



**HAL**  
open science

# Efficient Traffic Flow Guidance Feedback Strategy Considering Drivers' Disobedience in ITS

Zhengyan Hu, Nacima Labadie, Lyes Khoukhi

► **To cite this version:**

Zhengyan Hu, Nacima Labadie, Lyes Khoukhi. Efficient Traffic Flow Guidance Feedback Strategy Considering Drivers' Disobedience in ITS. Applied Sciences, 2023, 13 (9), pp.5788. 10.3390/app13095788 . hal-04429705

**HAL Id: hal-04429705**

**<https://utt.hal.science/hal-04429705v1>**

Submitted on 29 May 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.


L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

## Article

# Efficient Traffic Flow Guidance Feedback Strategy Considering Drivers' Disobedience in ITS

Zhengyan Hu <sup>1</sup>, Nacima Labadie <sup>1</sup> and Lyes Khoukhi <sup>2,\*</sup> 

<sup>1</sup> Computer Science and Digital Society (LIST3N) Laboratory, University of Technology of Troyes (UTT), 10420 Troyes, France

<sup>2</sup> UNICAEN, ENSICAEN, Normandie University, NRS, GREYC, 14000 Caen, France

\* Correspondence: lyes.khoukhi@ensicaen.fr

**Abstract:** Some research claims that the cultural features of people may have a major impact on driving behavior and can play a serious role in the driving safety. Indeed, unlike machines that follow the instructions strictly, drivers have their own will and may disobey; they sometimes do things completely opposite to the instructions given by the central vehicle system. On the other side, it is proven that the traffic flow guidance feedback strategy has a significant effect on improving road traffic flow capacity. This paper deals with traffic flow dynamics while real-time information is integrated. Drivers' obedience rate to the instructions provided by an intelligent transportation system (ITS) is also considered. In addition, we implemented the process of pre-filtering the drivers for better managing the traffic, to the best of our knowledge; thus this process is not considered in the related literature. We introduced a dynamic weighted vehicle density feedback strategy (DWVDFS) with SUMO software 19.386-16 for determining whether traffic congestion happens or not. The simulation model considers three-route scenarios with one entrance and one exit. The added value of the proposed pre-filtering process and DWVDFS is also evaluated.

**Keywords:** intelligent transportation system (ITS); traffic guidance; drivers' obedience and driving behavior



**Citation:** Hu, Z.; Labadie, N.; Khoukhi, L. Efficient Traffic Flow Guidance Feedback Strategy Considering Drivers' Disobedience in ITS. *Appl. Sci.* **2023**, *13*, 5788. <https://doi.org/10.3390/app13095788>

Academic Editors: Juan Francisco De Paz Santana, Vicente Julian Inglada and Habib Hamam

Received: 17 February 2023

Revised: 20 April 2023

Accepted: 5 May 2023

Published: 8 May 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Intelligent transportation systems (ITS) aim to provide efficient and safe services relating to different modes of transport and traffic management and to enable users to be better informed. Moreover, the integration of existing technologies can create new services.

Driving behavior is an important factor in ITS. Studies have shown that the cultural and educational backgrounds of people can have a major impact on driving behavior and can play an important role in driving safety in ITS. For instance, there are studies on Chinese drivers' behavior showing that aggressive driving makes a significant contribution to traffic accidents, and statistics show that 84.6% of traffic accidents are caused by driving rules violations [1]. Usually, the driving behavior in ITS is also impacted by the traffic control strategy.

To alleviate traffic control issues (e.g., congestion) and enhance the road capacity of existing infrastructure, researchers and practitioners are attracted by vehicular traffic flow problems; many theories have been proposed. For example, a mean velocity route guidance strategy was proposed by Lee et al. [2], in which the largest mean velocity as route preference is considered. Another strategy, the congestion coefficient route guidance strategy, was proposed by Wang et al. [3]. This latter strategy proved to be better than the travel time route guidance strategy and mean velocity route guidance strategy used in [1]. The congestion coefficient route guidance strategy was then popularly employed in [3–6]. Other strategies were designed, which one can cite, such as the pace flow route guidance strategy proposed by Chen et al. [7] and the vacancy length route guidance strategy, which offers easy operational applications. This last strategy was also suggested

by Chen et al. and is shown to be better than other strategies (e.g., travel time route guidance strategy, congestion coefficient route guidance strategy, space flow route guidance strategy, and time flow route guidance strategy [6]). A strategy based on the number of vehicles was introduced by Tang et al. [8], and its performance was assessed in terms of network efficiency and cost.

Only a few studies have focused on studying drivers' behaviors. For instance, some studies, such as [9–11], have shown that there is no difference in driving behavior between males and females; however, some disparities persist in certain aspects. The differences involve traffic accidents and offenses, although driving times, attitudes, education, and other background factors are controlled. In real-life conditions, the drivers, unlike machines that follow the instructions strictly, have their own wills; it is possible that they will disobey or sometimes do things completely opposite to the instructions given by the central vehicle system. Moreover, the uncertainty of drivers' obedience not only hinders the efficiency of the feedback strategy but also leads to a misjudgment of the system.

Thus, the originality of our work resides in the integration of the human factor, which most previous works did not consider. This factor is the obedience rate of drivers. For this purpose, we propose the use of a pre-filtering procedure before giving instructions to drivers. That is to say, the system classifies the drivers into two categories: those who tend to follow the instructions and those who tend not to, and then chooses to give instructions to the drivers who are more likely to obey the system.

The contributions of this study are listed hereafter:

- (1) A procedure based on machine learning algorithms, namely the  $k$ -means and SVM methods, is used to classify drivers according to their behavior.
- (2) A new traffic flow guidance algorithm, the dynamic weighted vehicle density feedback strategy (DWVDFS), is proposed. This latter is based on WVDFS [12] suggested by Dong and Ma. In WVDFS, the vehicle velocity is not taken into account when calculating the vehicle density. In this work, vehicle velocity is considered as an important factor for determining whether traffic congestion happens or not. The adapted DWVDFS therefore takes into account the vehicle velocity when calculating the vehicle density.
- (3) A performance evaluation, to assess the added value of both DWVDFS and the pre-filtering technique, is carried out within different configurations. Our solution has been validated through a series of experiments.

The rest of the paper is organized as follows. In Section 2, the SVM algorithm is introduced, along with the traffic flow guidance algorithm WVDFS, which is depicted in more detail. The new algorithms, DWVDFS and SVM, are presented in Section 3. The simulation results and performance analysis are shown in Section 4. Finally, a conclusion of the work is presented in Section 5.

## 2. Related Literature

In this section, we summarize the studies related to our work as follows: The first part is dedicated to the contributions considering driving behaviors; the second one focuses on the weighted vehicle density feedback strategy proposed in [12].

### (A) Driving behavior

Some studies claim that culture and people's traditions have a major impact on how drivers control their cars and driving safety. Studies on Chinese drivers' behavior have shown that aggressive driving contributes significantly to traffic accidents, i.e., around 84.6% of car accidents are caused by traffic violations [1]. This latter study proved that bad driving habits, a meager understanding of other road users, and a lack of respect for traffic regulations are reasons that contribute to traffic accidents.

Some studies pointed out that a system considered useful in one country can be seen as almost worthless (or even harmful) in another, and system settings feasible in one part of the world may not be suitable on the other side of the world [6].

Although some works dealing with driver behavior were published recently, there has been little attempt to understand and take into account driver behaviors and goals when designing products for a global ITS. We strongly believe that the key to an effective design is an understanding of how needs/requirements vary among several drivers. This may be particularly appropriate for ADAS (advanced driving assistance systems) [6] since not only the rules of the road but also social situations, norms, and driver behavior may differ considerably from state to state in terms of the behaviors of drivers (various cultures of driving). The authors in [13] conducted an experiment to determine anxiety status and driving behavior based on heart rate and concluded that anxiety is accompanied by increased heart rate and cautious driving.

### (B) Traffic Flow Guidance Feedback Strategy WVDFS

In this sub-section, we survey the weighted vehicle density feedback strategy proposed in [12]. At each time, vehicles on the road upload their positions to the traffic control center. Based on the uploaded information, the traffic control center could compute the weighted vehicle density of each route and display it on the board at the entrance of the route. The newly arrived dynamic vehicle would choose the route with a smaller weighted vehicle density. The weighted vehicle density is defined as:

$$\rho_w = \sum_{i=1}^N \frac{(k \times \frac{n_i}{L} + b)}{L} \quad (1)$$

where  $L$  is the length of each road,  $N$  is the total number of vehicles on the route at that moment,  $n_i$  is the position of  $i$ th vehicle,  $k$  is the weight factor and  $b$  is a constant, where  $k = -2.0$  and  $b = 2.0$ .

Our paper only considers the stationary density of a vehicle on the route; it does not reflect the real congestion situation on the routes. For example, take a scenario on the expressway where vehicles are crowded and are close to each other, but they are all moving at a high speed. In this scenario, this route cannot be defined as a route with severe congestion (as a driver drives more carefully when there are many vehicles around her/him). Applying the WVDFS strategy in this context may result in severe congestion. This might lead to a misunderstanding of the system and the actual situation in the driving environment.

## 3. Proposed Approach

In this section, we introduce our proposed SVM algorithm as well as the new traffic flow guidance algorithm, the dynamic weighted vehicle density feedback strategy (DWVDFS). First, we present the assumptions related to the classification setup, the input feature vector, and the input datasets. Then, we present the SVM kernel selection and the flow guidance feedback strategy.

### 3.1. Classification Setup

In this work, the driver's classification mainly focuses on drivers' driving habits and styles. To pre-filter the drivers regarding whether they will obey an instruction or not, an obedience feature vector is proposed. Many factors can be taken into account:

- The distance from home: Usually, drivers tend to follow the instructions of the vehicular system when they are not familiar with the routes. However, when they know the routes very well, there is a higher possibility for them to disobey the instructions. Thus, the distance between the vehicle's current location and the driver's home (or daily activity radius) is one of the factors that determines the driver's next action.
- The age of the driver: Generally, a driver who is older tends to know more routes than young drivers (i.e., the drivers usually become more stubborn when they are young). We may conclude that the drivers who are older may have a higher possibility of disobeying the driving instructions.

- The gender of the driver: Male drivers are usually more confident about themselves; sometimes this kind of confidence is closer to arrogance, which leads to the disregard of the driving instructions.
- The ratio of current velocity/max velocity of the route: Generally, a driver drives faster when he is in a hurry, which leads to a higher ratio. The ratio thus mainly represents the mood of the driver. If the driver is in a hurry, we believe he will tend to follow the instructions.
- The rate of obedience at past times: Although the four factors above have certain influences when the driver makes a decision, the driver's obedience rate is always important. This rate reflects the driver's habits to some extent. If the driver has tended to follow the driving instructions in the past, he will tend to follow the instructions in the near future.

Thus, the input vector is shown as follows:

ObedienceFeatureVect: {HistoricalObedienceRate, Distance, Age, Gender, VelocityRatio}

After describing the driver's classification, we discuss the proposed approach.

### 3.2. Labeling of Input Datasets

Our first contribution is based on the support vector machine (SVM) algorithm. SVM is a supervised learning algorithm that can be used for classification and regression issues. Although a feature vector is proposed, the conditions of this research did not provide an opportunity to collect data from drivers in real-life scenarios, and it also limits the possibility of operating a real test on whether drivers will choose to follow or disobey the instructions in a real driving environment. The obedience feature vector (or datasets) is thus generated randomly during the simulation. A supervised learning algorithm in this context means that the training datasets should be labeled. A labeling procedure should thus be added between the processes of dataset generation and SVM training. The data mining techniques, including clustering methods [14,15], have shown their powerful capability to elaborate and analyze the similarities and differences between the driving behavior characteristics regarding longitudinal driving behavior. The  $k$ -means clustering with  $k$  being 2 is chosen to be applied in this research.

$k$ -means is a classical unsupervised learning model. The goal of this iterative algorithm is to minimize intra-class inertia. Its main steps are briefly summarized hereafter:

- (1) Choose the  $K$  centers of  $K$  classes  $m_k^{(0)}$ ;  $t = 0$
- (2) Given the centers  $m_k^{(t)}$  ( $k = 1 \dots K$ ) by selecting the closest center, assign all the points (input vector) to class  $S_k^{(t)}$
- (3) Given the points in classes  $S_k^{(t)}$  ( $k = 1 \dots K$ ), calculate the new centers  $m_k^{(t+1)}$

$$m_k^{(t+1)} = \frac{1}{|S_k^{(t)}|} \sum_{x_j \in S_k^{(t)}} x_j \quad (2)$$

- (4) Repeat steps (2) and (3) until convergence:  $|m_k^{(t+1)} - m_k^{(t)}| < \epsilon, \forall k, \epsilon > 0$

In our work, the datasets could be clustered by using the  $k$ -means clustering methods. For example, points with low historical obedience rates (e.g.,  $R_o = 0.3$ ) and little distance (e.g.,  $d \leq 10\text{km}$ ) could be labeled as disrespect to the instruction. Based on the pre-labeled centers, all data could be manually labeled by  $y = \{-1, 1\}$ , where  $-1$  and  $1$  represent the disobedience and obedience of the driver, respectively.

### 3.3. SVM Kernel Selection

In this sub-section, we explain our method for SVM kernel selection. The most commonly applied kernels are the linear kernel, polynomial kernel, radial basis function (RBF) kernel, sigmoid kernel, and intersection kernel. In [16,17], a typical procedure was suggested by Ben-Hur and Weston. The procedure consists of trying a linear kernel first

and then checking if the performance is better when using a non-linear kernel (e.g., a polynomial kernel or RBF). In [14], an approach based on the RBF kernel showed better accuracy and convergence time compared to the polynomial kernel. Consequently, we have chosen to first try the linear kernel and then the RBF kernel.

In [18], a grid search for the support vector machine was carried out, and the accuracy was considered to evaluate the performance of different models. In fact, for the RBF kernel, besides the hyper-parameter  $\gamma$ , the selection of another parameter, the penalty coefficient  $C$ , is also important, as SVM adjusts the trade-off between the margin and fitting error by modifying the parameter  $C$ . Thus, the model needs to be tuned by applying several tests in order to avoid overfitting or underfitting as much as possible. A search of combinations for the values of the  $C$  and  $\gamma$  parameters was performed. Indeed, in this study, the RBF kernel, where  $\gamma = \frac{1}{\text{sample nbr.}}$ , outperforms in SVM training process tests; the penalty coefficient parameter  $C$  was set to 0.1. This combination showed an accuracy of 0.9905 in the tests.

### 3.4. New Traffic Flow Guidance Feedback Strategy DWVDFS

In the basic model of WVDFS (weighted vector directional filters), the vehicle velocity is not considered during the calculation of the vehicle density. However, in our solution, and as mentioned in Section 1, the vehicle velocity is determined to be one of the factors that has an impact on the vehicle density. A dynamic weighted vehicle density strategy (DWVDFS) is thus proposed. Based on the uploaded data, the intelligent driving system could compute the dynamic weighted vehicle density of the route, taking into account a threshold set in advance for the vehicle density. Indeed, if the vehicle density surpasses the threshold, the route is considered congested, and instructions need to be given to drivers. The newly arrived vehicles that are classified as obedient would be given the instruction to change to another route. In WVDFS, a weight factor  $k$  is applied in the calculation of the density. In DWVDFS, the factor  $k$  is replaced by a function containing the vehicles' velocities. The dynamic weighted vehicle density is defined as:

$$\rho_w^d = \sum_{i=1}^{N_w} \frac{(-\log(v_i + 1) \times \frac{n_i}{L_w} + b)}{L_w} \quad (3)$$

where  $\rho_w^d$  refers to the dynamic density,  $L_w$  is the length of route  $w$ ,  $N$  is the total number of vehicles on the route  $w$  at the current time step,  $n_i$  stands for the position of  $i$ th vehicle,  $v_i$  denotes the velocity of  $i$ th vehicle and  $b$  is a constant, where  $b = 5.0$ .

## 4. Simulation Results and Discussion

In this section, we start by introducing the experimental setup and then present our results in terms of performance evaluation.

### 4.1. Experimental Design

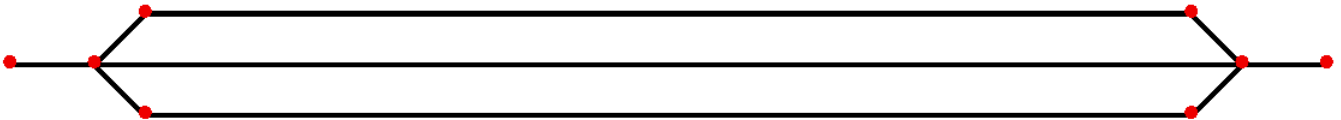
The traffic simulator is chosen to be SUMO, "simulation of urban mobility." The SVM and  $k$ -means are implemented in MATLAB. An open-source toolbox of MATLAB, LIBSVM [19], is used in our experimentation. Although MATLAB has its own function for implementing SVM, the functions in LIBSVM are considered more powerful than those in the MATLAB package. Thanks to TraCI4Matlab [20], which serves as an implementation of the Traffic Control Interface (TraCI) for MATLAB, an integration between MATLAB and SUMO is realized. TraCI4Matlab is an application programming interface (API) developed in MATLAB that allows communication between any application developed in this language and a microscopic road traffic simulator such as SUMO.



#### 4.2. Simulation Scenario and Performance Evaluation

In our simulation scenario, a route called the “original route” was first considered. We have chosen to make most of the arriving vehicles pass through the original route. The spare routes are also needed if the drivers are asked to change their route to avoid congestion.

In SUMO, if the entrance is congested by vehicles, newly arriving cars will not enter the scenario; therefore, in order to avoid such a situation caused by the congestion on other routes, the possibility of congestion on the other route must be limited at the lowest level. In the simulation, a three-route scenario was designed. The system will ask drivers to change to either spare route randomly. The three-route scenario is shown in Figure 1.



**Figure 1.** A three-route scenario.

The route in the middle is the original route, and the other two routes are the spare routes. A total of 80% of the vehicles entering the scenario are assumed to pass through the original route, while 10% of the vehicles are assumed to pass through each spare route. At the end of the scenario, the three routes share an intersection, and a traffic light is designed to create congestion on the original route.

The length of the original route is 680 m. The lengths of the spare routes are both 704.86 m. The traffic light is green for 30 s on the original route and for 40 s on both spare routes.

One vehicle will enter the scenario per second. The acceleration of each vehicle is  $1.2 \text{ m/s}^2$ . The deceleration of each vehicle is  $4.5 \text{ m/s}^2$ . The maximum speed for each vehicle on each route is 50 m/s (180 km/h). To ensure the diversity of driver behaviors when following other vehicles, a driver imperfection parameter is added with values taken in  $[0, 1]$ , where 0 denotes a perfect driver. The simulation starts at time step = 0 and ends at time step = 1799. The duration of the simulation is thus 1800 s in total.

In real life, it is impracticable for systems to collect road data and give instructions every second because of the limit on processor speed and data transfer speed; the surveillance period was set to 30 time steps (30 s) in the simulation. This means that the system collects the route/vehicle data and decides whether to give instructions or not every 30 s.

During each surveillance, the locations of vehicles on the route as well as their velocities are recorded, and the dynamic weighted vehicle density is calculated. A congestion threshold is used, just as mentioned in Section 3.4. If the dynamic weighted vehicle density surpasses the threshold, it is considered congestion, and the system will have to give instructions to the newly arriving drivers; otherwise, the system will not give instructions to drivers. In the simulation, several values of the threshold are tested, and in the end, the threshold value is set to 0.1. If the threshold is set too high, congestion will be detected too late.

If the system decides to instruct the newly arriving driver to change to a new route, it will choose one of the two spare routes randomly with equal probability  $w$ . This option will help to reduce the probability of congestion on the two spare routes.

The performance evaluation was carried every 5 time steps (5 s). During each evaluation, the mean vehicle speed and the number of vehicles that are not moving, were recorded. The flux is calculated, as in [21], with the following formula:

$$F = V_{mean} \times N/L \quad (4)$$

where  $V_{mean}$  denotes the mean velocity of all vehicles on the route,  $N$  stands for the number of vehicles on the route, and  $L$  is the length of the route. The flux indicates the mean number of vehicles that pass through the route per second.

Finally, a timer was added to record the duration of each congestion during the simulation. After the simulation, the mean congestion duration was also calculated.

#### 4.3. Simulation Results and Analysis

Firstly, a simulation in which no instructions were given was carried out, see Table 1. This simulation shows the route situation and whether congestion would disappear by itself if no interference was taken. Secondly, a comparison between giving instructions with/without the filtering of drivers by SVM was made.

**Table 1.** Congestion durations without instruction.

No Instruction
$8.075 \times 10^2$
$8.075 \times 10^2$
$8.075 \times 10^2$
$8.075 \times 10^2$
$8.075 \times 10^2$

In fact, if the goal is to reduce congestion on one route, the simplest way is to instruct all the newly arriving vehicles to change to other routes, but this is not efficient and effective since it will cause congestion on other routes. If new congestion appears on other routes, the system has to instruct all the newly arriving vehicles once again, which will not only extend the time for eliminating congestion but also confuse the drivers since contradictory instructions are given too often. Thus, since the surveillance takes place every 30 s and only one vehicle enters the scenario every second, it is sufficient to instruct 30 newly arriving vehicles to change routes during each surveillance.

If SVM is applied to pre-filter the drivers and choose those obeying the instructions, since the system needs to inform 30 drivers in each surveillance and drivers are separated into 2 groups, the system has to implement the SVM classification for the 60 newly arriving drivers each time. Around 30 drivers will therefore be chosen to change routes.

If SVM is not implemented and drivers are not pre-filtered, the system simply needs to instruct the next 30 newly arriving drivers. During the simulation, since real tests cannot be realized, two logics were proposed to simulate the driver's reaction when he or she receives instruction: (1) a combination of principal component analysis and *k*-means and (2) Fuzzy Logic.

The driver data is first collected, and *k*-means or Fuzzy Logic is then applied. If the result is 1, that means this driver will follow the instruction, and a command to change the route for the vehicle will be sent to SUMO; otherwise, no command is sent, and the vehicle will continue to pass through the original route.

The first simulation is the one with no instructions given by the system. The goal is to see what will happen on the road during congestion if no interference is made by the intelligent transportation system framework. At the beginning, the traffic lights created congestion on the road. The waiting queue became longer as vehicles kept entering the route. Although congestion was reduced a little bit when the traffic lights turned green, not all the vehicles in the waiting queue were able to exit the route before traffic lights turned red. The queue kept growing until the route was full of vehicles and the dynamic vehicle density reached its peak.

Figure 2a shows the mean velocity of the vehicles on the route. From the graphs of the simulation, we can see that if no interference is taken, the congestion will not be eliminated by itself. The mean velocity reaches its peak at first, but when congestion happens, it goes down. Because of zero interference, the congestion keeps getting worse, and only a few vehicles can pass through the route when the traffic light is green. The original route was full of vehicles when the simulation ended.



As depicted in Figure 2b, the number of stopped vehicles on the original route, contrary to the curve of the mean velocity of vehicles, has grown higher when the time step increases and reaches a peak because the whole route is filled with waiting vehicles. As explained previously, the vehicle flux is the mean number of vehicles that pass through the route per second; the higher the value, the better the route conditions. As displayed in Figure 2c, the vehicle flux decreases with the growing time steps.

Furthermore, after simulating the same set of data five times, the mean congestion duration was extremely high. The mean congestion lasts nearly half the time of the entire duration.

The next simulation corresponds to the case where the system uses SVM before sending instructions to drivers. The goal is to see what the road situation will be during congestion if the system uses SVM to pre-filter the drivers and then gives instructions to the filtered drivers.

At first, the traffic created congestion on the road. The waiting queue became longer as vehicles kept entering the route. When dynamic weighted vehicle density surpasses the threshold, the system launches SVM and then instructs the drivers labeled as “obeying instructions” to change routes. Some of the newly arriving vehicles will not enter the original route, and the number of vehicles exiting the route is greater than those entering the route; thus, the congestion was reduced or eliminated.

When the dynamic weighted vehicle density went down, the route was not regarded as congested. More vehicles would enter the original route, and the vehicle density would re-increase; the procedure of the pre-filtering of drivers and the delivery of instructions would be implemented once again.

Figure 3a shows the mean velocity of the vehicles on the route when SVM is implemented. We observe a cycle of the change in the mean velocity. Thanks to the driver pre-filtering and instruction delivery procedures, the mean velocity could be maintained at a high level, where the peak of the mean velocity is nearly 15 m/s.

In Figure 3b, we observe that the number of stopped vehicles on the original route increased multiple times because of the traffic light. Nevertheless, the peak number of stopping vehicles did not surpass 18, thanks to the instructions given by the ITS framework.

Figure 3c shows that the vehicle flux went up and down multiple times, just like the cycle shown in the figure of mean velocity.

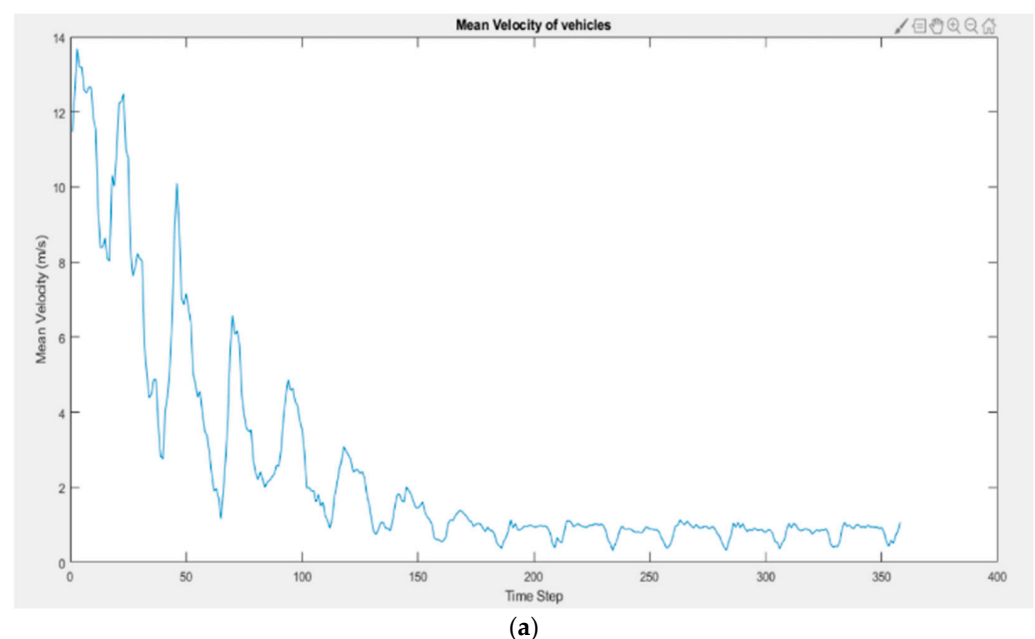
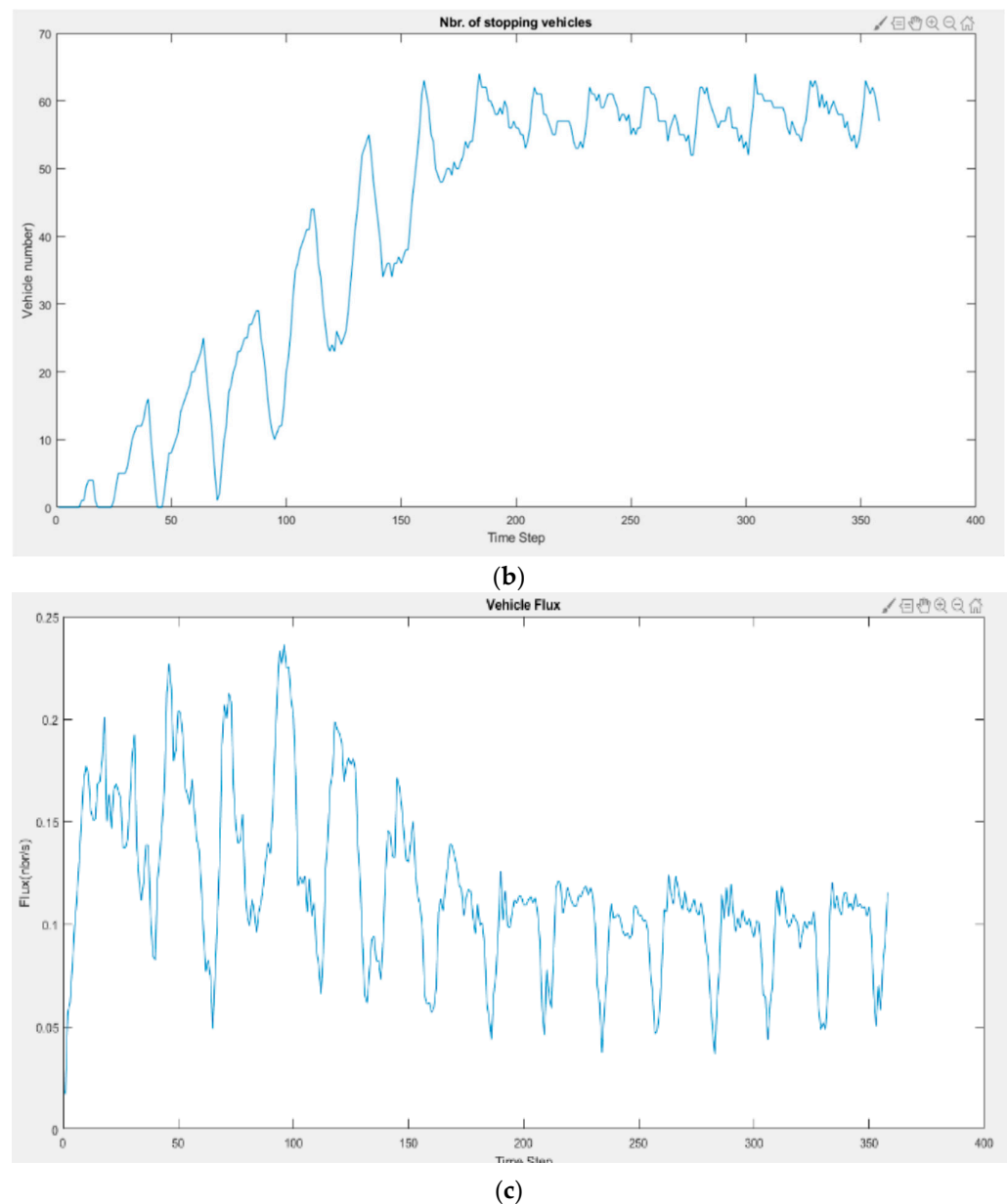


Figure 2. Cont.



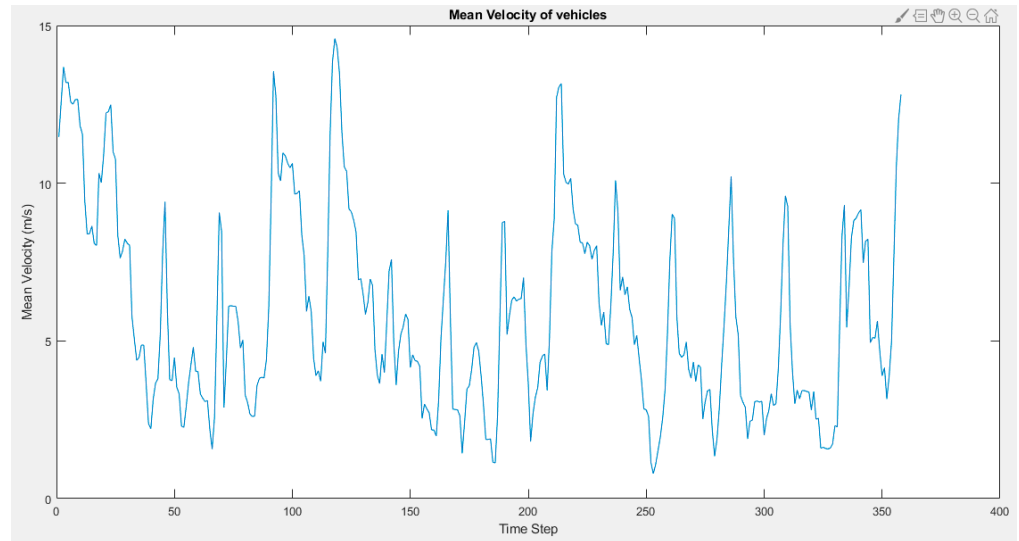
**Figure 2.** Results of no route guidance strategy. (a) Mean velocity without instruction; (b) Number of stopping vehicles; (c) Vehicle flux.

The third simulation is related to the case where instructions are given by the system without SVM. The goal is to see what will happen to the road situation during congestion if instructions are given to drivers directly by the system and no SVM or other pre-filtering procedures are implemented.

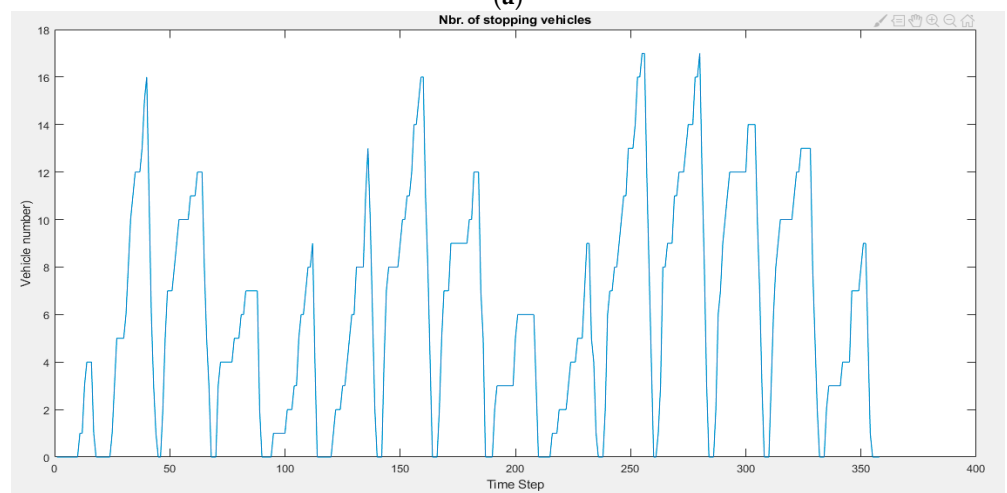
As explained before, during this study, it was not possible to perform a real test, and it is not easy to simulate what a driver will do in real life when he or she receives an instruction, so two methods to simulate the drivers' reactions are proposed. In this part, the simulation was realized by combining principal component analysis and  $k$ -means.

At first, the traffic lights created congestion on the road. The waiting queue became longer as vehicles kept entering the route. When dynamic weighted vehicle density surpasses the threshold, the system sends instructions to the newly arriving 30 drivers to change routes. Some of these drivers will follow the instruction, while others will not. This leads to the situation that some of the newly arriving vehicles will not enter the original route, and the number of vehicles exiting the route is greater than those entering the route; the congestion is hence reduced or eliminated. When the dynamic weighted vehicle den-

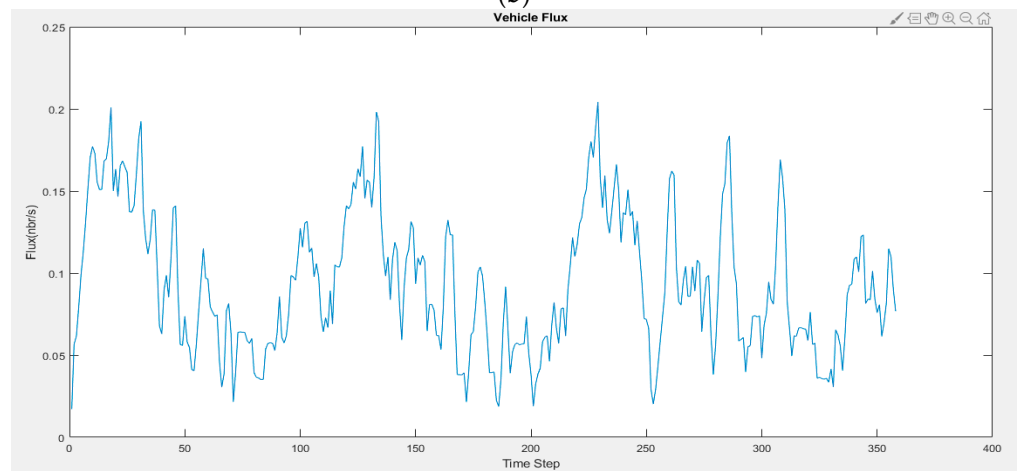
sity went down, the route was not regarded as congested; more vehicles would enter the original route, the vehicle density would re-increase, and the procedure of the delivery of instructions would be implemented once again.



(a)



(b)



(c)

**Figure 3.** Results of instruction with SVM. (a) Mean velocity without instruction; (b) Number of stopping vehicles; (c) Vehicle flux.

Figure 4a shows the mean velocity of the vehicles on the route when SVM is not implemented. We can observe a cycle of change in the mean velocity. Because of the instruction delivery procedure, the congestion would still be eliminated, but with less efficiency. The mean velocity could be maintained at a high level; the peak value of the mean velocity not using SVM was nearly 14 m/s, while it is a little bit lower when SVM is employed. We can also observe that for a short period of time, the mean velocity dropped to zero, which was not observed in the simulation with SVM.

As illustrated in Figure 4b, the number of stopped vehicles on the original route increased several times due to the traffic light but decreased right after; this is because of the instructions given by the system. However, the peak number of stopped vehicles reached 18, which is a little bit higher compared to the result of the simulation with SVM.

Figure 4c shows that the vehicle flux went up and down multiple times during the time steps, just like the cycle displayed in the figure of mean velocity. However, the values of flux are lower than those of the simulation with SVM.

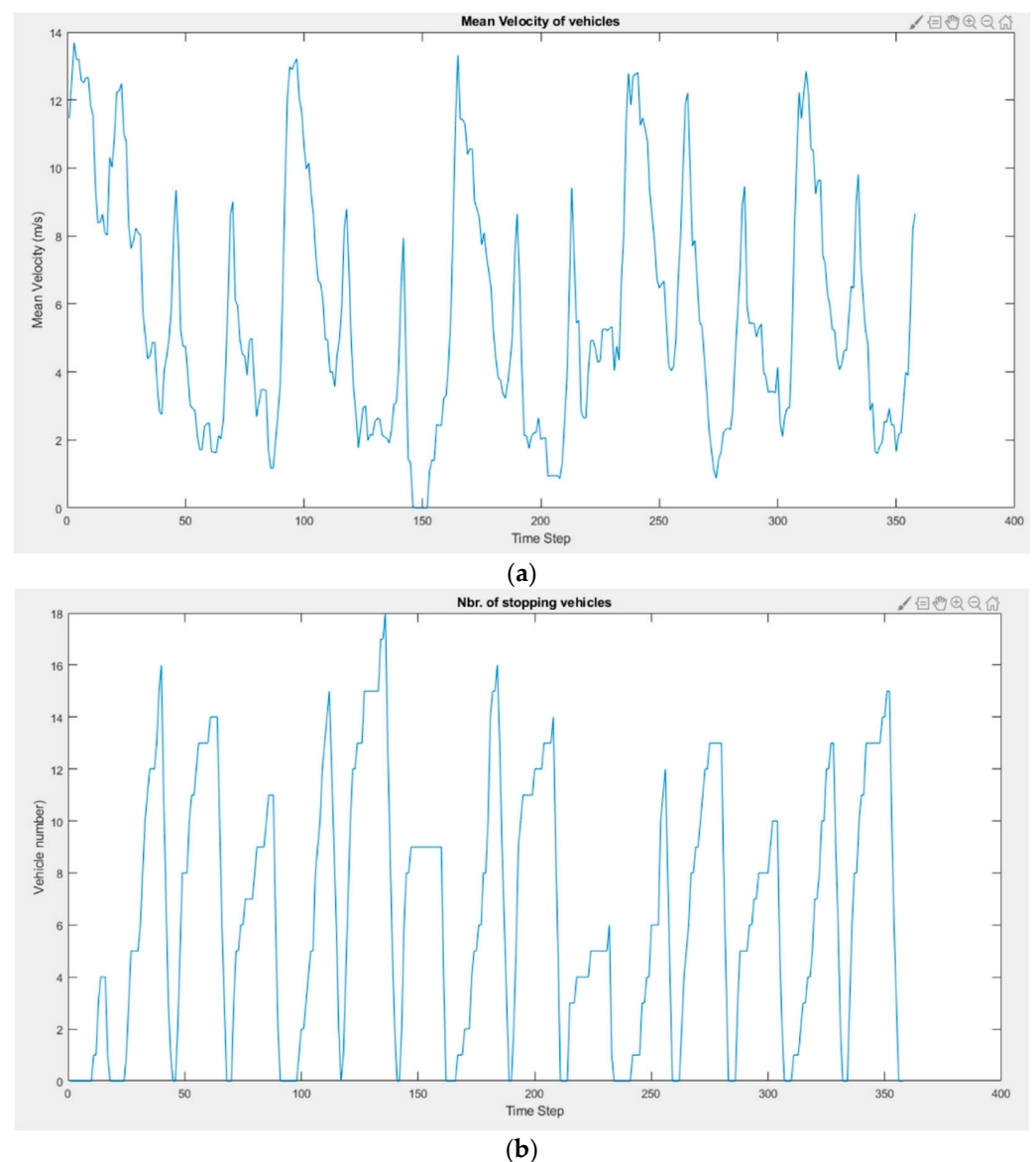
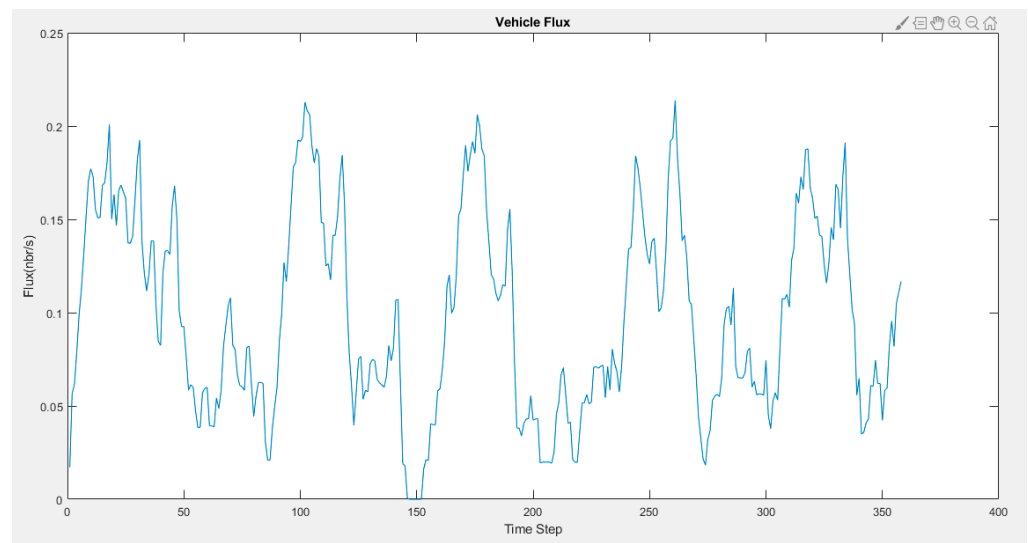


Figure 4. Cont.



(c)

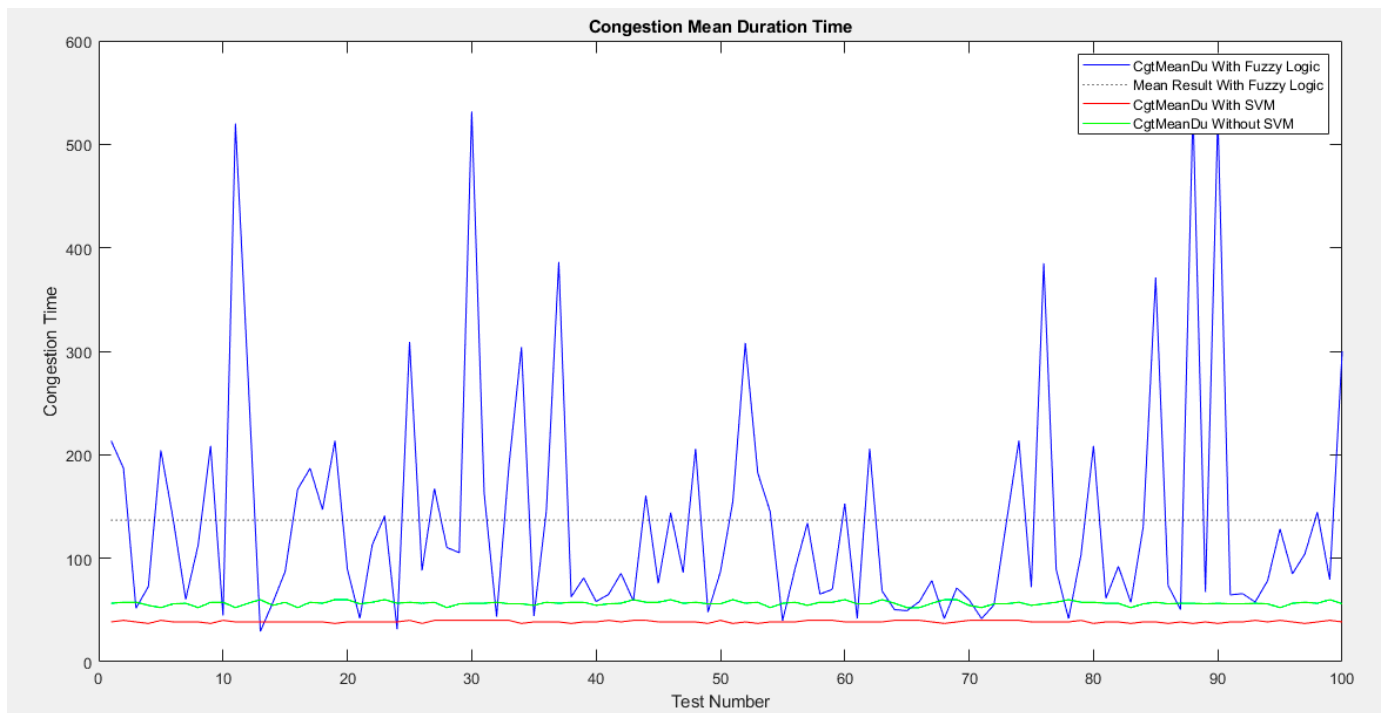
**Figure 4.** Results of instruction without SVM. (a) Mean velocity without instruction; (b) Number of stopping vehicles; (c) Vehicle flux.

#### Analysis of Different Scenes

In this sub-section, 100 durations of simulations are taken for each type of instruction (the three types discussed above) in order to compare the mean duration of congestions in different scenes. To ensure the accuracy and credibility of the comparison, the 300 durations of simulations all use the same set of driver data.

The three types of instructions are: (a) the instruction with pre-filtering using SVM; (b) the instruction without pre-filtering using *K*-means; and (c) the instruction without pre-filtering using Fuzzy Logic.

We can observe in Figure 5 that the mean duration for instruction without pre-filtering using Fuzzy Logic varies considerably, although it has a better performance than SVM several times. The mean duration is too unstable, and the mean value for the 100 durations is much greater than that of SVM. The mean duration for instructions without pre-filtering using *k*-means is more stable than that of Fuzzy Logic, but the mean duration value is still higher than with SVM. This result shows that although instructions without SVM can help reduce congestion, they are not as efficient as the instructions with SVM and *k*-means or Fuzzy Logic.



**Figure 5.** Congestion mean duration time.

## 5. Conclusions

Driving behavior is an important feature to consider in the next generation of ITS, as research has proven that the cultural and educational backgrounds of people can have a major impact on driving environments. In fact, drivers do not always obey the instructions provided by the vehicular systems, which may lead to a misjudgment of driving safety and hinder the efficiency of traffic management. In this paper, we propose a guidance approach that includes several metrics related to human behavior. A pre-filtering procedure based on SVM is thus proposed to identify the class of drivers that is more likely to follow the ITS instructions. A new route guidance algorithm, DWVDFS, was also designed based on WVDFS [12]. The simulation results showed that DWVDFS can alleviate congestion efficiently, whether a pre-filtering procedure is implemented or not. When comparing the configurations with and without pre-filtering, the results proved that the pre-filtering step leads not only to less time spent eliminating congestion but also to a more stable performance.

The aim of this work is to focus on an aspect rarely considered in the literature. The drivers' obedience to the ITS instructions will become a challenge in the future (unless all driving behaviors are accomplished by artificial intelligence in fully autonomous cars). The uncertainty of drivers' behavior should remain a major concern in all vehicular feedback systems.

**Author Contributions:** Conceptualization, Z.H., N.L. and L.K.; methodology, Z.H., N.L. and L.K.; validation, Z.H., N.L. and L.K.; writing—original draft preparation, Z.H., N.L. and L.K.; writing—review and editing, Z.H., N.L. and L.K.; supervision, N.L. and L.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** No concerned.

**Informed Consent Statement:** No concerned.

**Data Availability Statement:** No concerned.

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



## References

1. Chen, F.; Wang, M.; Duan, L. Letting Drivers Know What is Going on in Traffic. In *Advances in Intelligent Vehicles*; Academic Press: Cambridge, MA, USA, 2014; Chapter 10; pp. 291–318. [[CrossRef](#)]
2. Lee, K.E.; Hui, P.; Wang, B.H.; Johnson, N. Effects of announcing global information in a two route traffic flow model. *J. Phys. Soc. Jpn.* **2001**, *70*, 3507–3510. [[CrossRef](#)]
3. Wang, W.X.; Wang, B.H.; Zheng, W.C.; Yin, C.Y.; Zhou, T. Advanced information feedback in intelligent traffic systems. *Phys. Rev. E* **2005**, *72*, 066702. [[CrossRef](#)] [[PubMed](#)]
4. Chen, B.K.; Wang, Z.W.; Gao, Y.C.; Zhang, K.; Wang, B.H. Effects of traffic lights for Manhattan-like urban traffic network in intelligent transportation systems. *Transp. B Transp. Dyn.* **2018**, *6*, 4–16.
5. Wang, W.; Xi, J.; Chong, A.; Li, L. Driving Style Classification Using a Semisupervised Support Vector Machine. *IEEE Trans. Hum. Mach. Syst.* **2017**, *47*, 650–660. [[CrossRef](#)]
6. Togou, M.A.; Hafid, A.; Khoukhi, L. A Novel CDS-Based Routing Protocol for Vehicular Ad Hoc Networks in Urban Environments. In Proceedings of the 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, 6–10 December 2015; pp. 1–6. [[CrossRef](#)]
7. Chen, B.K.; Tong, W.; Zhang, W.Y.; Sun, X.Y.; Wang, B.H. Flux information feedback strategy in intelligent traffic systems. *Europhys. Lett.* **2012**, *97*, 1–6. [[CrossRef](#)]
8. Tang, T.Q.; Yu, Q.; Liu, K. Analysis of the traffic running cost in a two-route system with feedback strategy. *Phys. A Stat. Mech. Its Appl.* **2017**, *466*, 1–9. [[CrossRef](#)]
9. Togou, M.A.; Khoukhi, L.; Hafid, A. Performance Analysis and Enhancement of WAVE for V2V Non-Safety Applications. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2603–2614. [[CrossRef](#)]
10. Liu, Y.; Wang, X. The analysis of driver's behavioral tendency under different emotional states based on a Bayesian Network. *IEEE Trans. Affect. Comput.* **2023**, *14*, 165–177. [[CrossRef](#)]
11. Bichicchi, A.; Belaroussi, R.; Simone, A.; Vignali, V.; Lantieri, C.; Li, X. Analysis of Road-User Interaction by Extraction of Driver Behavior Features Using Deep Learning. *IEEE Access* **2020**, *8*, 19638–19645. [[CrossRef](#)]
12. Dong, C.F.; Ma, X. Dynamic weight in intelligent transportation systems: A comparison based on two exit scenarios. *Phys. A Stat. Mech. Its Appl.* **2012**, *391*, 2712–2719. [[CrossRef](#)]
13. Barnard, M.P.; Chapman, P. The effects of instruction and environmental demand on state anxiety, driving performance and autonomic activity: Are ego-threatening manipulations effective? *Transp. Res. Part F Traffic Psychol. Behav.* **2018**, *55*, 123–135. [[CrossRef](#)]
14. Higgs, B.; Abbas, M. Segmentation and clustering of car-following behavior: Recognition of driving patterns. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 81–90. [[CrossRef](#)]
15. Qi, G.; Du, Y.; Wu, J.; Xu, M. Leveraging longitudinal driving behaviour data with data mining techniques for driving style analysis. *IET Intell. Transp. Syst.* **2015**, *9*, 792–801. [[CrossRef](#)]
16. Ben-Hur, A.; Weston, J. A User's Guide to Support Vector Machines. In *Data Mining Techniques for the Life Sciences*; Springer: New York, NY, USA, 2010; pp. 223–239.
17. Han, R.; Wen, Y.; Bai, L.; Liu, J.; Choi, J. Age of information aware UAV deployment for intelligent transportation systems. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 2705–2715. [[CrossRef](#)]
18. Silva, I.; Eugenio Naranjo, J. A Systematic Methodology to Evaluate Prediction Models for Driving Style Classification. *Sensors* **2020**, *20*, 1692. [[CrossRef](#)] [[PubMed](#)]
19. Chang, C.-C.; Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27. [[CrossRef](#)]
20. Acosta, A.F.; Espinosa, J.; Espinosa, J. TraCI4matlab: Enabling the Integration of the SUMO Road Traffic Simulator and Matlab® Through a Software Re-engineering Process. In Proceedings of the Modeling Mobility with Open Data: 2nd SUMO Conference, Berlin, Germany, 15–16 May 2014; Springer: Berlin/Heidelberg, Germany, 2015; pp. 155–170.
21. Dong, C.F.; Ma, X.; Wang, G.W.; Sun, X.Y.; Wang, B.H. Prediction feedback in intelligent traffic systems. *Phys. A Stat. Mech. Its Appl.* **2009**, *388*, 4651–4657. [[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.