IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 November 2022, accepted 16 December 2022, date of publication 26 December 2022, date of current version 3 January 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3232554

RESEARCH ARTICLE

An Intelligent Evolving Car-Following Model

MAJID ABDOLLAHZADE [1] AND REZA KAZEMI [2]
[1]Department of Mechanical Engineering, Islamic Azad University, Pardis Branch, Pardis New City 1658174583, Iran
[2]Department of Mechanical Engineering, K. N. Toosi University of Technology, Tehran 1969764499, Iran

Corresponding author: Majid Abdollahzade (m.abdollahzade@pardisiau.ac.ir)

ABSTRACT The existing data-driven car-following models do not take structural and range variations of velocity and acceleration into account. This paper develops a new approach to address both structural and range variations in the car-following process through time. The proposed approach relies upon an intelligent evolving time-variant local model (ETLM), capable of changing its structure and adapting its parameters. The evolving model includes a network of temporal local linear models, each covering a range of car-following behavior in a microscopic traffic flow. Furthermore, a decision-making procedure is designed to determine if model should evolve to a new structure is sole adaptation of its parameter is sufficient to describe the new behavior of the car-following process. The decision-making is carried out based on the comparison between the current temporal linear behavior of the process and existing temporal local linear models. Results of implementation of the ETLM on several benchmark case studies as well as real traffic data demonstrate the efficacy of the proposed approach. Comparisons to other methods show the superior performance of the ETLM model.

INDEX TERMS Car-following behavior, traffic simulation, evolving models.

I. INTRODUCTION

Traffic design, planning and management, as three major areas of traffic studies, require reliable traffic modeling and simulation toolkits. Traffic simulation models are generally divided into macroscopic, microscopic and mesoscopic models [1]. While the macroscopic models focus on the traffic flow as a homogenous process and employ fluid dynamics principles to estimate vehicles propagation in the traffic network, the microscopic traffic models deal with the individual vehicles in more detail and study their reactions, e.g. lane change maneuvers, to the traffic status [2]. Mesoscopic models may be seen as a combination of the macroscopic and microscopic models. Car-following models (CFM), as an important sub-class of microscopic traffic models, describe the process of drivers following each other in a traffic stream. The CFMs have been originally developed to streamline intelligent transportation systems (ITS). Today, with the development of autonomous vehicles, the CFMs will play a stronger role, not only in development of the ITS, but also in providing safety and comfortableness to the modern driving experience.

The associate editor coordinating the review of this manuscript and approving it for publication was M. Venkateshkumar.

A CFM, in essence, attempts to model and then control the behavior of the driver in order to ensure a safe distance to the preceding vehicle [2]. Hence, due to the involvement of the human behavior, the associated uncertainty makes the development of the CFMs a challenging task.

Car-following process is time-variant and features an uncertain nature due to the influence of the human behavior (e.g. level of consciousness, drowsiness, and mental status) as well as ambient factors, such as road condition. These influential factors may lead to wide-range changes in vehicle's velocity and acceleration or structural changes (e.g. delay in reaction to a leading vehicle maneuver) in the driving pattern during a car-following process.

The primary CFMs were developed in 1950s and 60s. The General Motors' or GHR model is the most well-known CFM developed by Chandler, Herman and Montroll at the General Motors (GM) research labs in Detroit [3]. The GM model is a stimulus-response CFM and specifies the stimulus as the relative velocity of vehicles, i.e. each vehicle tends to move at the same speed of its front vehicle. The basic GM model expresses the acceleration of vehicle n in a traffic flow at time t based on its speed and the relative spacing and speed between vehicle n and vehicle $n+1$ (the vehicle immediately in front) assessed at earlier time $t-T$. As the key to the GM model is the specification of its parameters, many researchers focused on optimization of the parameters of the GM model to match different microscopic and macroscopic traffic data [4].

Later, Kometani and Sasaki developed collision avoidance (also known as safety distance) CFM based on a different idea than the GM model. The CA model does not describe a stimulus-response type function but seeks to find a safe following distance within which a collision would be unavoidable if the driver of the FV acted unpredictably [5].

Linear models and psychophysical (or action point models) are another classes of the analytic models developed for car-following modeling. While linear models and included additional terms for the adaptation of the acceleration according to whether the vehicle in front was braking [6], the psychophysical CFMs assume that a driver will perform an action when a threshold, expressed as a function of velocity difference and distance, is exceeded [7].

The optimal velocity model and its derivatives, as another class of stimulus-response models, brought about remarkable improvements to the car-following modeling. The OV model was developed through the assumption that the driver of the FV seeks a safe velocity determined by the distance from the LV [8]. Later, the basic idea was altered and improved by other researchers. For instance, Sawada [9] defined a generalized optimal velocity (GOV) model based on the assumption that the FV's driver pays attention not only to its headway but also to the headway of the immediately preceding vehicle (LV). The generalized force (GF) model made another improvement into the OV model by considering the acceleration caused by the relative speed of the successive vehicles [10]. Next, it was argued by Jiang et al. that the relative speed between the LV

and FV affects the behavior of the FV's driver and therefore should be considered explicitly. Hence they took both positive and negative velocity differences into account and developed the full velocity difference (FVD) car-following model [11]. Through some numerical investigations, it was found the FVD model results in too high deceleration. Hence, the two-velocity difference (TVD) model was proposed by Ge [12] to resolve this shortcoming by taking into account the velocity difference of the two successive preceding vehicles. Later, Wang et al. [13] proposed a generalized multiple velocity difference (MVD) model which considers multiple preceding vehicles' stimulus. In [14], the researcher developed a car-following model based on a piecewise linear approximation of the fundamental traffic diagram. In the piecewise linear model, the stability of the car-dynamics is characterized and the stationary regimes is determined, when they exist.

Later, the more sophisticated data-driven models were applied to enhance the performance of the CFMs [15]. From fuzzy inference systems to neural networks and neuro-fuzzy systems, various data-driven CFMs were developed by the researchers to tackle the difficulties of car-following modeling. For instance, in [16], a fuzzy neural network was developed to forecast travel speed for multi-step ahead. The authors in [17] combined machine learning (a back-propagation neural network) with kinematics models to improve the safety level and robustness of the car-following control of automated vehicles. In [18], it was discussed that the data-driven models have poor interpretability while the theoretical models are unable to describe the individualized features and models of the driver. Hence, they developed a fusion modelling approach including adaptive Kalman filter algorithm and long-short-time memory neural network. The test results of real driving data indicated the superior performance of the fusion model over the individual model. In order to provide flexibility and accuracy to describe complicated human actions in car-following behavior, the authors in [19] proposed a deep neural network. The results of testing on empirical trajectory records showed the higher performance of the deep neural network car-following model over the conventional techniques.

Although there are improvements to the classical modes, but the offline nature of these approaches, i.e. fixed structure with fixed parameters, fails to capture the time-variant dynamics of the car-following behavior. The online models may be put forward as a remedy to improve modeling performance, but they will fail if structural changes occur in the car-following process.

The above-mentioned discussions necessitates the development of CFMs, which are able to cope with nonlinear as well as time-variant dynamics of the car following. A relatively recent class of data-driven models, known as evolving models, are capable to provide these necessary features. This is substantially different to the adaptive models where adaptation achieved only be re-estimation of the parameters of the model. A comparison between adapting and evolving models is depicted Fig. 1. The evolving models are not only able to change the value of their parameters, but also their structure in order to better describe the recent stream of data [20]. During the recent years, remarkable efforts have been made to develop evolving models [21], [22], [23]. Most of these studies have been focused on proposing effective models to cope with nonlinear dynamic systems [24]. Taking the nonlinear dynamics of the car-following behavior, evolving models seem a fit choice to cope with the afore-mentioned property of the car-following process.

This paper proposes a general approach to model processes with time-variant and nonlinear dynamics. The proposed evolving time-variant local model (ETLM) includes an evolution algorithm, which allows for change in the structure of the model based on the most recent stream of data.

More specifically, the developed approaches is comprised of a network of temporal local linear model, each responsible for describing a specific character of the car-following process through time. The ETLM is capable of adapting to the new behavior of the car-following process or evolving to a new structure in order to better describe the process, when sole adaptation is not sufficient to cover the whole process.

The rest of this paper is organized as follows. The overview of the car-following strategy is presented in Section 2.

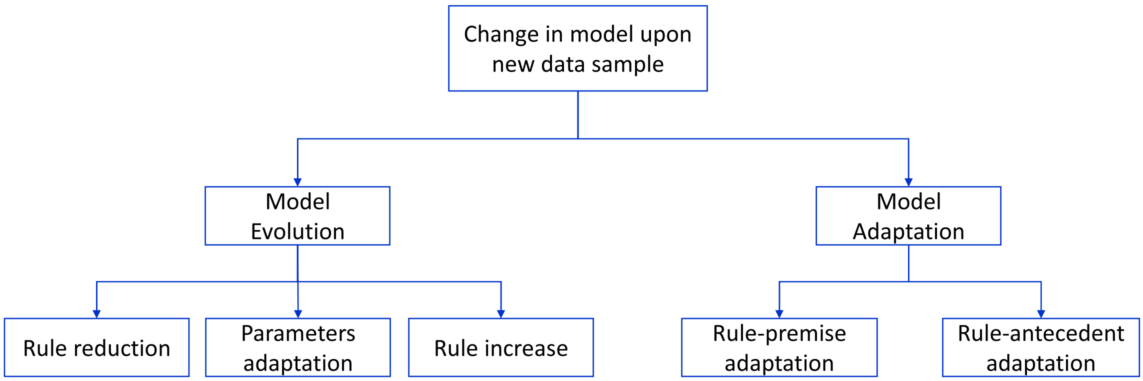

FIGURE 1. Comparison of model change based on new data samples.

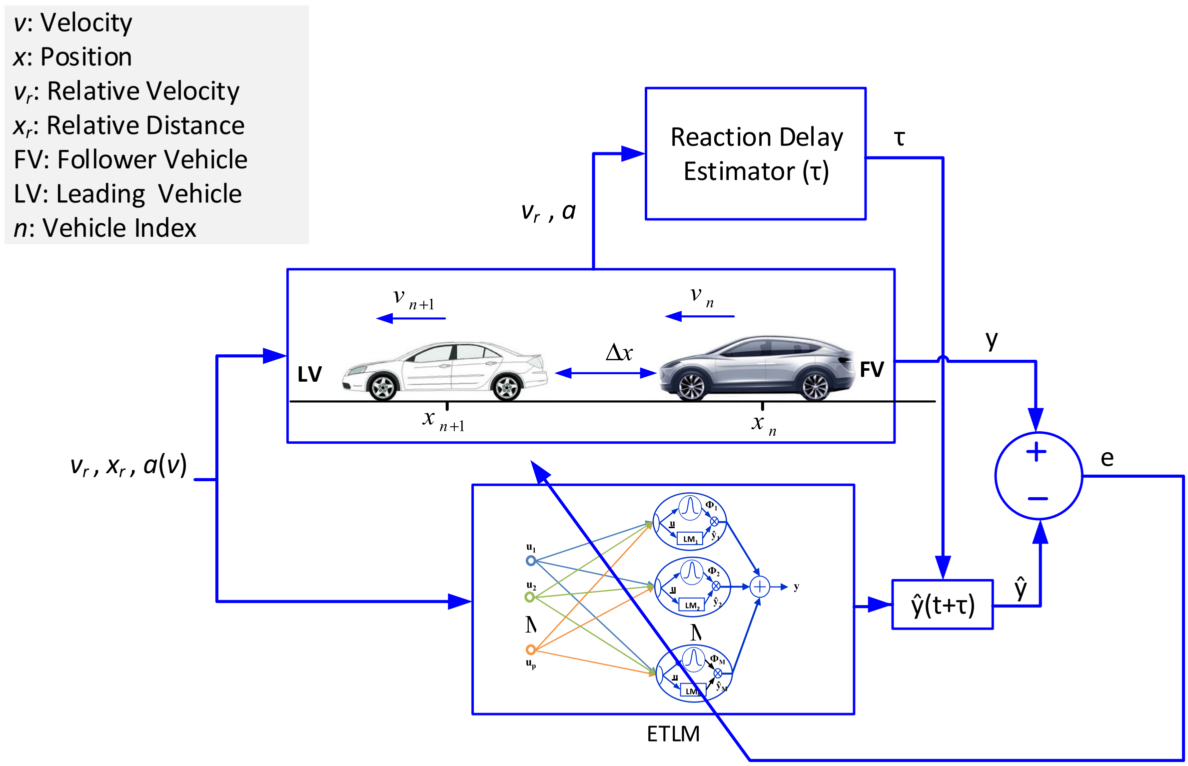

FIGURE 2. Proposed structure to model car-following behavior.

The ETLM and its learning algorithm are explained in Sections 3 and 4, respectively. Results of implementation on several benchmarks and real datasets are reported in Section 5. Finally, concluding remarks are drawn in Section 6.

II. CAR-FOLLOWING FRAMEWORK

A CFM models the behavior of a following vehicle (FV) driver in order to ensure a safe distance to the leading vehicle (LV). More specifically, a car-following model attempts to predict the future acceleration and velocity of the F, using the previous values of acceleration and velocity together with the distance and relative velocity between FV and LV.

Mathematically speaking,

$$a\left(t+\tau\right)=f_a\left(\underline{a}\left(t\right),\underline{v}_r\left(t\right),\underline{x}_r\left(t\right)\right) \quad (1)$$

$$v\left(t+\tau\right)=f_v\left(\underline{v}\left(t\right),\underline{v}_r\left(t\right),\underline{x}_r\left(t\right)\right) \quad (2)$$

where, f_a and f_v are nonlinear functions which predict the future values of acceleration and velocity, respectively, t indicated time, τ is the driver's reaction delay, and the vectors $\underline{a}$, $\underline{v}$, $\underline{v}_r$, and $\underline{x}_r$ indicate velocity, relative velocity and relative distance, respectively. In fact, functions f_a and f_v are evolving models, constructed using the measurement data. The described framework is shown in Fig. 2.

The purpose of this paper is to realize f_a and f_v using the proposed ETLM.

III. EVOLVING LOCAL TIME-VARIANT MODEL

Evolving models are a relatively new concept in the field of data-driven modeling, with the ability to change their structure according to the variations of the system or process [25], [26]. The evolving models are, in fact, data-driven models, which are adapted, extended and evolved on-the-fly in an automatic and dynamic manner based on the new incoming data samples [27].

To develop the ETLM model, we start with the assumption that a nonlinear process with uncertainties and bounded external disturbances can be represented by a linear time-variant system around every spatiotemporal snapshot of the process. In other words, at a specific time and space snapshot, the process is described by the following temporal linear model (TLM),

$$\hat{y}(t) = \vartheta_0(t) + \vartheta_1(t) u_1 + \ldots + \vartheta_p(t) u_p \tag{3}$$

where, $\hat{y}$ is the output of the linear model, $u = \left[u_1 u_2 \ldots u_p\right]^T$ indicates the input vector, and $J(t) = \left[\vartheta_0(t)\, \vartheta_1(t) \ldots \vartheta_p(t)\right]^T$ is the vector of time-dependent linear parameters. The following recursive least squares estimation with forgetting factor [28] can be applied to estimate y:

$$J(t+1) = J(t) + P(t) u^T(t) e(t) \tag{4}$$

$$e(t) = y(t) - \hat{y}(t) \tag{5}$$

$$P(t) = \frac{1}{\lambda(t)} \times \left(P(t-1) - \frac{P(t-1)\tilde{u}(t)\tilde{u}^T(t) P(t-1)}{\lambda(t) + \tilde{u}^T(t) P(t-1)\tilde{u}(t)}\right) \times 0 < \lambda(t) \le 1 \tag{6}$$

where, $\tilde{u}^T(t) = \left[1 u^T(t)\right]$ is the augmented input vector, P represents the covariance matrix and $\lambda(t)$ is known as the forgetting factor.

The model presented in (3)-(6), exhibits the current temporal linear behavior (CTLB) of the main nonlinear process. Now, to develop a model, capable of describing wide-range behavior of a nonlinear process with uncertainties, the time-variant model in (3)-(6) is extended to an evolving time-variant local model (ETLM). The ETLM is composed of a number of time-variant local linear models (TLLM) and is mathematically expressed by,

$$\hat{y}(t) = \sum_{i=1}^{M(t)} \text{TLLM}_i(u(t))\, \Phi_i(u(t))$$
$$\text{TLLM}_i(u(t)) = \theta_{i,0}(t) + \theta_{i,1}(t) u_1 + \ldots + \theta_{i,p}(t) u_p \tag{7}$$

where, $\hat{y}$ is the output of the ETLM, $\theta_i = \left[\theta_{i,0}\theta_{i,1} \ldots \theta_{i,p}\right]^T$ represent the linear parameters of the model, Φ_i is the validity function associated with TLLM_i and $M(t)$ is the number of TLLMs. The model expressed by (7) represents a fuzzy inference system with $M(t)$ fuzzy rules (or equivalently a neural network with $M(t)$ neurons).

The variations of linear parameters, θ_i, indicate adaptive characteristic of the model while the variable number of fuzzy rules through time indicates the evolving feature of the model.

We will start with the estimation of the linear parameters of the TLLMs, θ_i, and then focus on how model evolves through time.

Similar to the linear time-variant model in (3)-(6), estimation of the linear parameters of the ETLM, θ_i, is carried out adaptively by means of a recursive weighted least squares (RWLS) algorithm, as expressed below [29],

$$\theta_i(t+1) = \theta_i(t) + P_i(t)\Phi_i \tilde{u}_i^{\mathrm{T}}(t) e_i(t),\ 1 \le i \le M(t) \tag{8}$$

where,

$$\tilde{u}_i^{\mathrm{T}} = \Phi_i \tilde{u}^{\mathrm{T}}(t) \tag{9}$$

$$e_i(t) = \left(y(t) - \hat{y}(t)\right)\Phi_i \tag{10}$$

$$P_i(t) = \frac{1}{\lambda_i(t)} \times \left(P_i(t-1) - \frac{P_i(t-1)\tilde{u}_i(t)\tilde{u}_i^{\mathrm{T}}(t) P_i(t-1)}{\lambda_i(t) + \tilde{u}_i^{\mathrm{T}}(t) P_i(t-1)\tilde{u}_i(t)}\right) \times 0 < \lambda(t) \le 1 \tag{11}$$

In (8)-(11), P_i represents the covariance matrix and $\lambda_i(t)$ is known as the forgetting factor, associated with TLLM_i.

In this paper, the suitable value of the forgetting factor in (11) is determined based on the following adaptive gradient computation, in terms of estimation error function $L_i(t) = e_i^2(t)$:

$$\lambda_i(t) = \lambda_i(t-1) - \eta \frac{\partial L_i(t)}{\partial \lambda_i(t-1)} \tag{12}$$

where, η denotes learning rate. By taking into account that,

$$\partial P_i(t-1)/\partial \lambda_i(t-1) = -P_i(t-1)\left(\partial P_i^{-1}(t-1)/\partial \lambda_i(t-1)\right) P_i(t-1) \tag{13}$$

and using (8), the following form can be derived for computation of the adaptive forgetting factor,

$$\begin{aligned}\lambda_i(t) &= \lambda_i(t-1) + 2\eta e_i(t)\tilde{u}_i^{\mathrm{T}}(t) \times \left(P_i(t-1)\left(\partial P_i^{-1}(t-1)/\partial \lambda_i(t-1)\right) P_i(t-1)\right) \times \tilde{u}_i(t-1) e_i(t-1) \\ &= \lambda_i(t-1) + 2\eta e_i(t)\tilde{u}_i^{\mathrm{T}}(t) \times \left(P_i(t-1) P_i^{-1}(t-2) P_i(t-1)\right) \times \tilde{u}_i(t-1) e_i(t-1)\end{aligned} \tag{14}$$

In this paper, a fixed learning rate approach is pursued.

The adaptive RWLS algorithm, presented through (8)-IV, provides adaptive estimation for the TLLMs' parameters based on the stream of data.

IV. LEARNING ALGORITHM

A. DECIDING UPON ADAPTATION OR EVOLUTION

To identify the structure of the ETLM, i.e. the validity functions and number of fuzzy rules, a learning algorithm is proposed. This is carried out by continuous comparison between the CTLB of the car following process and the existing TLLMs of the ETLM. In other words, by arrival of each new data sample, first, the TLM (described by (3)-(6)) is compared to all existing TLLMs. If the difference is significant (i.e. higher than a pre-specified threshold), then a new fuzzy rule is added to the model; otherwise, the existing TLLMs are only adapted to the new data. When the latter is the case, adaptation of the TLLMs is simply carried out through (8)-(11). However, for the former case, a procedure should be designed.

Now, suppose new data sample $u(t)$ is arrived at time instant t. This would lead to a new estimation for the TLM, as stated in (4). At this stage, the ETLM would include $M(t)$ TLLMs (or fuzzy rules). To decide whether the model should be adapted to new data, or evolve, the following measure, representing variations in the process structure, is first defined,

$$d_i = \frac{\|J-\theta_i\|^2}{4\left(\|J\|^2+\|\theta_i\|^2\right)}, 1 \le i \le M(t) \tag{15}$$

In (15), the distance between the parameter vectors of the TLM and all TLLMs is computed. Based on (15), the structural changes can be taken into account.

Now, to take range variations into account, the following measure is introduced,

$$r_i = \|u(t+1) - c_i\|, 1 \le i \le M(t) \tag{16}$$

where, $u(t+1)$ is the new data sample and c_i represents the center of TLLMi, as shown in Fig. 3.

The final decision is made by considering a trade-off between r_i and d_i, as stated below:

$$f_i = \alpha d_i + (1-\alpha) r_i \tag{17}$$

The new model evolves if the minimum value of the combinational distance in IV-B exceeds the threshold value, f_{th},

$$m = \arg\min_i \{f_i\}, 1 \le i \le M(t) f_m > f_{th} \tag{18}$$

Hence, the minimum distance, f_m, is compared to a threshold value, f_{th}. If $f_m > f_{th}$, then a new TLLM (or new fuzzy rule) must be added to the evolving model in order to describe the current behavior of the process;

$$M(t+1) = M(t) + 1 \tag{19}$$

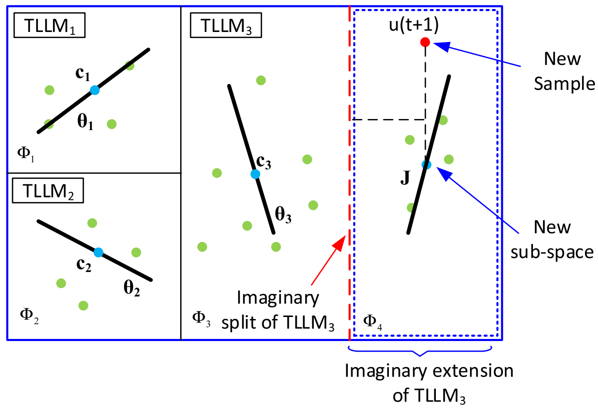

FIGURE 3. Fictitious extension of LLM_3 to construct validity function for new LLM.

otherwise, adaptation of the existing TLLMs is carried out using (8)-(11).

B. EVOLUTION PROCEDURE

Previously, we discussed how a new rule is added based on the Euclidean distance between the parameter vectors of the TLM and existing TLLMs. Now, the problem is how to identify parameters of a new rule; i.e. how to compute the parameters of the validity function associated with the newly added rule.

In this paper a fast heuristic approach, termed evolving partitioning strategy (EPS) is pursued to identify the structure of the new TLLM. The proposed heuristic approach, established based on the idea of hierarchical binary tree (HBT) [31], is fast and efficient and hence suits online applications.

To illustrate how EPS works, let us consider Fig. 3, where a new temporal local linear model (TLLM4) along with its validity function, Φ_4, are supposed to be added to the ETLM. Estimation of the new TLLM's parameters was discussed previously. To construct the new validity function (in this case Φ_4) the idea of HBT in [31] can be effectively employed.

In the HBT algorithm, the validity functions are constructed through axis-orthogonal split of the input space and, then, multiplication of sigmoid functions. However, in our approach no split of the input space occurs, but new sub-spaces are added to the model. To utilize HBT for estimation of the new validity functions parameters, one can assume the sub-space of the new TLLM (for instance, $TLLM_4$ shown by dashed lines in Fig. 3), as the fictitious extension of the adjacent TLLM (for instance $TLLM_3$) and then perform the proper split. In other words, it seems as if there is a large TLLM which is going to be split into $TLLM_3$ and LLM_4.

To demonstrate this idea clearly, let us focus on Fig. 3 with more details. In the situation shown in this Fig., $TLLM_4$ should be added to the model. To identify Φ_4, first the sub-space of $TLLM_3$ is further extended to the right such that the new TLLM (i.e. $TLLM_4$) is exactly covered. Then, the extended sub-space of $TLLM_3$ is split along the suitable direction (shown by red dashed line) to yield sub-spaces for $TLLM_3$ and $TLLM_4$. Finally, the validity function Φ_4 can be computed using the HBT algorithm and based on the performed fictitious split.

For construction of the new validity functions, sigmoid functions are employed based on the HBT idea [31].

The combination of the decision-making algorithm (decision upon adaptation or evolution) and new validity function estimation algorithm empowers the ETLM with the ability to change its structure (in addition to the ability of parameter adaptation). This new features improves model performance to address time-variant processes with changing structure over the time.

V. SIMULATION RESULTS

To present a detailed assessment of the proposed ETLM, four different case studies, implemented in MATLAB, are reported. The case studies include Mackey-Glass time series, laser intensity time series, time-variant system identification and car-following process. The first three cases are benchmark modeling problems, while the fourth case includes two car-following datasets.

To evaluate the proposed car-following approach, the US Federal Highway Administration's NGSIM data are employed [30].The data were collected in 0.1-sec intervals. Any measured sample in this data set has 18 features of each drive-vehicle-unit (DVU), such as longitudinal and lateral position, velocity, acceleration, time, number of road, vehicle class, front vehicle, etc. To have a more detailed analysis, acceleration and velocity of the FV are predicted for two different datasets.

In order to assess the performance of the proposed ETLM numerically, the root man square error (RMSE) and mean absolute percentage error (MAPE), defined below, are used.

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(y_t-\hat{y}_t\right)^2} \tag{20}$$

$$MAPE=\frac{100}{N}\sum_{t=1}^{N}\left|\frac{y_t-\hat{y}_t}{y_t}\right| \tag{21}$$

where, $y(t)$ and $\hat{y}_t$ are the actual and estimated outputs at sample t, respectively, and N is the number of identified samples.

Besides, comparison to the offline LLNF ($\mathrm{LLNF}_{\mathrm{offline}}$) model [29], ANFIS model [32] and adaptive LLNF model ($\mathrm{LLNF}_{\mathrm{adaptive}}$) [29] are presented to demonstrate the effectiveness of online approaches for car-following modeling.

A. MACKEY-GLASS TIME SERIES

Mackey-Glass time is expressed by

$$\frac{dx}{dt}=\frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)}-0.1x(t) \tag{22}$$

This time series is a model for white blood cell production and is chaotic for $\tau > 16.8$. The standard input variables for this case are $x(t-18)$, $x(t-12)$, $x(t-6)$ and $x(t)$ for predicting $x(t+6)$.

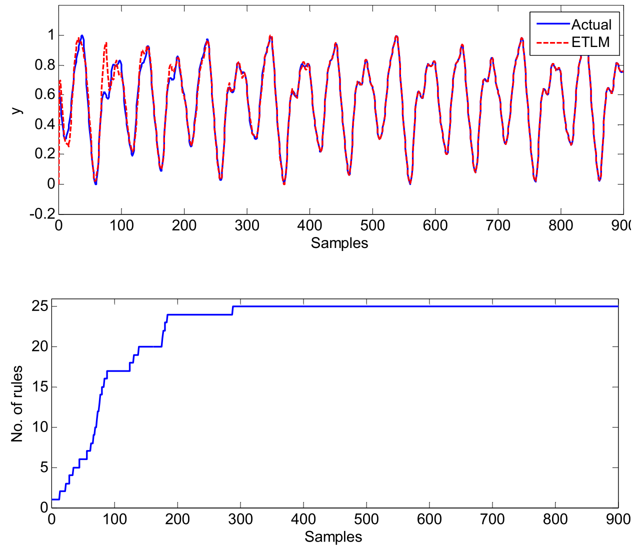

FIGURE 4. ETLM performance and number of generated rules in case of Mackey-Glass time series.

The ETLM is applied to the time series and the produced results are illustrated in Fig. 4. In this Fig. the performance of the ETLM and number of the generated fuzzy rules are presented. It is clear that the proposed model successfully converges to the original time series after about 100 samples.

The produced results are compared to other existing methods in [33] in Table 1, indicating the superior performance of the proposed approach.

TABLE 1. Comparison of the performance in case of mackey-Glass time series.

Model	RMSE
Neural tree model [33]	0.0069
WNN [33] + gradient	0.0071
WNN [33] + hybrid	0.0059
LLWNN [33] + gradient	0.0041
LLWNN [33] + hybrid	0.0036
IT2FNN-3 [33]	0.0020
MSBFNN [33]	0.0024
SEIT2FNN [33]	0.0034
FLNFN-CCPSO [33]	0.0084
LLNFoffline	0.0027
ETLM	0.0022

B. LASER INTENSITY TIME SERIES

The laser intensity time series represents an experimental data, measured in a physical laboratory experiment from a Far-Infrared-Laser in a chaotic state. The measurements were made on an 81.5-micron 14NH3 cw (FIR) laser with the signal-to-noise ratio of about 300 [34]. The laser intensity time series, shown in Fig. 5, exhibits complicated behavior and is predictable on short time scales, but challenging on a global scale modeling [35].

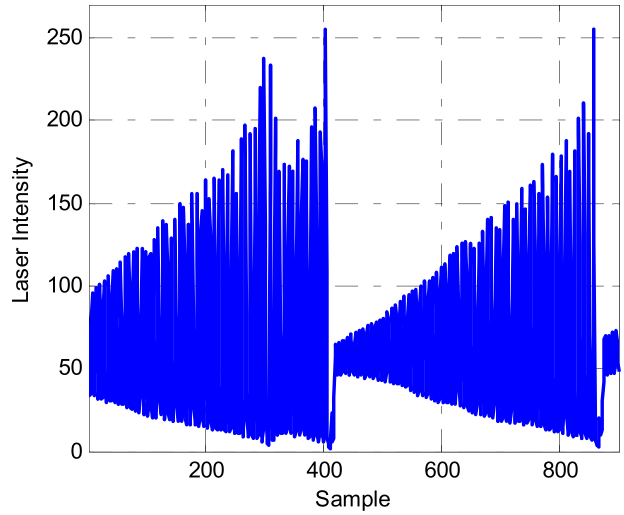

FIGURE 5. Laser intensity time series.

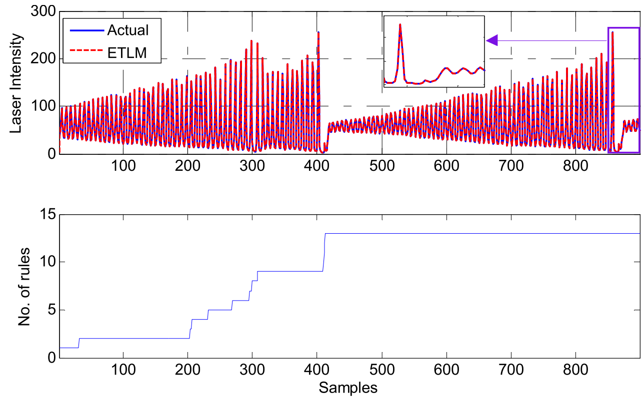

FIGURE 6. Prediction of laser intensity time series (top) and number of generated rules (bottom) by the ETLM.

The proposed ETLM is applied to the laser intensity time series and the results are illustrated in Fig. 6. Interestingly the evolving model has successfully captured the chaotic behavior of the laser intensity time series over the whole time range, as emphasized by the zoomed area, designated at the end of the series.

To provide a comparative study with the available approaches, the performance of the ETLM is presented against several adaptive models proposed in [35] including the recursive Bayesian recurrent neural networks, in Table 2.

TABLE 2. Comparison of the performance in case of laser intensity time series.

Model	RMSE
MLP-EKF [35]	0.00468
MLP-BLM [35]	0.00093
RNN-BPTT [35]	0.01092
RNN-RTRL [35]	0.00876
RNN-EKF [35]	0.00436
RBLM-RNN [35]	0.00060
ETLM	0.00029

C. TIME-VARIANT SYSTEM IDENTIFICATION

The third case study is devoted to modeling of the following time-variant system [36],

$$y(t) = \frac{y(t-1)}{2 + y^2(t-1)} + x^3(t) + h(t) \tag{23}$$

where,

$$h(t) = \begin{cases} 0, 1 \le t \le 1000 \\ 1, 1001 \le t \le 2000 \\ 0, 2001 \le t \end{cases} \tag{24}$$

and

$$x(t) = \sin(2\pi t/100) \tag{25}$$

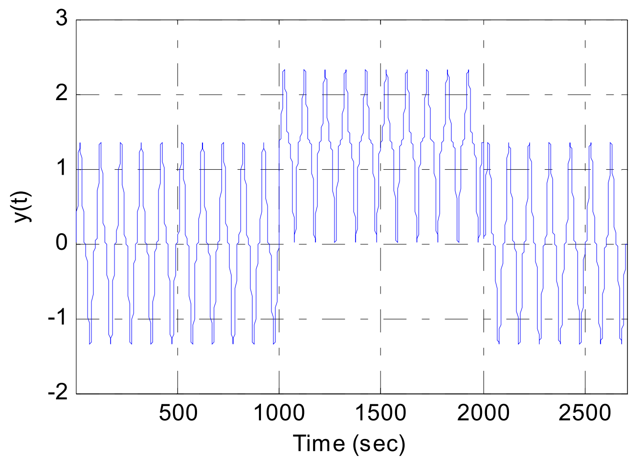

FIGURE 7. The process output in case of time-variant system identification.

A snapshot of the whole system is shown in Fig. 7.

To model the system in (22)-(24), the inputs and output Table 3 are used.

TABLE 3. Inputs and outputs of the ETLM for the case of time-variant system identification.

Output variable	$y(t+1)$
Input variables	$y(t)$, $h(t)$

The proposed approach is applied to the data generated from (22). Prior to $t = 1000$, six TLLMs are identified by the EPS algorithm. After $t = 1000$, when the system behavior changes, the EPS algorithm adds another TLLM and its associated validity function to the ETLM (it is seen in Fig. 8 that the range of system output changes). Performance of the proposed approach against this variation is shown in Fig. 8. It is seen that the ETLM quickly evolves and follows the behavior of the system after $t = 1000$. The jump in the

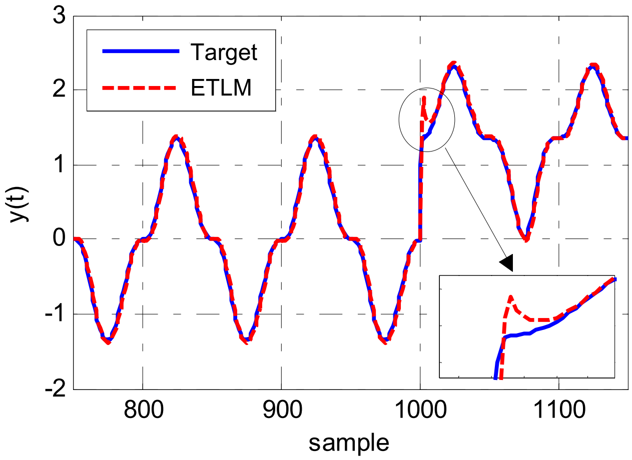

FIGURE 8. Evolution of the ETLM model for the case of time-variant system identification at $t = 1000$ s.

output of the ETLM after $t = 1000$ has occurred due to the change in the structure of the model.

On the other hand, at $t = 2000$ the system returns to its previous state prior to $t = 1000$. Since this behavior has been experienced by the ETLM previously, it is expected that the model follows this behavior without undergoing any evolution. The simulation results demonstrate that no new LLM is added after $t = 2000$ and the ETLM quickly follows the system, as depicted in Fig. 9. Interestingly, in contrast to previous change, no jump or spike is observed in the output of the ETLM after $t = 2000$ since model structure remains unchanged.

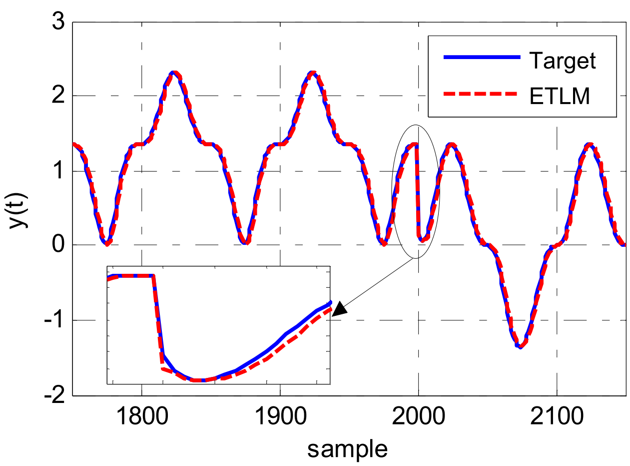

FIGURE 9. Successful tracking of process output in the case of time-variant system identification after variation at t = 2000 s.

To demonstrate the capability of the proposed approach, a comparison to the $\text{LLNF}_{\text{adaptive}}$ model with adaptation of the linear parameters by the RWLS algorithm, is shown Table 4. Clearly, the performance of the proposed model after the first change, $t = 1000$, is much more better than the LLNFadpative model. Furthermore, results of numerical comparison of the proposed approach to LLNFoffline, LLNFonlinf and ANFIS are reported in [36] indicating the satisfactory performance of the evolving model.

TABLE 4. Comparison of performance of different models for the case of time-variant system identification.

Method	RMSE
$\text{LLNF}_{\text{offline}}$	0.060
$\text{LLNF}_{\text{adaptive}}$	0.035
ANFIS	0.038
ETLM	0.021

D. CAR-FOLLOWING PROCESS

1) SELECTED DATASETS

It must be noted that NGSIM database includes a huge volume of car-following data sets, featuring different driver behaviors during the car following process. It is worth noting that in most of available studies, usually datasets with normal behavior and insignificant changes have been employed. For instance, in most of the published researches, the leading vehicle does not change during the car following process.

However, in this paper to demonstrate the performance of the proposed evolving model in case of significant variations in car-following process, two different datasets are employed. In the first dataset, a normal behavior is observe while in the second one, the leading vehicle changes twice over the selected interval, as shown in Figs. 10 and 11, respectively. Clearly, for the second dataset, the leading vehicle changes at two time instants.

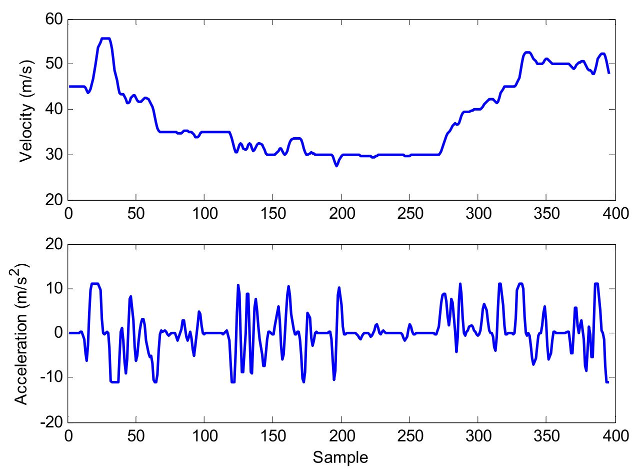

FIGURE 10. Car-following data: Case 1.

2) CASE 1

For the first dataset, ETLM with 3 and 7 fuzzy rules are resulted as shown in Figs. 12 and 13, respectively. It is observed that the ETLM has successfully evolved during the through time to account for changes and variations in the driver's behavior during the car-following process.

On the other hand Fig. 14 shows the velocity and acceleration profiles, produced by the $\text{LLNF}_{\text{offline}}$ model.

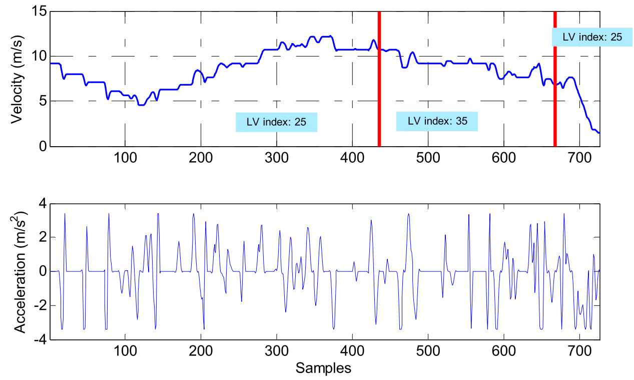

FIGURE 11. Car-following data: Case 2.

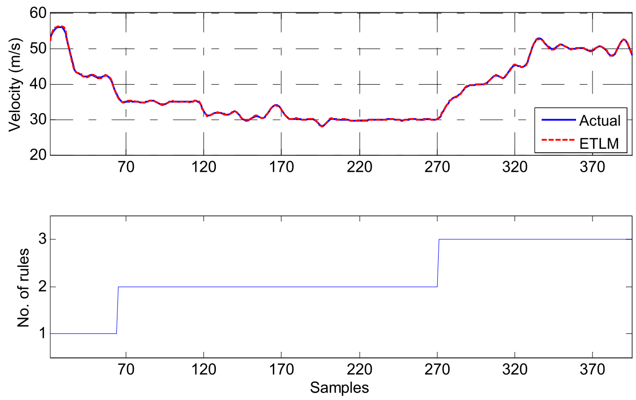

FIGURE 12. Velocity modeling performance (top) and number of generated rules by the ETLM for Case 1.

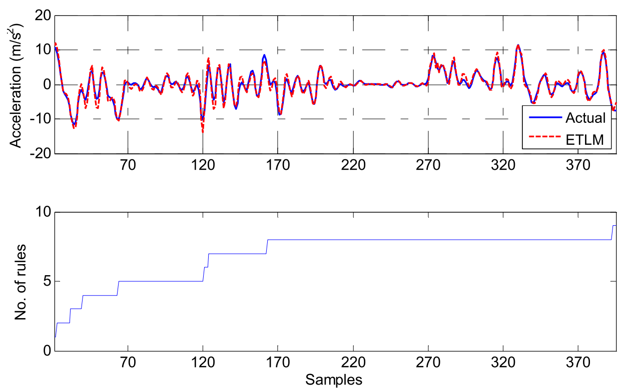

FIGURE 13. Acceleration modeling performance (top) and number of generated rules by the ETLM for Case 1.

Although quite inferior to the proposed ETLM, but the performance of the offline model is fairly acceptable.

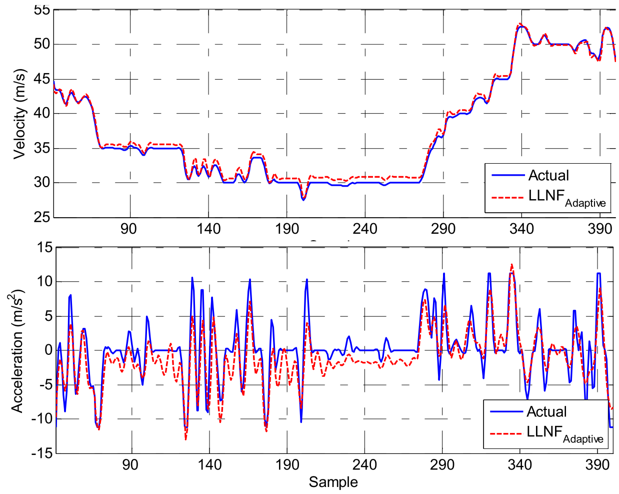

FIGURE 14. Velocity (top) and acceleration (bottom) modeling performance of the $LLNF_{offline}$ model for case 1.

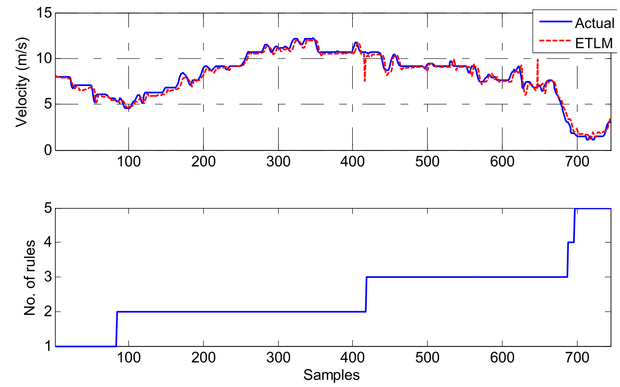

FIGURE 15. Velocity modeling performance (top) and number of generated rules by the ETLM for Case 2.

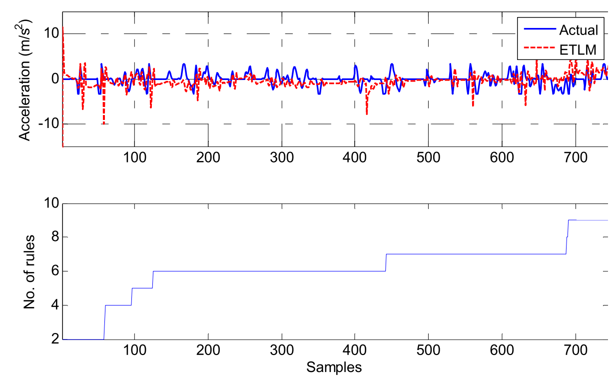

FIGURE 16. Acceleration modeling performance (top) and number of generated rules by the ETLM for Case 2.

3) CASE 2

Performance of the proposed ETLM for predicting velocity and acceleration of the first dataset is shown in Figs. 15 and 16, respectively. These Figs. also demonstrate the evolution behavior of the proposed model in terms of number of generated rules over the time. Interestingly, the ETLM has tracked the variations in velocity and acceleration of the FV even when the LV changes.

For the second dataset, performance of the $LLNF_{offline}$ model is depicted in Fig. 17. It is obvious that at time instant when the LV changes, the performance of the model fails.

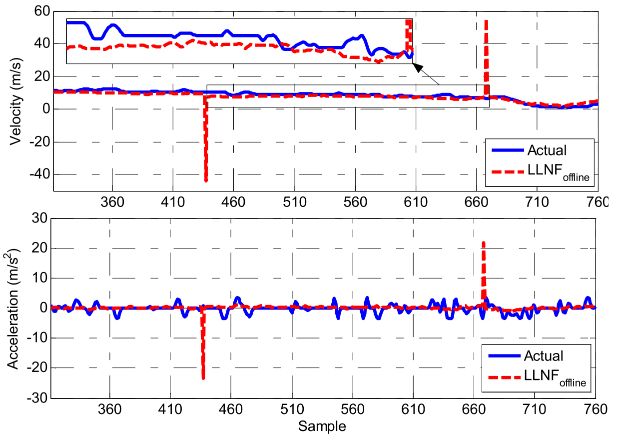

FIGURE 17. Velocity (top) and acceleration (bottom) modeling performance of the LLNFoffline model for case 2.

This indicates that the offline trained model is not suitable for time-varying application such as car-following process modeling.

TABLE 5. Numerical assessment and comparison for velocity modeling.

Dataset / Method	Case 1		Case 2	
	RMSE (m/s)	Improvement (%)	RMSE (m/s)	Improvement (%)
$LLNF_{offline}$	0.39	69.23	4.95	82.02
$LLNF_{adaptive}$	0.21	42.85	2.26	60.61
ANFIS	0.26	53.84	3.12	71.47
ETLM	0.12	-	0.89	-

TABLE 6. Numerical assessment and comparison for acceleration modeling.

Dataset / Method	Case 1		Case 2	
	RMSE (m/s^2)	Improvement (%)	RMSE (m/s^2)	Improvement (%)
$LLNF_{offline}$	2.96	69.59	11.38	90.07
$LLNF_{adaptive}$	1.88	52.12	4.04	72.02
ANFIS	2.09	56.93	4.56	75.21
ETLM	0.90	-	1.13	-

4) NUMERICAL ASSESSMENT AND COMPARISON

Numerical assessment and comparison between the proposed ETLM, $LLNF_{offline}$ model, $LLNF_{adaptive}$ model and ANFIS is presented in this section. Table 5 and Table 6 summarize the comparison results in terms of RMSE for velocity and acceleration modeling, respectively, in both case 1 and case 2 datasets. Moreover, improvement achieved by the ETLM over the other three models is reported in these tables, as well.

The superior performance of the proposed approach is obvious in all case studies. The interesting finding is the higher improvement achieve through the ETLM for case 2, which includes a more challenging dataset. This indicate that in case of significant variations in the structure of car-following data, the evolving model exhibits its major advantage over the offline and adaptive approaches, which in case of normal data, performance of the evolving and other models is relatively comparable.

VI. CONCLUSION

Uncertainties associated with the human behaviors makes the modeling of car-following process a challenging task. This paper proposed an evolving approach to model car-following process in a traffic flow. The evolving model is composed of temporal local linear model (TLLMs), added through time when evolution process is necessary to account for the variations in changes observed in the car-following behavior. In the proposed approach, the distance between the existing TLLMs and current temporal linear behavior of the car-following process was used as a measure for evolution or adaptation of the ETLM. Results of simulations on several benchmark datasets as well as real car-following data set demonstrated effectiveness of the proposed evolving model, particularly when the variations in the structure of the car-following behavior is significant. Comparison to adaptive models showed the superiority of the proposed approach.

ACKNOWLEDGMENT

The authors would like to thank the U.S. Federal Highway Administration and Next Generation SIMulation (NGSIM) for providing the dataset used in this article.

REFERENCES

[1] L. A. Pipes, "An operational analysis of traffic dynamics," *J. Appl. Phys.*, vol. 24, pp. 274–287, Mar. 1953.
[2] A. Ghaffari, A. Khodayari, N. Hosseinkhani, and S. Salehinia, "The effect of a lane change on a car-following manoeuvre: Anticipation and relaxation behaviour," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 229, no. 7, pp. 809–818, Jun. 2015.
[3] R. E. Chandler, R. Herman, and E. W. Montroll, "Traffic dynamics: Studies in car following," *Oper. Res.*, vol. 6, no. 2, pp. 165–184, 1958.
[4] M. Brackstone and M. McDonald, "Car following: A historical review," *Transp. Res., F, Traffic Psychol. Behav.*, vol. 2, pp. 181–196, Dec. 1999.
[5] E. Kometani and T. Sasaki, "Dynamic behaviour of traffic with a nonlinear spacing-speed relationship," in *Proc. Symp. Theory Traffic Flow, Res.* New York, NY, USA: Elsevier, 1959, pp. 105–119.
[6] T.-H. Yang and C.-W. Zu, "Linear dynamic car-following model," in *Proc. 5th World Congr. Intell. Control Automat.*, vol. 6, Jun. 2004, pp. 5212–5216.
[7] W. Wang, W. Zhang, D. Li, K. Hirahara, and K. Ikeuchi, "Improved action point model in traffic flow based on driver's cognitive mechanism," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2004, pp. 447–452.
[8] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, "Dynamics model of traffic congestion and numerical simulation," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 51, pp. 1035–1042, Feb. 1995.
[9] S. Sawada, "Nonlinear analysis of a differential-difference equation with next-nearest-neighbour interaction for traffic flow," *J. Phys. A, Math. Gen.*, vol. 34, no. 50, pp. 11253–11259, Dec. 2001.
[10] D. Helbing and B. Tilch, "Generalized force model of traffic dynamics," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 58, no. 1, pp. 133–138, Jul. 1998.
[11] R. Jiang, Q.-S. Wu, and Z.-J. Zhu, "A new continuum model for traffic flow and numerical tests," *Transp. Res. B, Methodol.*, vol. 36, no. 5, pp. 405–419, Jun. 2002.
[12] H. X. Ge, S. Q. Dai, Y. Xue, and L. Y. Dong, "Stabilization analysis and modified Korteweg–de Vries equation in a cooperative driving system," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 71, no. 6, Jun. 2005.

[13] T. Wang and J. Zhang, "Analysis of multiple velocity difference model in two-lane traffic flow with an accident," in *Proc. IEEE Int. Conf. Autom. Logistics (ICAL)*, Sep. 2008, pp. 2805–2807.
[14] N. Farhi, "Piecewise linear car-following modeling," *Transp. Res. C, Emerg. Technol.*, vol. 25, pp. 100–112, Dec. 2012.
[15] V. Papathanasopoulou and C. Antoniou, "Towards data-driven car-following models," *Transp. Res. C, Emerg. Technol.*, vol. 55, pp. 496–509, Jun. 2015.
[16] J. Tang, F. Liu, Y. Zou, W. Zhang, and Y. Wang, "An improved fuzzy neural network for traffic speed prediction considering periodic characteristic," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2340–2350, Sep. 2017.
[17] D. Yang, L. Zhu, Y. Liu, D. Wu, and B. Ran, "A novel car-following control model combining machine learning and kinematics models for automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 1991–2000, Jun. 2019.
[18] Y. Li, X. Lu, C. Ren, and H. Zhao, "Fusion modeling method of car-following characteristics," *IEEE Access*, vol. 7, pp. 162778–162785, 2019.
[19] X. Wang, R. Jiang, L. Li, Y. Lin, X. Zheng, and F.-Y. Wang, "Capturing car-following behaviors by deep learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 910–920, Jul. 2017.
[20] M. Komijani, C. Lucas, B. N. Araabi, and A. Kalhor, "Introducing evolving Takagi–Sugeno method based on local least squares support vector machine models," *Evolving Syst.*, vol. 3, no. 2, pp. 81–93, Jun. 2012.
[21] M. Pratama, J. Lu, and G. Zhang, "Evolving type-2 fuzzy classifier," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 3, pp. 574–589, Jun. 2016.
[22] H. H. Y. Sa'ad, N. A. M. Isa, and M. M. Ahmed, "A structural evolving approach for fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 2, pp. 273–287, Feb. 2020.
[23] P. V. D. C. Souza and E. Lughofer, "EFNN-NullUni: An evolving fuzzy neural network based on null-uninorm," *Fuzzy Sets Syst.*, vol. 449, pp. 1–31, Nov. 2022.
[24] A. P. F. Evangelista and G. L. D. Serra, "Type-2 fuzzy instrumental variable algorithm for evolving neural-fuzzy modeling of nonlinear dynamic systems in noisy environment," *Eng. Appl. Artif. Intell.*, vol. 109, Mar. 2022, Art. no. 104620.
[25] N. Kasabov, *Evolving Connectionist Systems: The Knowledge Engineering Approach*, 2nd ed. London, U.K.: Springer, 2007.
[26] P. D. Pantula, S. S. Miriyala, and K. Mitra, "An evolutionary neuro-fuzzy C-means clustering technique," *Eng. Appl. Artif. Intell.*, vol. 89, Mar. 2020, Art. no. 103435.
[27] E. Lughofer, "A dynamic split-and-merge approach for evolving cluster models," *Evolving Syst.*, vol. 3, no. 3, pp. 135–151, Sep. 2012.
[28] S. Haykin, *Adaptive Filter Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
[29] R. Kazemi and M. Abdollahzade, "An adaptive framework to enhance microscopic traffic modelling: An online neuro-fuzzy approach," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 230, no. 13, pp. 1767–1779, Nov. 2016.
[30] *[Dataset] U. S. Federal Highway Administration's Next Generation Simulation*. Accessed: 2016. [Online]. Available: http://ngsim.fhwa.dot.gov
[31] M. Abdollahzade and R. Kazemi, "A developed local polynomial neuro-fuzzy model for nonlinear system identification," *Int. J. Artif. Intell. Tools*, vol. 24, no. 3, Jun. 2015, Art. no. 1550004.
[32] A. Khodayari, A. Ghaffari, R. Kazemi, F. Alimardani, and R. Braunstingl, "Improved adaptive neuro fuzzy inference system car-following behaviour model based on the driver–vehicle delay," *IET Intell. Transp. Syst.*, vol. 8, no. 4, pp. 323–332, Jun. 2014.
[33] S. Yilmaz and Y. Oysal, "Fuzzy wavelet neural network models for prediction and identification of dynamical systems," *IEEE Trans. Neural Netw.*, vol. 21, no. 10, pp. 1599–1609, Oct. 2010.
[34] *[Dataset] He Santa Fe Time Series Competition Data*. Accessed: 2019. [Online]. Available: http://www-psych.stanford.edu/ andreas/Time-Series/SantaFe.html
[35] D. T. Mirikitani and N. Nikolaev, "Recursive Bayesian recurrent neural networks for time-series modeling," *IEEE Trans. Neural Netw.*, vol. 21, no. 2, pp. 262–274, Feb. 2010.
[36] C.-F. Juang and Y.-W. Tsao, "A self-evolving interval type-2 fuzzy neural network with online structure and parameter learning," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 6, pp. 1411–1424, Dec. 2008.

MAJID ABDOLLAHZADE received the B.Sc. degree in mechanical engineering from the K. N. Toosi University of Technology, Tehran, Iran, in 2005, the M.Sc. degree from the University of Tehran, Tehran, in 2008, and the Ph.D. degree from the K. N. Toosi University of Technology. He is currently an Assistant Professor at Islamic Azad University, Pardis Branch, Tehran. He has been involved in various research projects in the area of nonlinear system identification, AI techniques, and machine learning. He has published more than 30 papers in international journals and conference proceedings. His current research interests include computational intelligence, machine learning, nonlinear system identification, control, signal processing, and time series prediction. He is a reviewer of several international journals in the area of computational intelligence and forecasting.

REZA KAZEMI was born in Ahwaz, Iran, in 1970. He received the B.S. and M.S. degrees in mechanical engineering from the Isfahan University of Technology, Isfahan, Iran, in 1993 and 1995, respectively, and the Ph.D. degree in mechanical engineering from the Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2001. He is currently an Associate Professor in mechanical engineering and the Head of the Advanced Vehicle Systems Research Center (AVSRC), K. N. Toosi University of Technology, Tehran. He is also involved in modeling, design, and improvement of vehicle steering, suspension, braking systems, vehicle dynamics, and nonlinear and intelligent control systems.