Typical prediction strategies are artificial intelligence (AI) and stochastic-based methods [9]. AI-based methods, such as artificial neural networks (ANNs) and radial basis function neural networks (RBF-NNs), have demonstrated a strong predictive accuracy through their learning processes [10]. However, AI-based methods require a large number of representative samples for offline training, making them less suitable for online applications [11]. Additionally, those numerous training samples rely on high-accuracy GPS, or onboard radar, which are not widely available in these early stages of ITS or the automotive industry [6]. Consequently, the practical implementation of AI-based methods is hindered. Therefore, there is a need to develop low-cost and easy-to-implement methods that can predict future V–G along a driving route using local historical information.

Stochastic-based methods, such as Markov chains (MCs) [12] and ARIMA models, establish relationships between past and future features to enable predictions [13]. In general, MCs require additional calculations to generate stochastic Markov emissions, making them more computationally involved than AI-based methods [4]. On the other hand, ARIMA models are easily implemented and can overcome a lack of data, making them widely applicable in areas such as energy generation [14], wind-power predictions [15], and other fields with limited sample data. In recent years, ARIMA models have been introduced in EVs for online V–G prediction. In a past study [5], an ARIMA-based V–G predictor was developed and combined into a PEMS. The simulation results demonstrated that the ARIMA model could achieve reasonable V–G predictions without the need for external devices. In another study [16], day-ahead traffic flow was predicted using a functional time series approach that outperformed traditional ARIMA models. Overall, an ARIMA model is based on the assumption that a process is stationary (i.e., stationary in its differences), and its prediction accuracy will decay sharply when a series is nonstationary. Thus, traditional ARIMA models with fixed structures have a poor performance in predicting driving conditions when the V–G have significant fluctuations. In general, the accuracy of V–G predictions suffers from two main issues with constructing power demands online, impacting PEMS performance. Without accurate V–G predictions, the performance of a PEMS can be disappointing [4]. Therefore, improving an ARIMA model's prediction accuracy to align with a PEMS is meaningful.

To improve the prediction accuracy of an ARIMA model, three approaches are commonly employed: fitting the data online with an updating technique [17], hybrid approaches [18], and flexible approaches [10]. The online data fitting updating technique involves updating the training data in real time using a sliding window approach [19]. This technique ensures that the training data reflect the most recent measurements, reducing the mismatch between the training data and the actual data and improving the prediction accuracy. For example, [15] proposed a sliding window-based ARIMA for wind speed forecasting, which reduces the overall RMSE by 75% for daily predictions and 50% for weekly predictions. In short, the sliding window technique allows for accurate predictions at a minimal computational cost. Hybrid approaches combine ARIMA with other machine learning methods to compensate for the nonlinear limitations of ARIMA, such as ARIMA-LSTM [20] and ARIMA-FSVR [21], where LSTM (long short-term memory) and FSVR (fuzzy support vector regression) are combined with ARIMA, respectively. These hybrid approaches have led to a significant improvement in prediction accuracy compared to using ARIMA, LSTM, or FSVR alone. However, it is important to note that hybrid approaches increase the complexity of the algorithm and may limit practical applications.

Currently, flexible approaches are being widely utilized to optimize strategy structures and improve adaptability. In [10], an adaptive radial basis function neural network (ARBF-NN) with flexible width and order was designed to perform online vehicle velocity prediction via the Akaike information criterion (AIC) and Bayesian information criterion (BIC). The results show that the adjusted structure in real time could further improve the prediction accuracy. Generally, AIC and BIC methods are commonly used for online selection of strategy structures. In [22], the kernel dictionary selection and weight vector were optimized with AIC in real time to ensure the time series was stationary, thus improving the prediction performance significantly. These flexible-structure approaches focus on online parameter updates and ensuring series stationarity, resulting in improved strategy performance and adaptability. In ARIMA, the differencing order and lags of the model both belong to the structure parameter, which determines the series stationary and the fitting order and thus affects the prediction accuracy. Therefore, inspired by the above content, the sliding window manner and flexible approaches could improve the ARIMA prediction performance [23]. However, further research is required to explore this more extensively.

1.2. Research Gaps

In this manuscript, a novel flexible-structure-based ARIMA (FS–ARIMA), with a variable differencing order and lags of the model, was designed to improve the V–G prediction accuracy. Overall, FS–ARIMA offers the following advantages:

(1) The sliding windows technology is used to generate the fitting data in real time, ensuring prediction accuracy with less computational effort;
(2) A theoretical structure determination method is provided. The differencing order and lags of the model have been adjusted adaptively via the augmented Dickey–Fuller (ADF) test and BIC;
(3) No external devices or massive historical database for offline training are required in this approach;
(4) The effectiveness of the FS–ARIMA is validated with two actual and typical driving cycles compared with LSTM, RBF, and ARBF.

The structure of the paper is as follows: Section 2 introduces the mathematical model of ARIMA with sliding window technology. Section 3 presents the proposed FS–ARIMA in detail; Section 4 discusses the performance of the FS–ARIMA and provides an overall evaluation. Finally, Section 5 summarizes the key findings and proposes directions for future research.

2. Velocity and Road Gradient Prediction with ARIMA

2.1. Actual Driving Cycle Collection

Figure 1 depicts an inertial navigation measurement system used for collecting actual V–G data. A gyroscope is used to test the velocity data; two external GNSS are used to obtain the longitudinal and vertical position information, which can be converted into road gradient data. The iNAV2 is a signal processor within the system, processing with an extended Kalman filter to eliminate the measured noise. The system operates at a frequency of 50 Hz, guaranteeing measurement accuracy and responsiveness to changes in the V–G data. Finally, 1 s interval data series are processed in an industrial computer and used for the online prediction afterward.

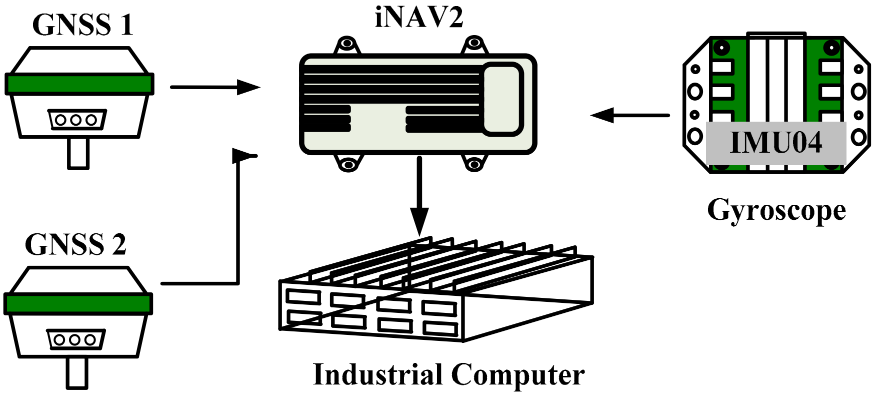


Figure 1. The inertial navigation measurement system.

In this study, three different driving cycles were chosen from Beijing, which include sections of the highway, the third ring road, and the fourth ring road, as shown in Figure 2. To thoroughly evaluate the performance of the proposed FS–ARIMA algorithm, three combined driving cycles were chosen, namely Actual 1, Actual 2, and a combination of the NEDC (New European Driving Cycle), UDDS (Urban Dynamometer Driving Schedule), and WLTC (Worldwide Harmonized Light Vehicles Test Cycle). Figure 3 provides a visualization of these combined driving cycles, and more detailed information can be found in Table 1. These combined cycles were designed to cover a wide range of real-world driving scenarios and replicate different driving patterns and characteristics. The inclusion of these combined cycles ensures a comprehensive assessment of the FS–ARIMA algorithm under various driving conditions.

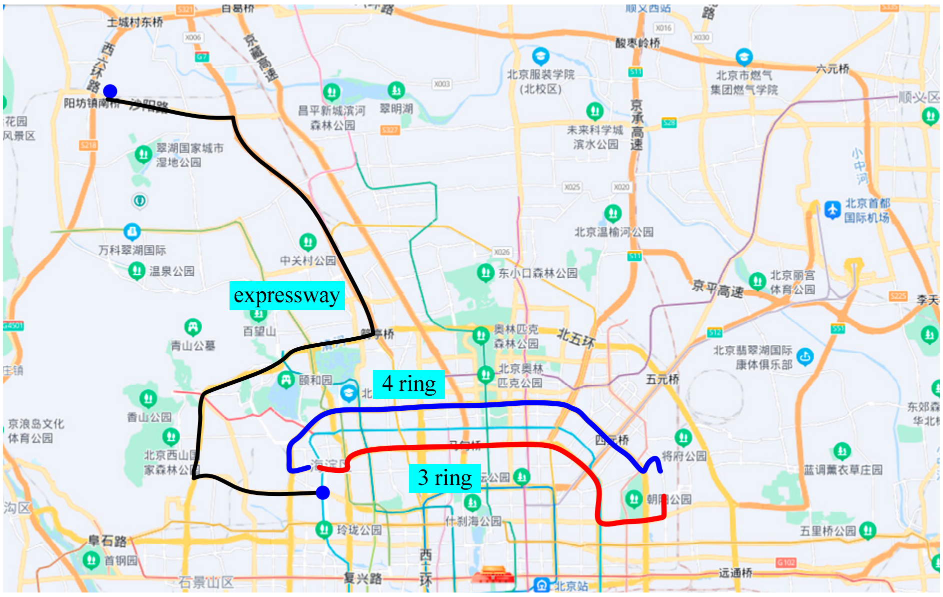


Figure 2. Three driving conditions in Beijing.

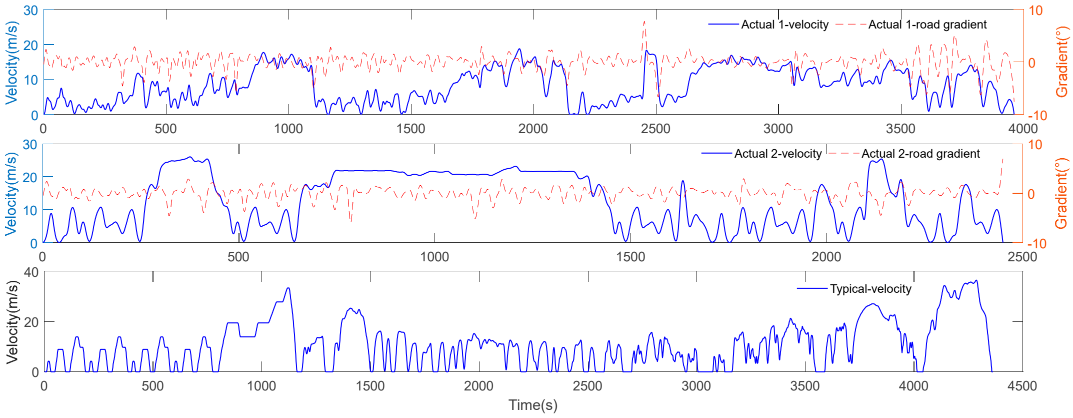


Figure 3. Three combined driving cycles: Actual 1, Actual 2, and typical cycles.

Table 1. The combination of actual and typical cycle information and road gradient cycles.

Parameter	Cycle	Length (s)	Describe
V	Actual 1	3740	Actual cycle combined with part 3-ring and part 4-ring
V	Actual 2	2450	Actual city cycle in daily life: contains an expressway section
V	Typical	4003	Combined NEDC, UDDS, and WLTC in a fixed sequential
G	Actual 1	3740	Actual cycle combined with part 3-ring and part 4-ring
G	Actual 2	2450	Actual city cycle in daily life: contains an expressway section

2.2. ARIMA Formulation

The ARIMA model is a popular choice for time-series forecasting as it provides a framework to analyze the underlying patterns and trends in stationary data. In this paper, the ARIMA is selected to predict the V–G. The general form of the ARIMA model is expressed as ARIMA (p, d, q), where p represents the order of the autoregressive (AR) model, d represents the order of differencing, and q represents the order of the moving average (MA) model. The ARIMA can be expressed as

$$X_t = c + \phi_1 x_{t-i} + \cdots + \phi_p x_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}, \tag{1}$$

where x_t is the stationary V–G time series, X_t is the prediction results, ϕ_i is the autoregressive coefficient of the AR sequence for the response process, θ_j is the moving average coefficient of the MA sequence for the stochastic process, and ε_t is the white noise random error sequence, which is assumed to be independent and contain identically distributed variables sampled from a Gaussian distribution with zero means.

In general, the procedure for constructing ARIMA (p, d, q) involves five iterative steps: First, the stationarity is checked to determine the differencing order of d; second, the lags of p and q parameters are determined; third, the coefficient of ϕ and θ is estimated; fourth, the model diagnosis is checked; finally, the forecast is conducted. The procedure of the ARIMA construction process is shown in Figure 4.

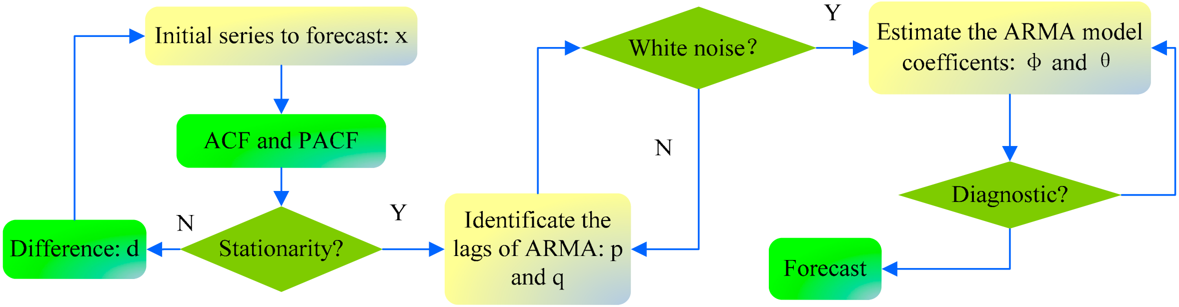

Initial series to forecast: x
ACF and PACF
Stationarity?
Difference: d
N
Y
Identificate the lags of ARMA: p and q
White noise?
N
Y
Estimate the ARMA model coefficents: ϕ and θ
Diagnostic?
Forecast

Figure 4. The procedure of ARIMA (p,d,q) construction.

In the stationary checking phase, the differencing order d is identified with the offline manner autocorrelation function (ACF) and partial autocorrelation function (PACF) via a judgment of autocorrelation function decays [5]. If a time series is nonstationary, the data need to be transformed by using the method of differencing. The first differencing procedure can be accomplished by

$$x_t = z_t - z_{t-1}, \tag{2}$$

where z_t is the raw data of the V–G series.

If the time series x_t is not stationary after the first differencing, the second difference needs to be determined by

$$x_t = (z_t - z_{t-1}) - (z_{t-1} - z_{t-2}). \tag{3}$$

When the second differencing does not provide a stationary time series, third or further differences should be implemented. Generally, the differencing should be lower than a particular value, because overdifferencing will cause a loss of autocorrelation.

In a stationary time series analysis, we usually define L as the lag operator, which represents an element of a time series to time the previous element. For the stationary time series of x_t, the L lag operation is

$$Lx_t = x_{t-1}, \text{ for all } t > 1. \tag{4}$$

Then, Equation (1) is formulated as follows:

$$\phi(L)x_t = c + \theta(L)\varepsilon_t, \tag{5}$$

where
$L^i x_t = x_{t-i};\ L^i \varepsilon_t = \varepsilon_{t-i}$
$\phi(L) = 1 - \phi L - \phi^2 L^2 - \cdots - \phi^p L^p$
$\theta(L) = 1 + \theta L + \theta^2 L^2 + \cdots + \theta^p L^q.$

Then, the stationary series can be used to determine the p and q parameters, which are the lags of the model. Generally, the lags of the model can be obtained by examining the sample ACF and PACF.

Afterward, in the estimation phase, the coefficients of the ARMA, ϕ and θ, can be estimated by the maximum likelihood estimator with the stationary series. The log-likelihood function $l(\widetilde{\beta};x)$ of the stationary V–G series x_t can be written as follows:

$$\begin{cases} \dfrac{\partial}{\partial \sigma_\varepsilon^2} l(\widetilde{\beta};x) = -\dfrac{n}{2\sigma_\varepsilon^2} + \dfrac{S(\widetilde{\beta})}{2\sigma_\varepsilon^4} = 0 \\ \dfrac{\partial}{\partial \widetilde{\beta}} l(\widetilde{\beta};x) = -\dfrac{1}{2}\dfrac{\partial \ln|\Omega|}{\partial \widetilde{\beta}} - \dfrac{1}{\sigma_\varepsilon^2}\dfrac{\partial S(\widetilde{\beta})}{2\partial \widetilde{\beta}} = 0 \end{cases}, \tag{6}$$

where

$$\begin{cases} S(\widetilde{\boldsymbol{\beta}}) = \boldsymbol{x}' \begin{bmatrix} \sum\limits_{i=0}^{\infty} G_i^2 & \cdots & \sum\limits_{i=0}^{\infty} G_i G_{i+n-1} \\ \vdots & & \vdots \\ \sum\limits_{i=0}^{\infty} G_i G_{i+n-1} & \cdots & \sum\limits_{i=0}^{\infty} G_i^2 \end{bmatrix}^{-1} \boldsymbol{x}_{\mathrm{t}} \\ \tilde{\boldsymbol{\beta}} = \left(\phi_1, \cdots, \phi_p, \theta_1, \cdots, \theta_q\right)^T \end{cases}$$

$$x_t = \{x_{t-1}, x_{t-2}, \cdots x_{t-p}\}.$$

Here, G is the Green's function of ARIMA.

Then, the goodness-of-fit of the coefficients is examined with LB (Ljung–Box) in the model diagnosis checking phase, which is also called a white noise test of the residual sequence [24]. The equation is as follows:

$$LB = n(n+2)\sum_{k=1}^{p}\left(\frac{\widetilde{\rho}_k^2}{n-k}\right), \quad (7)$$

where $\widetilde{\rho}_k$ is the coefficient of autocorrelation of the samples, and n is the number of serial periods. Finally, if the fitted model passes the diagnostic check, the model can be used to make the forecast.

2.3. The Sliding Window Method

V–G data are typically generated as a time series using sensors on board. However, using the entire V–G data cycle for prediction can be time-consuming. In order to strike a balance between code execution efficiency and computational effort, the sliding window technique is employed to extract a new sample from the local history V–G data for online fitting and prediction. The sliding window technique, as shown in Figure 5, is used to update the samples in a sequential manner. The procedure can be classified into two steps. Firstly, the sample size of the window is determined based on the total length of the time series. Then, a loop is used to slide the window along the time series, computing the results window by window, with a fixed sample size.

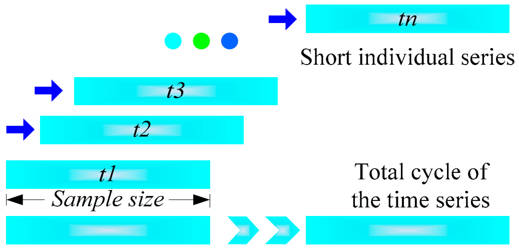


Figure 5. Sliding windows technique for sample updating.

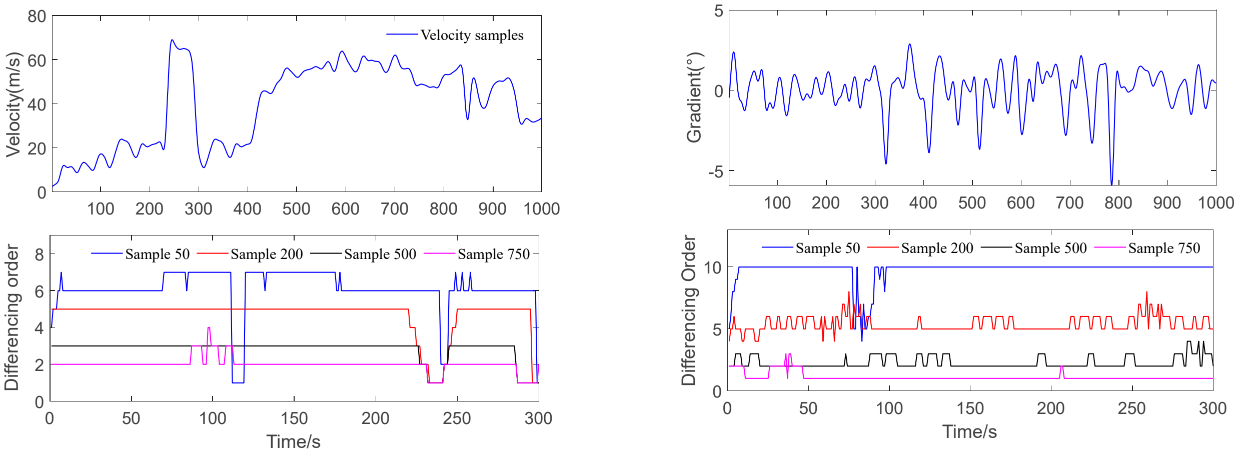
With the sliding window technique, the updated data include innovations in real time to eliminate differences between the forecast data and the history series. At t seconds, the input of ARIMA for autoregression can be represented as follows:

$$\boldsymbol{x} = \{x_{t-1}, x_{t-2}, \cdots x_{t-\chi}\}, \quad (8)$$

where χ is the sample size of the history series for V–G.

2.4. V–G Stationary Examination

In ARIMA, it is assumed that the time-series data are stationary for accurate prediction. However, the original V–G series is nonstationary. By using the sliding window technique in combination with the appropriate differencing order, the series can be made stationary. Figure 6 shows the variation in the differencing order with different sample sizes. It can be observed that the optimal value of d changes to maintain stationarity as the sample size of the sliding window increases. Figure 7 illustrates the variation in the average differencing order d to ensure the velocity series is stationary. The trend shows that as the sample size increases, the average d decreases and then stabilizes beyond a specific limit, regardless of the cycle. This suggests that a larger sample size, combined with a lower differencing order, ensures stationarity in the V–G series. Excessive differencing can lead to a loss of prediction accuracy and instability in the original sequence. Therefore, careful consideration and experimentation are required to determine the optimal values for the sample size and p, d, and q parameters to ensure effective predictions. In general, the sample size and p, d, and q have a homologous effect on the V–G series, and only the velocity series is illustrated in the following explanation.



(**a**) The velocity series

(**b**) The road gradient series

Figure 6. The variation in the differencing order *d* to ensure the series is stationary with different sample sizes of sliding windows.

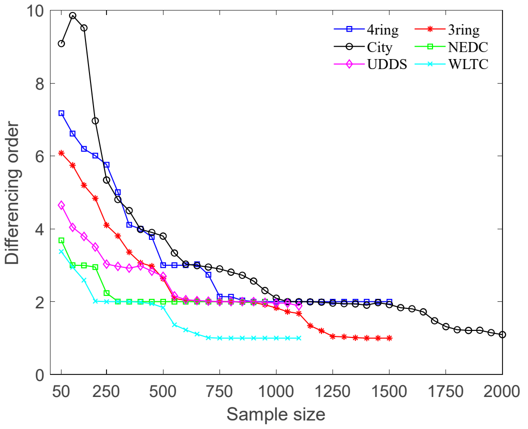


Figure 7. The *d* statistics feature with sample size in different velocity cycles.

2.5. Determination of Structural Parameters

Two error evaluation factors, the root-mean-square error (RMSE) and average root-mean-square error (ARMSE), are selected to evaluate the prediction performance. The functions are

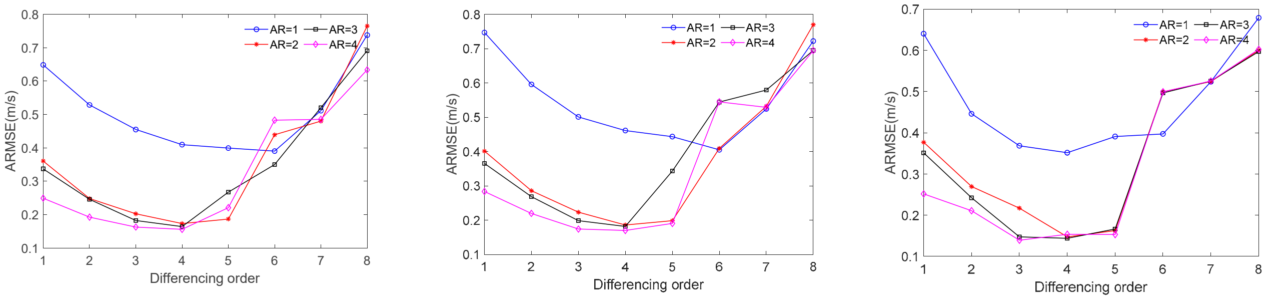
$$RMSE = \sqrt{\frac{1}{p}\sum_{k=1}^{p}(Y_p^k - Y_d^k)^2} \tag{9}$$

$$ARMSE = \frac{1}{M}\sum_{step=1}^{M} RMSE, \tag{10}$$

where Y_d^k is the original data value, Y_p^k is the value predicted with predictor, M is the number of time series, and p is the prediction horizon. Generally, lower values of RMSE and ARMSE indicate better performance of the forecasting task.

The sample size and p, d, and q have a coupled effect on prediction performance, and optimal flexible parameters could enhance the ARIMA prediction accuracy. In general, the parameter q corresponds to the stochastic part (white noise) and could be fixed ($q = 1$) without decaying the prediction performance. Thus, only the sample size, d, and p are considered flexible structural parameters used to analyze the coupling effect on prediction performance in depth.

The results presented in Figure 8 show the prediction accuracy under different sample sizes, 500, 1000, and 1500, with varying values of the differencing order d and lags of the AR model p. Based on the statistical results, it can be observed that the prediction accuracy remains excellent when the sample size is larger than 500. With a sample size of 500, lags in the model p affect the prediction in a particular range, then cooperate with differencing order d to maintain the prediction accuracy. Generally, the prediction has a preferable precision when the lags of the model p are higher than 2 and then remain at a certain level. However, as the differencing order d exceeds 5, the prediction performance begins to weaken due to excessive differencing, which can destabilize the original series. Therefore, it is crucial to carefully tune and select appropriate values for both d and p to ensure accurate predictions. In summary, a relatively larger sample size of 500, along with a lower number of lags in the model, with p set to 2, is recommended for updating the series. These parameters strike a balance between prediction accuracy and the stability of the original sequence.



(**a**) 500 sample size (**b**) 1000 sample size (**c**) 1500 sample size

Figure 8. The RMSE statistics with different sample sizes under the influence of differencing order and lags of the AR model.

From the results depicted in Figure 9, it can be observed that, regardless of the prediction method used, the prediction accuracy decreases significantly as the prediction horizon increases. The accuracy drops sharply when the prediction horizon exceeds 10 s. In addition, previous studies have also suggested that a prediction horizon of 10 s is beneficial for PEMS [25]. Therefore, it is reasonable to limit the prediction horizon to 10 s. Taking into account considerations such as code execution efficiency and calculation time, the recommended structure parameters are as follows: sample size of 500, d ranging from 2 to 4, and p ranging from 2 to 4. Figure 10 illustrates the performance of velocity prediction using the recommended structural parameters. The results indicate that the selected parameters ensure a superior prediction performance. Therefore, the online update of the structural parameters for ARIMA is meaningful and necessary.

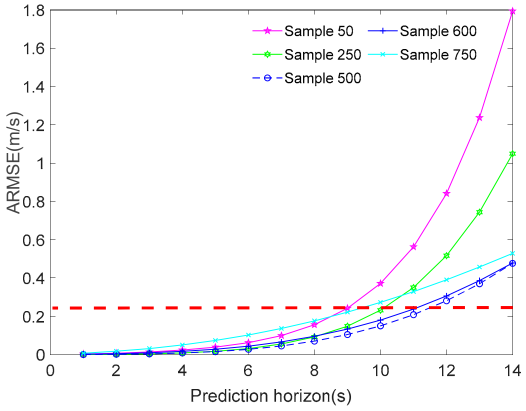


Figure 9. The sample size of the sliding window effect on the prediction accuracy with fixed p = 2.

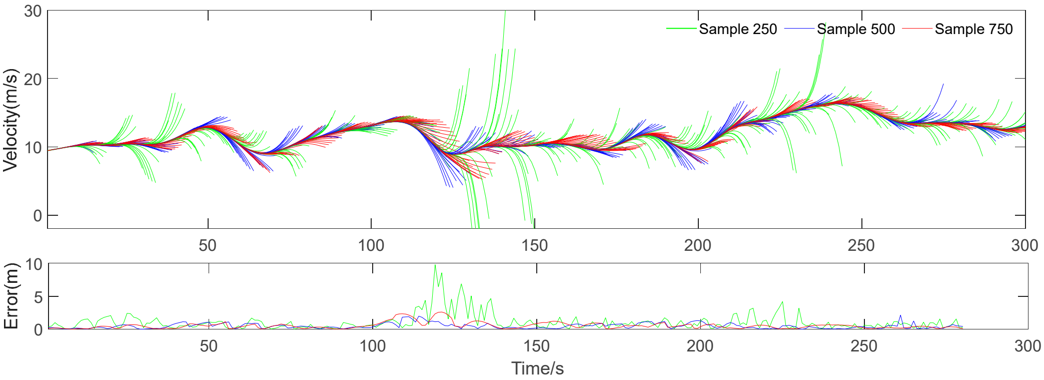


Figure 10. Prediction distribution with different sample sizes: p = 2, d = 2, q = 1.

3. Flexible-Structure-Based ARIMA Model

3.1. Selecting Differencing Orders with ADF Test

Previous research has demonstrated that a flexible structure can enhance prediction performance. However, it is important to ensure that the V–G series used for prediction is stationary, as per the fundamentals of ARIMA. To achieve this, the structural parameters are adjusted online to ensure stationarity and the consistency of statistical autocorrelation properties [18]. In this paper, the strict statistical method of the ADF test is used for online stationary checking. In principle, the inspection quantity τ of stationary series is lower at a hypothetical level. The function can be expressed as

$$\tau = \rho / S(\rho), \tag{11}$$

where τ is the inspection quantity and ρ and $S(\rho)$ are the ADF statistical and standard deviation of the inspection quantity, respectively.

To simplify the statistical calculation, we can judge the characteristic roots by whether $\rho < 0$ or not. This technology is also called the ADF unit root test. For the online application of the ADF unit root test, we rewrite the determining component in Equation (1) as follows:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \varepsilon_t. \tag{12}$$

Then, we can change the V–G stationary sequence x_t into differential variables. The equivalent expression of the differential is

$$\nabla x_t = \varphi x_{t-1} - \sum_{j}^{p-1} \xi_j \nabla x_{t-p+1} + \varepsilon_t \tag{13}$$

$$\text{i.e.}\begin{cases} \varphi = \phi_1 + \phi_2 + \cdots + \phi_p - 1 \\ \xi_j = \phi_{j+1} + \phi_{j+2} + \cdots + \phi_p, j = 1, 2, \ldots, p-1 \end{cases},$$

where ∇ is the forward difference of x_t and φ is the parameters of the statistical ADF statistical inspection quantity.

During each step of updating the V–G database, the corresponding ADF test statistics can be obtained to assess the stationarity of the series. Figure 11 presents the ADF statistical results using a sliding window sample size of 500. To avoid the loss of valid information, the maximum value of d is set to 4. According to the results, the ADF test statistics are always lower than the hypothetical level, indicating that the variable differencing order approach ensures the stationarity of the velocity series in real time. On the other hand, when a fixed differencing order is used, the constant statistical assumption is violated when the series exhibits significant fluctuations, despite potentially being higher than the hypothetical level in most cases. Thus, the variable differencing order approach is effective in ensuring the series remains stationary and contributes to the improvement of ARIMA prediction performance.

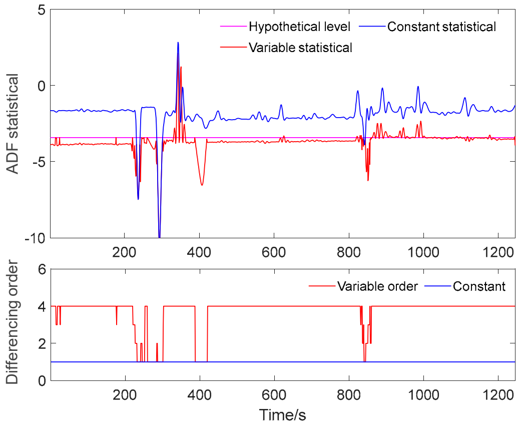


Figure 11. The ADF statistical results with variable differencing order: with a sample size of 500.

3.2. Lags of the Model Selection with BIC

The appropriate p and q also have a significant influence on the prediction precision. From the basics, the AIC and BIC may select the optimal model [26]. BIC is preferred for a case with a large sample set to prevent overfitting [11]. This study adopts BIC to select the optimal p online, and the smaller BIC value implies better prediction. The BIC can be expressed as follows:

$$BIC = (p + q + 2)\ln\hat{\sigma}_{\varepsilon}^{2} + 2\ln(p + q + 2), \tag{14}$$

where $\hat{\sigma}_{\varepsilon}^{2}$ is the maximum likelihood function value corresponding to the ARMSE.

In the ARIMA model, the MA component represents a stochastic process and affects the estimate less than the AR component. Additionally, most researchers prefer fitting the AR model rather than the MA model. The number of MA lags is set smaller than the number of AR lags to strike a balance between prediction accuracy and computational burden. Thus, the max p and q are set to 4 and 2, respectively. The selection of lag orders for one step of the AR and MA model is shown in Figure 12. In the current step, we chose p = 1 and q = 1 due to the minimum BIC being −6158. A comparison of BIC results along the whole series is presented in Figure 13. Based on these results, variable lags for the model yield smaller BIC values compared to using constant lags, and a smaller BIC value indicates a better fitting performance for the model. Specifically, the MA lag is consistently set at 2, while the AR lag fluctuates between 1 and 2. For the velocity series, the fixed p value indicates the main part of velocity is the response process, which has a tight relationship with previous data. Hence, to enhance velocity prediction accuracy, the stochastic process should be dynamically adjusted in real time. The results consist of the prediction accuracy with fixed lags of the model having an unsatisfactory prediction performance when the velocity series has fluctuations. In summary, the variable d ensures that the data are stationary and the optimal p and q correspond to the prediction accuracy.

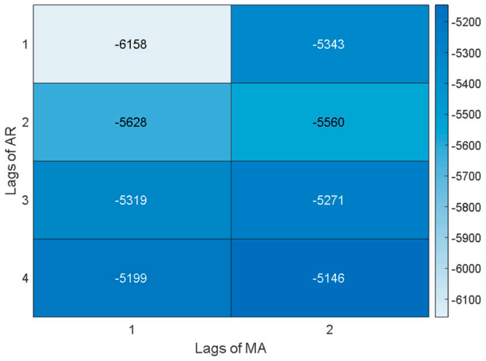


Figure 12. Heatmap order determination in one step of the BIC value.

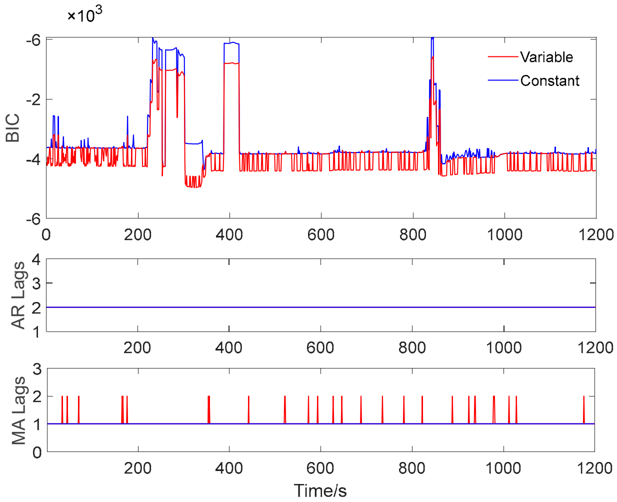


Figure 13. BIC results and variable lags of the AR and MA.

3.3. Overall Strategy Design

This paper proposes an FS–ARIMA approach with flexible structural parameters to improve short-term V–G prediction accuracy. The flowchart of the FS–ARIMA is shown in Figure 14. The specific implementation process of the strategy involves the following steps:

(1) New sample update with the sliding window: At time t, new local V–G data are collected and converted. The sliding window technique is used to update the sample data for online fitting and prediction. After the update, a new round of the prediction process begins;

(2) Stationary examination with variable d: The updated sample data undergo an ADF test, which determines the appropriate differencing order d in an adaptive manner. The initial value of d is set to 4 to avoid excessive differencing;

(3) Optimal p and q identification: The values of p and q, determined online with BIC via the stationary samples, are set to 4 and 2, respectively, to balance the fitting accuracy and computing time;

(4) The model parameter estimation and prediction module: The coefficient is estimated, and the ARIMA (p, d, q) prediction module is constructed for the forecast. It should be noted that least-squares regression is employed to estimate the coefficient in this step.

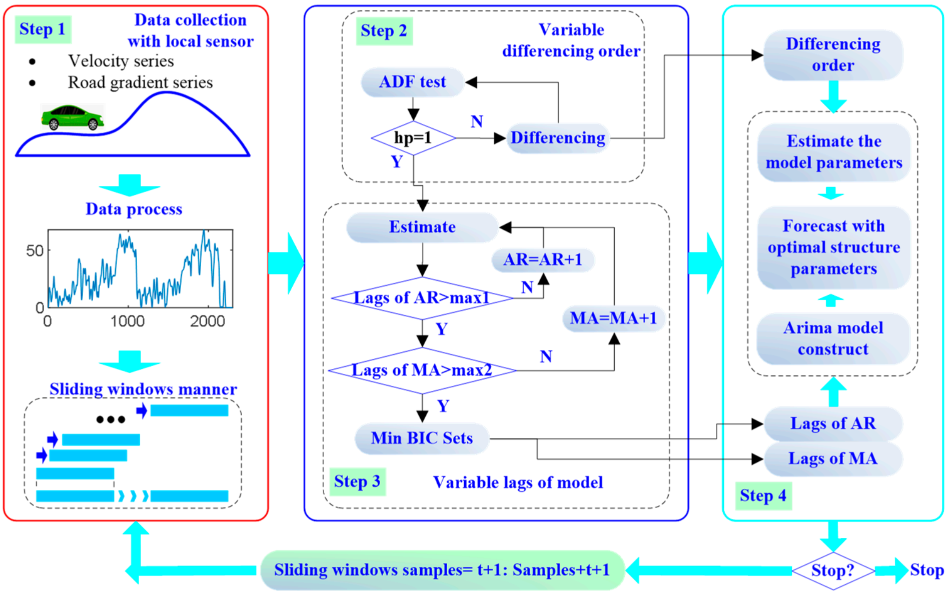


Figure 14. The flowchart of the adaptive structure parameters-based FS–ARIMA predictor.

By following this process, the FS–ARIMA approach enhances the prediction accuracy of short-term V–G data.

4. Results and Discussion

4.1. Performance of the Proposed FS–ARIMA

A typical prediction horizon of 10 s was selected to illustrate the prediction performance of FS–ARIMA. As a benchmark, the conventional ARIMA with fixed structure parameters was selected. Figure 15 illustrates the original velocity series of Actual 1 and the differential results with two differencing orders. According to the results, the differential of the velocity data was intuitively stationary and varied between −0.2 and 0.2. To further analyze the stationarity of the differential data, the autocorrelation function (ACF) and partial autocorrelation function (PACF) are examined in Figure 16. The majority of the ACF and PACF parameters fell within the desired boundaries. This indicates that the differential of the original Actual 1 velocity data was indeed stationary. Therefore, the ARIMA (2, 2, 1) model can be utilized for prediction purposes. Similarly, the other cycles were omitted from consideration as they exhibited similar conditions to the ARIMA (2, 2, 1) model.

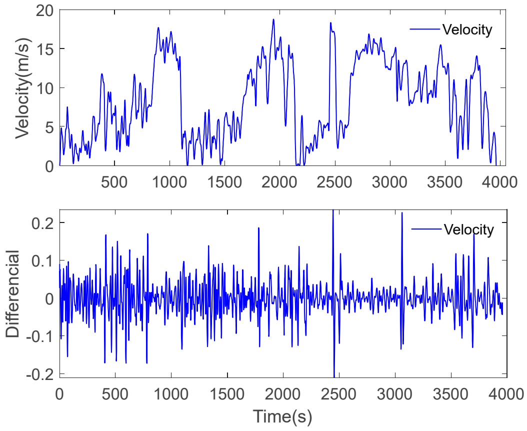


Figure 15. The original velocity series and differencing series with two−order differencing.

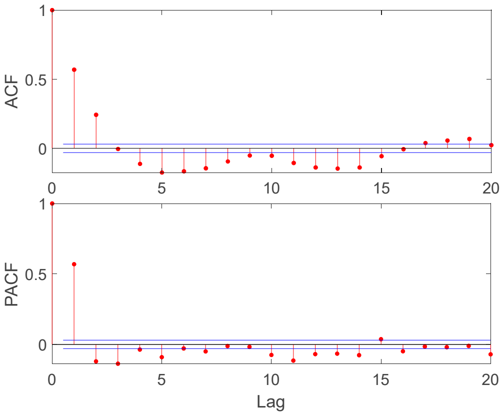
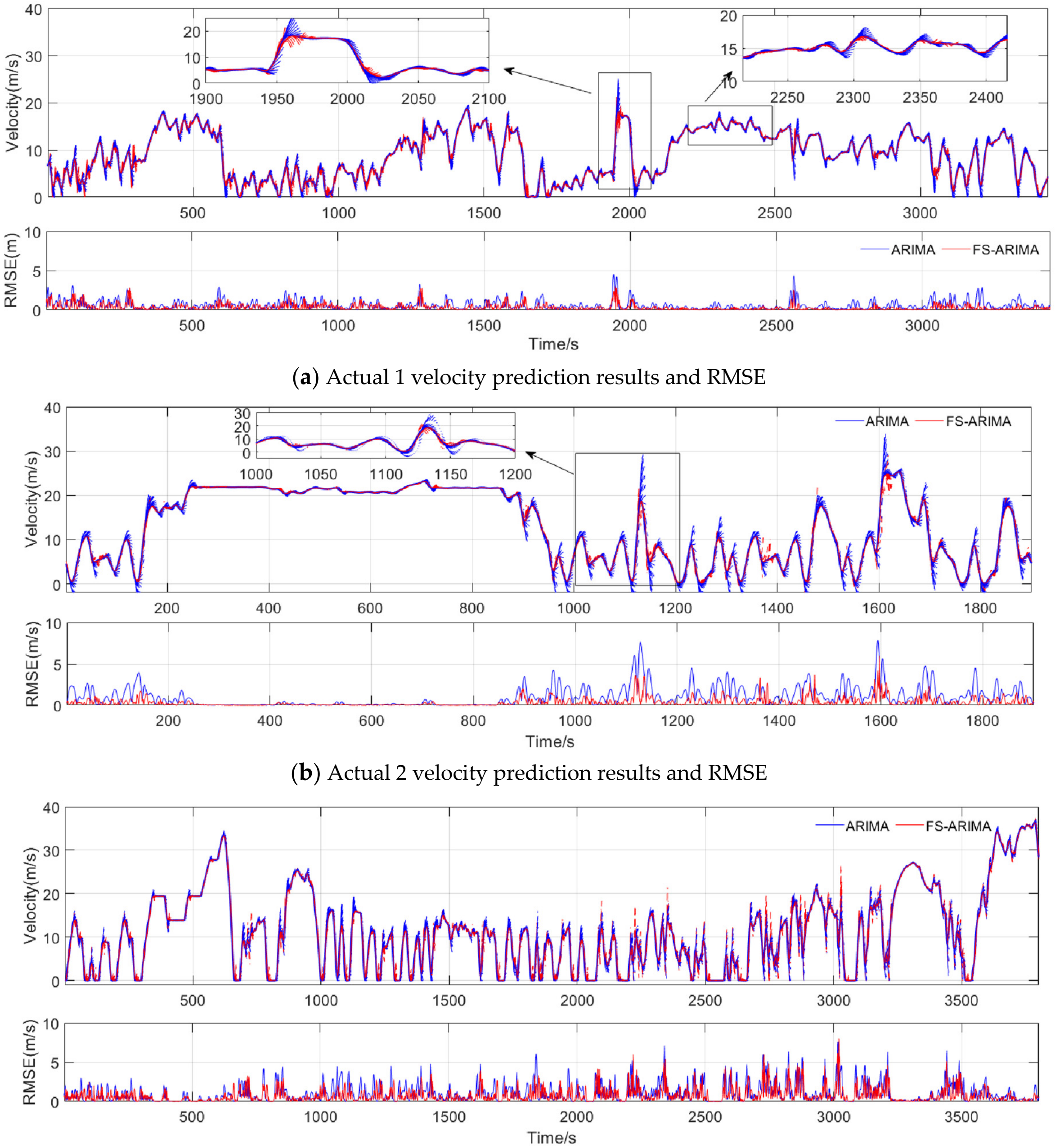


Figure 16. The ACF and PACF results of velocity series after two−order differencing.

Figures 17 and 18 show the performance of FS–ARIMA on V–G prediction compared with conventional ARIMA. In these figures, the blue and short red lines represent the forecasted values at each time point. It can be observed that the V–G prediction based on FS–ARIMA closely aligned with the actual V–G information at each time point, yielding a smaller RMSE. When the V–G was stationary or exhibited minor fluctuations, FS–ARIMA demonstrated a monotone-varying characteristic and excelled at capturing the varying characteristics. In these scenarios, the prediction was nearly perfect, with a relatively low RMSE—such as the 2000 s to 3500 s highway episode in the “actual 1” velocity series, as well as the entire cycle of "actual 2" velocity with traffic. However, when the V–G suddenly changed its variation trend, particularly when it reached its maximum point and subsequently declined dramatically, conventional ARIMA struggled to accurately predict the future road gradient, resulting in a relatively high RMSE. In such cases, the variable differencing order ensured the data were adaptively stationary, while the variable lags of the model enhanced the fitting accuracy. Consequently, FS–ARIMA achieved a remarkable improvement in performance. In summary, the proposed FS–ARIMA outperformed conventional ARIMA throughout the entire cycle.

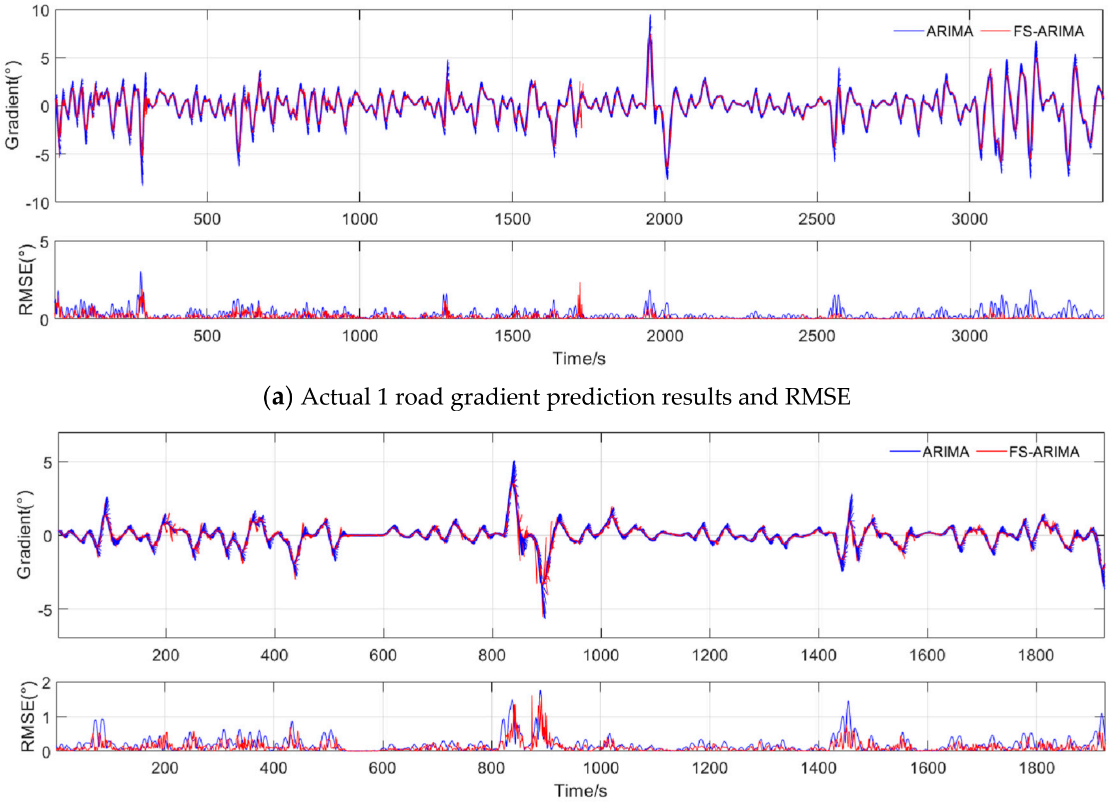


(**a**) Actual 1 velocity prediction results and RMSE

(**b**) Actual 2 velocity prediction results and RMSE

(**c**) Typical velocity prediction results and RMSE

Figure 17. Comparison of velocity prediction and RMSE results between FS–ARIMA and ARIMA: (**a**) Actual 1, (**b**) Actual 2, (**c**) Typical.



(**a**) Actual 1 road gradient prediction results and RMSE

(**b**) Actual 2 road gradient prediction results and RMSE

Figure 18. Comparison of road gradient prediction and RMSE results between FS−ARIMA and ARIMA: (**a**) Actual 1, (**b**) Actual 2.

The prediction accuracy and computational efficiency of V–G with the proposed FS–ARIMA under different prediction horizons are listed in Table 2. To measure the prediction performance, ARMSE was used, while the computational efficiency was quantified by the runtime. The reported time represents the average runtime across the various prediction horizons. In general, the ARMSE is small with a lower prediction horizon. On the other hand, it was observed that the prediction performance decreased as the prediction horizon lengthened. This is consistent with the fundamental laws of prediction, as longer horizons introduce more uncertainty and make it more challenging to forecast accurately. The computational efficiency varies depending on the duration of the operating cycles. However, on average, the proposed FS–ARIMA demonstrated an overall efficiency of around 13% of the total time. This suggests that the model was efficient in terms of the computational resources required for prediction. To summarize, the proposed FS–ARIMA demonstrated a good ability to capture the varying characteristics of V–G within the prediction horizon, thanks to its flexible structure.

Table 2. ARMSE of the FS–ARIMA on V–G prediction.

Accuracy		**ARMSE(m/s)**				**Time**	**Variance**
		4 s	**6 s**	**8 s**	**10 s**	**(s)**	**(m/s)**
V	Actual 1	0.08	0.13	0.28	0.51	478	0.17
	Actual 2	0.02	0.07	0.18	0.38	252	0.09
	Typical	0.27	0.53	0.85	1.21	506	0.54
G	Actual 1	0.05	0.13	0.27	0.51	412	0.17
	Actual 2	0.06	0.12	0.32	0.65	207	0.11

4.2. Overall Evaluation

In order to highlight the performance of the FS–ARIMA, additional methods have been included for comparison. These methods are the modified online velocity prediction method ARBF-NN [27], conventional RBF-NN, and LSTM. For all the methods, the prediction horizon was set to 10 s. The dominant structures of ARBF-NN and RBF-NN were specified as 50 and 30, respectively. It is important to note that ARBF-NN is an adaptive RBF-NN with flexible structural parameters. In the case of LSTM, there were 1486 samples available for offline training, and 136 test samples were used. As a benchmark, conventional ARIMA (2, 2, 1) was set and its performance was represented as 100% normalized ARMSE. The comparison results under different prediction horizons are listed in Table 3. The ARMSE indicates the average accuracy of the three velocity cycles and two actual road gradient cycles.

Table 3. The comparison of the FS−ARIMA for the whole cycles of V–G prediction via different approaches.

	Horizon	ARMSE (m/s) 4 s	6 s	8 s	10 s	Accuracy (%)	Improvement (%)	Time (s)	Variance (m/s)
V	FS–ARIMA	0.12	0.24	0.44	0.70	58.37	41.63	412	0.27
	ARIMA	0.28	0.49	0.76	0.96	100	X	378	0.56
	ARBF-NN	0.48	0.61	0.93	1.06	−124.63	−24.63	325	0.68
	RBF-NN	0.79	0.99	1.18	1.36	−173.49	−73.49	345	0.91
	LSTM	0.55	1.11	1.25	1.57	−179.91	−79.91	435	0.92
G	FS–ARIMA	0.04	0.09	0.17	0.33	57.81	42.19	370	0.14
	ARIMA	0.11	0.21	0.32	0.45	100	X	354	0.28
	ARBF-NN	0.12	0.16	0.33	0.45	97.248	2.752	361	0.24
	RBF-NN	0.18	0.25	0.49	0.65	−144.03	−44.03	325	0.37
	LSTM	0.16	0.32	0.51	0.78	−162.38	−62.38	445	0.42

The results demonstrate that FS−ARIMA achieved an excellent prediction accuracy while maintaining an acceptable computational efficiency, for both velocity and road gradient prediction. In general, ARIMA−based prediction has an identical performance to AI-based and stochastic-based methods. For the velocity prediction, the ARMSE of FS–ARIMA is 0.12 m/s when the prediction horizon is 4 s, outperforming the ARIMA of 0.28 m/s, the ARBF−NN of 0.48 m/s, the RBF−NN of 0.79 m/s, and the LSTM of 0.55 m/s. The flexible structure of the proposed FS−ARIMA has a significant impact on the prediction accuracy, especially for shorter prediction horizons. As the prediction horizon increases, the accuracy of all the methods tends to diminish. However, FS−ARIMA showed a remarkable improvement compared to the other methods. In the prediction horizon of 10 s, the ARMSE of FS−ARIMA was 0.70 m/s, the ARIMA was 0.96 m/s, the ARBF-NN was 1.06 m/s, the RBF−NN was 1.36 m/s, and the LSTM was 1.57 m/s. According to the results, ARBF−NN was an improvement compared with RBF−NN, and this is consistent with the FS−ARIMA. That is because, with the variable differencing order and lags of models, the prediction is also noteworthy, despite a slight increase in runtime. The average prediction is highly improved when the data fluctuate. The improvement in road gradient runtime increases from 378 to 412 due to the online determination of the differencing order and lags of the model, representing a 9.6% increase, which is still acceptable.

Overall, the average prediction performance of FS−ARIMA improved by 41.63% for velocity and 42.19% for road gradient series compared with conventional ARIMA. Furthermore, FS−ARIMA demonstrated the smallest variance among the methods. In summary, FS−ARIMA significantly improved prediction accuracy while maintaining acceptable computational efficiency, particularly when dealing with data fluctuations. It surpassed conventional ARIMA as well as other AI-based and stochastic-based methods.

5. Conclusions and Future Work

5.1. Conclusions

The accuracy of V–G prediction is crucial for improving the performance of PEMS in EVs. A conventional ARIMA model with fixed structural parameters may not always be suitable for online prediction when dealing with data fluctuations. To address this limitation, a novel FS–ARIMA model was developed, incorporating variable differencing orders and lags to enhance V–G prediction accuracy. The sliding window method was utilized to produce the V–G time series in real time. By continuously updating the series, the impact of differencing orders and lags on prediction accuracy was thoroughly investigated. It was observed that the fixed structure of the conventional ARIMA model lacked adaptability in online prediction. By introducing a differencing order and lags of the model determination method with the ADF test and the BIC, FS–ARIMA was designed to further improve the prediction accuracy. The effectiveness of the proposed FS–ARIMA model was validated through simulations using actual and typical driving cycles. The results demonstrated an approximate improvement of 41.63% and 42.19% in V–G series prediction accuracy, respectively. Furthermore, the FS–ARIMA model did not require extensive historical or numerous typical databases for offline training, making it useful for early-stage ITS applications related to PEMS in EVs.

5.2. Future Work

The work reported in this paper is only one step toward the development of an FS–ARIMA with a flexible structure for online V–G predictions. It can improve prediction accuracy. In future work, we will carry out research in the following areas: (1) Balance the structure to extend ARIMA applications, such as velocity and trajectory prediction for self-driving. (2) Consider the variables of sample size and prediction horizon to further balance the prediction accuracy and the calculation time. (3) Develop a more advanced optimization algorithm to obtain the optimal structure and improve the computational efficiency.

Author Contributions: Conceptualization, B.M. and P.L.; methodology, B.M. and X.G.; software, B.M.; validation, B.M. and X.G.; formal analysis, B.M. and P.L.; writing—original draft preparation, B.M. and X.G.; writing—review and editing, B.M., P.L., H.Z. and Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of Beijing (3212005), the National Natural Science Foundation of China (51608040), an Open Fund Project of the Key Laboratory of the Transportation Industry in Transportation Vehicle Operation Safety Technology (KFKT2022-06), a Pilot Project of the Ministry of Transport's Highway Science Research Institute for Strengthening Transportation (2021-321(QG2021-4-19-2)), and a Science and Technology Research Project of Beijing Shanghai High Speed Railway Co., Ltd. (JingHu Scientific Research- 2022-14).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Guo, N.; Zhang, X.; Zou, Y.; Du, G.; Wang, C.; Guo, L. Predictive Energy Management of Plug-in Hybrid Electric Vehicles by Real-Time Optimization and Data-Driven Calibration. *IEEE Trans. Veh. Technol.* **2022**, *71*, 5677–5691. [CrossRef]
2. Liu, T.; Tan, W.; Tang, X.; Zhang, J.; Xing, Y.; Cao, D. Driving Conditions-Driven Energy Management Strategies for Hybrid Electric Vehicles: A Review. *Renew. Sustain. Energy Rev.* **2021**, *151*, 111521. [CrossRef]
3. Zhou, Y.; Ravey, A.; Péra, M.-C. A Survey on Driving Prediction Techniques for Predictive Energy Management of Plug-in Hybrid Electric Vehicles. *J. Power Sources* **2019**, *412*, 480–495. [CrossRef]
4. Sun, C.; Hu, X.; Moura, S.J.; Sun, G. Velocity Predictors for Predictive Energy Management in Hybrid Electric Vehicles. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 1197–1204. [CrossRef]
5. Guo, J.; He, H.; Sun, C. ARIMA-Based Road Gradient and Vehicle Velocity Prediction for Hybrid Electric Vehicle Energy Management. *IEEE Trans. Veh. Technol.* **2019**, *68*, 5309–5320. [CrossRef]
6. Liu, T.; Hu, X.; Li, S.E.; Cao, D. Reinforcement Learning Optimized Look-Ahead Energy Management of a Parallel Hybrid Electric Vehicle. *IEEEASME Trans. Mechatron.* **2017**, *22*, 1497–1507. [CrossRef]

7. Pei, J.; Su, Y.; Zhang, D.; Qi, Y.; Leng, Z. Velocity Forecasts Using a Combined Deep Learning Model in Hybrid Electric Vehicles with V2V and V2I Communication. *Sci. China Technol. Sci.* **2020**, *63*, 55–64. [CrossRef]
8. Hassanzadeh, M.; Rahmani, Z. A Predictive Controller for Real-Time Energy Management of Plug-in Hybrid Electric Vehicles. *Energy* **2022**, *249*, 123663. [CrossRef]
9. Liu, Y.; Li, J.; Gao, J.; Lei, Z.; Zhang, Y.; Chen, Z. Prediction of Vehicle Driving Conditions with Incorporation of Stochastic Forecasting and Machine Learning and a Case Study in Energy Management of Plug-in Hybrid Electric Vehicles. *Mech. Syst. Signal Process.* **2021**, *158*, 107765. [CrossRef]
10. Hou, J.; Yao, D.; Wu, F.; Shen, J.; Chao, X. Online Vehicle Velocity Prediction Using an Adaptive Radial Basis Function Neural Network. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3113–3122. [CrossRef]
11. Jiang, B.; Fei, Y. Vehicle Speed Prediction by Two-Level Data Driven Models in Vehicular Networks. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1793–1801. [CrossRef]
12. Lin, X.; Zhang, G.; Wei, S. Velocity Prediction Using Markov Chain Combined with Driving Pattern Recognition and Applied to Dual-Motor Electric Vehicle Energy Consumption Evaluation. *Appl. Soft Comput.* **2021**, *101*, 106998. [CrossRef]
13. Topić, J.; Škugor, B.; Deur, J. Receding-Horizon Prediction of Vehicle Velocity Profile Using Deterministic and Stochastic Deep Neural Network Models. *Sustainability* **2022**, *14*, 10674. [CrossRef]
14. Aasim; Singh, S.N.; Mohapatra, A. Repeated Wavelet Transform Based ARIMA Model for Very Short-Term Wind Speed Forecasting. *Renew. Energy* **2019**, *136*, 758–768. [CrossRef]
15. Sheoran, S.; Pasari, S. Efficacy and Application of the Window-Sliding ARIMA for Daily and Weekly Wind Speed Forecasting. *J. Renew. Sustain. Energy* **2022**, *14*, 053305. [CrossRef]
16. Shah, I.; Muhammad, I.; Ali, S.; Ahmed, S.; Almazah, M.M.A.; Al-Rezami, A.Y. Forecasting Day-Ahead Traffic Flow Using Functional Time Series Approach. *Mathematics* **2022**, *10*, 4279. [CrossRef]
17. Rajeevan, A.K.; Shouri, P.V.; Nair, U. ARIMA Based Wind Speed Modeling for Wind Farm Reliability Analysis and Cost Estimation. *J. Electr. Eng. Technol.* **2016**, *11*, 869–877. [CrossRef]
18. Fan, D.; Sun, H.; Yao, J.; Zhang, K.; Yan, X.; Sun, Z. Well Production Forecasting Based on ARIMA-LSTM Model Considering Manual Operations. *Energy* **2021**, *220*, 119708. [CrossRef]
19. Wang, Z.; Qu, J.; Fang, X.; Li, H.; Zhong, T.; Ren, H. Prediction of Early Stabilization Time of Electrolytic Capacitor Based on ARIMA-Bi_LSTM Hybrid Model. *Neurocomputing* **2020**, *403*, 63–79. [CrossRef]
20. Zhang, Y.; Chen, Z.; Li, G.; Liu, Y.; Huang, Y.; Cunningham, G.; Early, J. Integrated Velocity Prediction Method and Application in Vehicle-Environment Cooperative Control Based on Internet of Vehicles. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2639–2654. [CrossRef]
21. Villalba, A.; Berral, J.L.; Carrera, D. Constant-Time Sliding Window Framework with Reduced Memory Footprint and Efficient Bulk Evictions. *IEEE Trans. Parallel Distrib. Syst.* **2019**, *30*, 486–500. [CrossRef]
22. Liu, X.; Lin, Z.; Feng, Z. Short-Term Offshore Wind Speed Forecast by Seasonal ARIMA—A Comparison against GRU and LSTM. *Energy* **2021**, *227*, 120492. [CrossRef]
23. Lim, S.; Kim, S.J.; Park, Y.; Kwon, N. A Deep Learning-Based Time Series Model with Missing Value Handling Techniques to Predict Various Types of Liquid Cargo Traffic. *Expert Syst. Appl.* **2021**, *184*, 115532. [CrossRef]
24. Guo, J.; Chen, H.; Zhang, J.; Chen, S. Structure Parameter Optimized Kernel Based Online Prediction with a Generalized Optimization Strategy for Nonstationary Time Series. *IEEE Trans. Signal Process.* **2022**, *70*, 2698–2712. [CrossRef]
25. Wu, J.; Zhang, J.; Tan, W.; Lan, H.; Zhang, S.; Xiao, K.; Wang, L.; Lin, H.; Sun, G.; Guo, P. Application of Time Serial Model in Water Quality Predicting. *Comput. Mater. Contin.* **2023**, *74*, 67–82. [CrossRef]
26. Li, Y.; Su, Y.; Shu, L. An ARMAX Model for Forecasting the Power Output of a Grid Connected Photovoltaic System. *Renew. Energy* **2014**, *66*, 78–89. [CrossRef]
27. Zhang, S.; Xiong, R. Adaptive Energy Management of a Plug-in Hybrid Electric Vehicle Based on Driving Pattern Recognition and Dynamic Programming. *Appl. Energy* **2015**, *155*, 68–78. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.