



# Spatial–temporal transformer-based ecological car-following strategy for connected electric vehicles in dynamic environments

Hao Sun <sup>a</sup> <sup>1</sup>, Shuang Li <sup>b</sup>, Bingbing Li <sup>c</sup>\*, Mingyang Chen <sup>a</sup>, Sunan Zhang <sup>c</sup>, Weichao Zhuang <sup>c</sup>, Guodong Yin <sup>c</sup>, Boli Chen <sup>a</sup>,\*

<sup>a</sup> Department of Electronic and Electrical Engineering, University College London, London, WC1E 6BT, UK

<sup>b</sup> School of Transportation, Southeast University, Nanjing, 211189, China

<sup>c</sup> School of Mechanical Engineering, Southeast University, Nanjing, 211189, China

## ARTICLE INFO

### Keywords:

Car-following strategy  
Spatial–temporal transformer  
Robust model predictive control  
Convex optimization

## ABSTRACT

This study proposes a learning-based ecological car-following strategy for connected and autonomous vehicles (CAVs). Leveraging advanced vehicle-to-vehicle and vehicle-to-infrastructure technologies, CAVs on roads can receive updated traffic flow information and predict the speed profile of the preceding vehicle using a macro-micro fused spatial–temporal transformer. To handle the discrepancies between the predicted and actual speeds of the preceding vehicle and achieve less conservative results, a robust learning-in-the-loop model predictive control algorithm has been developed. Furthermore, to enable real-time computation for practical applications, the system models are transformed from the time domain to the spatial domain, integrating a fictitious control input design for system convexification. Finally, a comprehensive assessment of the proposed strategy is conducted through numerical simulations and on-road vehicle experiments.

## 1. Introduction

As human society advances, environmental pollution has emerged as one of the most pressing challenges of our time, posing threats to ecosystems, human health, and the sustainability of our planet [1]. The emission of transportation exhaust is one of the major factors contributing to environmental pollution. To handle this, electrifying conventional internal combustion engines is regarded as a trend in the future development of the automotive industry [2,3].

In the process of vehicle electrification, the battery range has attracted widespread attention, prompting researchers to conduct studies on ecological driving based on connected and autonomous technologies [4–8]. [8] proposes an adaptive cruise control (ACC) strategy for electric vehicles, taking into account the impact of prediction horizon length on optimality. It assumes that the speed profile for each time step is known to the ego vehicle, which is not practical. In [5], an Eco-driving algorithm based on model predictive control (MPC) for hybrid electric vehicles is introduced, where the acceleration of the preceding vehicle is set constant within a predictive horizon. However, the assumption made in the aforementioned study could cause deviations in the speed profile of the preceding vehicle, potentially resulting in collisions. Therefore, many researchers investigate ecological ACC combining data-driven vehicle prediction [9–12].

The issue of predicting vehicle speed has been extensively researched over a prolonged period. Various statistical models have been applied for speed estimation to reduce energy consumption, such as Auto Regressive Integrated Moving Average (ARIMA) [13,14] and Markov Chain models [15,16]. However, these models are unable to account for the non-linear and long temporal dependencies of traffic conditions. Neural networks have exhibited better performance in prediction for vehicle energy management than Markov Chain [17]. Therefore, various networks, such as Radial basis function neural networks (RBF-NN) [17–19], Back propagation neural networks (BPNN) [20,21], have been applied to improve the accuracy of vehicle speed prediction. The deep learning technologies further enhanced the vehicle speed prediction performance [22]. To capture the time dependence while avoiding the gradient explosion problem, the Long Short-term Memory (LSTM) network has been widely used to predict the speed of electric vehicles [23,24]. Convolutional Neural Network (CNN) has also been combined with bidirectional LSTM to capture local temporal features to improve the accuracy of vehicle speed prediction [25]. As Transformer shows outstanding ability for long temporal sequence, [11] proposes an Informer-based speed predictor for the preceding vehicle while planning the speed of connected vehicles. Although previous studies have proposed various efficient predictors,

\* Corresponding authors.

E-mail addresses: [bingli@seu.edu.cn](mailto:bingli@seu.edu.cn) (B. Li), [boli.chen@ucl.ac.uk](mailto:boli.chen@ucl.ac.uk) (B. Chen).

<sup>1</sup> Hao Sun and Shuang Li contributed equally to this work.

Nomenclature	
ACC	Adaptive Cruise Control
ARIMA	Auto Regressive Integrated Moving Average Model
BPNN	Back Propagation Neural Network
CAV	Connected and Autonomous Vehicle
CNN	Convolutional Neural Network
GCN	Graph Convolutional Network
LEC	Learning-based Ecological Car-following strategy
LSTM	Long Short-Term Memory
MAH	Multi-Head Attention Mechanism
MFST	Macro-micro Fused Spatial-temporal Transformer
MPC	Model Predictive Control
RBF-NN	Radial Basis Function Neural Network
STT	Spatial-Temporal Transformer
$C_d, C_f$	Air drag and rolling resistance coefficients
$E, \Delta t$	Kinetic energy and time headway
$F$	Propulsive force supplied by the motor
$g$	Gravitational coefficient
$m$	Vehicle mass
$P$	Power from the motor
$p, v$	Position and velocity of vehicle
$r$	Radius of the wheel
$T$	Driving/Braking torque
$x, u$	Actual state and input
$\hat{x}, \hat{u}$	Predictive state and input
$\bar{x}, \bar{u}$	Nominal state and input
$\Delta s$	Sampling distance interval
$\eta_m$	Motor efficiency
$\eta_t$	Transmission efficiency of the powertrain
$\theta$	Road slope angle
$\omega$	Angular velocity

their models only focused on the self-temporal sequence, neglecting the importance of spatial features. As vehicle speed is significantly affected by the surrounding environment, the macro traffic flow of the road networks is vital for speed prediction [26]. This type of information not only offers a more comprehensive understanding of traffic conditions but also captures the spatial correlations between each road segment. Such correlations are invaluable for enhancing prediction accuracy but seldom be considered in speed predictor design.

However, regardless of the accuracy of the predicted velocity, discrepancies always exist between the predicted velocity and the actual velocity of the CAV. Moreover, modeling uncertainties and measurement noises are also commonly present in the dynamic system. In order to handle these errors,  $H$ -infinity control [27,28] and sliding mode control [29,30] are the most widely adopted robust control methods. Meanwhile, with the capability to directly incorporate hard constraints, many studies have increasingly focused on employing robust MPC strategies. [31] introduces a learning-based MPC, integrating the Gaussian process model to handle the state estimation errors. Recently, an ecological and robust control strategy [32] has been proposed for plug-in hybrid electric vehicles. This strategy incorporates a deep learning module with advanced feature fusion technologies to enhance speed prediction. Additionally, a tube-based value function is integrated into the robust MPC-based control layer, aiming to further alleviate system uncertainties. However, the aforementioned studies [31,32]

face challenges in handling nonlinear optimization problems. Therefore, it is meaningful to design robust control algorithms that can be implemented in real time at the current stage.

In this paper, we propose a Learning-based Ecological Car-following strategy (LEC) for electric CAVs. The speed of the preceding vehicle is predicted through a Macro-micro Fused Spatial-temporal Transformer (MFST), while any discrepancies between the actual and predicted states can be addressed in the control layer. Specific contributions of this paper are summarized as follows:

- A spatial-temporal transformer model is proposed in this study. The micro and macro traffic information are fused by tailored modules for preceding vehicle speed prediction, and we obtained the best prediction performance compared to the baseline methods.
- Following the last point, the proposed LEC strategy can handle the discrepancies between the predicted state and the actual state of the preceding vehicle. Compared to the conventional robust approach [33], our method can yield less conservative solutions while maintaining robust/safe behavior. Additionally, the dynamic system is reformulated in the spatial domain to achieve real-time computation.
- Through comparison with state-of-the-art algorithms, the advantages of the LEC strategy are highlighted in terms of computational efficiency and energy savings. Furthermore, the on-road vehicle experiment is conducted to validate the practical applicability of the proposed method.

The rest of this paper is organized as follows. Section 2 describes the car-following strategy and specifies the system models. In Section 3, the methodology of the proposed LEC strategy is introduced. Numerical simulation results are presented in Section 4 and the on-road vehicle experiment is demonstrated in Section 5. Finally, the conclusion and the future work are given in Section 6.

**Notation:** Let  $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{>0}$  denote the real, the non-negative real, and the strict positive real sets of numbers, respectively. Given a vector  $\mathbf{x} \in \mathbb{R}^n$ , the Euclidean norm of  $\mathbf{x}$  is denoted by  $|\mathbf{x}|$ . Given a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , then  $\|\mathbf{A}\|$  will denote  $\max_{\mathbf{x} \in \mathbb{R}^n \setminus \{0\}} \{|\mathbf{Ax}|/|\mathbf{x}|\}$ , and  $\mathbf{A}_{[i,j]}$  represent  $(i,j)$ th entry of the matrix. The Minkowski sum of sets  $\mathbb{W}, \mathbb{V}$  is  $\mathbb{W} \oplus \mathbb{V} = \{x + y \mid x \in \mathbb{W}, y \in \mathbb{V}\}$ ; the Minkowski difference of sets  $\mathbb{W}, \mathbb{V}$  is  $\mathbb{W} \ominus \mathbb{V} = \{x \mid x \oplus \mathbb{V} \subseteq \mathbb{W}\}$ .

## 2. Problem formulation

This paper considers an ecological car-following strategy for electric vehicles, incorporating speed profile forecasting and tracking. The velocity of the preceding vehicle can be forecasted using the MFST, and a robust learning-in-the-loop MPC algorithm within the control layer can achieve car-following behavior and address the discrepancies between the predicted state and the actual state. Furthermore, the proposed strategy is capable of real-time computation based on a spatial domain modeling approach. For more details, the vehicle dynamic model and the power consumption model of the in-wheel motor are introduced, along with reformulation into convex form in the spatial domain, which is presented in the following subsections (see Fig. 1).

### 2.1. In-wheel electric motor model

The CAV is driven by two PD18 in-wheel motors, whose power consumption models can be characterized as follows:

$$P(t) = \begin{cases} \frac{T(t)\omega(t)}{\eta_m(T(t), \omega(t))}, & T(t) \geq 0 \\ T(t)\omega(t)\eta_m(T(t), \omega(t)), & T(t) < 0 \end{cases} \quad (1a)$$

$$T(t) = \frac{F(t)r}{\eta_t}, \quad (1b)$$

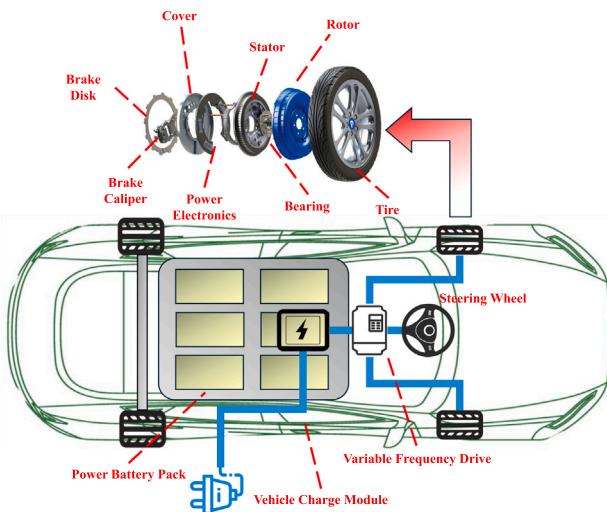


Fig. 1. The configuration of the CAV being driven by two PD18.

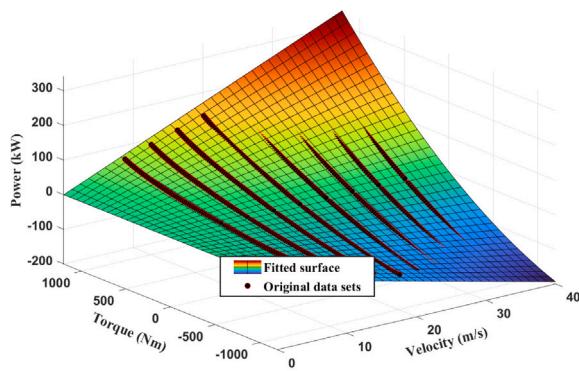


Fig. 2. Power fitting map of the in-wheel motor.

$$\omega(t) = \frac{v(t)\eta_t}{r} \quad (1c)$$

where  $\eta_m(T(t), \omega(t))$  represents the motor efficiency which depends on the angular velocity  $\omega(t)$  and the input torque  $T(t)$ .  $F(t)$  is the propulsive force supplied by the motor,  $\eta_t$  denotes the transmission efficiency and  $r$  is the radius of the wheel. Moreover,  $T(t) \geq 0$  and  $T(t) < 0$  represent the traction and braking processes, respectively. Based on the experimental data of PD18 provided by Protean Electric Inc, the efficiency map of the employed motor is established as a look-up table. In order to improve computational efficiency and enable real-time computation of the proposed algorithm, we replace the method of obtaining power  $P(t)$  through table lookup with a fitted model for the in-wheel motor. We approximate  $P(t)$  in Fig. 2 by a regression technique, and a quadratic fitted model of  $T(t)$  and  $v(t)$  depicted in Fig. 2 can be written as follows:

$$P(t) = aT^2(t)v(t) + bT(t)v(t) + cv(t) \quad (2)$$

where  $a, b, c$  are fitting parameters obtained through the nonlinear least squares method. As a result, the coefficient of determination  $R^2$  exceeds 0.99. Upon transformation from the time domain to the spatial domain, the resultant expression becomes:

$$\int P(t)dt = \int \frac{P(s)}{v(s)}ds = \int (aT^2(s) + bT(s) + c)ds \quad (3)$$

which follows a quadratic form.

## 2.2. Dynamic modeling in spatial domain

First, we briefly review the time-domain dynamic models as follows:

$$\dot{p}(t) = v(t), \quad (4a)$$

$$\dot{v}(t) = \frac{1}{m} \left( 2\frac{\eta_t}{r} T(t) - C_d v^2(t) - mg \sin \theta(t) - mgC_f \cos \theta(t) \right) \quad (4b)$$

where  $p$ ,  $v$  and  $T$  denote the position, velocity, and torque of the CAV, respectively.  $m$  is the vehicle mass and  $g$  is the gravity constant.  $\frac{\eta_t}{r} T$  represents the driving/braking force on tires where  $\eta_t$  is the transmission efficiency of the powertrain and  $r$  is the tire radius.  $C_d v^2$  is the air-drag force and  $C_d$  is the aerodynamic drag coefficient.  $\theta$  is the road slope, leading to gradient force  $mg \sin \theta(t)$ .  $C_f$  is the rolling resistance coefficient, and  $mgC_f \cos \theta$  represents the tire friction.

Next, to incorporate accurate space-dependent coefficients (such as rolling resistance, road slopes, etc.), time-domain models (4) are reformulated into a convex structure in the spatial domain via

$$\frac{d}{ds} = \frac{1}{v} \frac{d}{dt} \quad (5)$$

Herein, the independent variable is transformed from time  $t$  to traveled distance  $s$ ,

$$\frac{d}{ds} \Delta t(s) = \frac{1}{\sqrt{2E(s)/m}} - \frac{1}{\sqrt{2E_0(s)/m_0}}, \quad (6a)$$

$$\begin{aligned} \frac{d}{ds} E(s) = & 2\frac{\eta_t}{r} T(s) - 2\frac{C_d}{m} E(s) \\ & - mg \sin \theta(s) - mgC_f \cos \theta(s) \end{aligned} \quad (6b)$$

where  $\Delta t = t(s) - t_0(s)$  and  $E(s) = \frac{1}{2}mv^2(s)$  denote the time headway and kinetic energy, respectively.  $t_0$  and  $E_0$  are the corresponding states of the preceding vehicle. Furthermore, it is noted that  $E(s) > 0, \forall s$  to avoid the singularity.

In order to improve the computation efficiency, vehicle dynamic models (6) are reformulated into a convex form:

$$\frac{d}{ds} \Delta t(s) = \xi(s) - \frac{1}{v_0(s)}, \quad (7a)$$

$$\begin{aligned} \frac{d}{ds} E(s) = & 2\frac{\eta_t}{r} T(s) - 2\frac{C_d}{m} E(s) \\ & - mg \sin \theta(s) - mgC_f \cos \theta(s), \end{aligned} \quad (7b)$$

$$\xi(s) \geq \frac{1}{\sqrt{2E(s)/m}} \quad (7c)$$

where  $v_0$  is the velocity of the preceding vehicle, and  $\xi(s)$  is a fictitious control input that is designed for system convexification.

## 3. Learning-based ecological car-following control framework

In this section, we introduce the car-following strategy framework illustrated in Fig. 3. Multi-source traffic-related data, including micro vehicle speed, external information, and macro traffic flow, are collected and preprocessed to train the proposed MFST, which is mainly composed of an ensemble of diverse Transformer variants. At each sampling interval, the forecasted speed profile of the preceding vehicle can be derived based on the measured state and the well-trained MFST. This resultant profile  $\hat{E}_0(\cdot | s)$  is then integrated into a robust learning-in-the-loop MPC, which handles the discrepancies between the actual state and the predicted state. The proposed strategy is capable of offering real-time solutions while considering a trade-off among tracking performance, energy consumption, computation efficiency and driver comfort. Further details can be found in the subsequent subsections.

### 3.1. Macro-micro fused spatial-temporal transformer

The micro speed of CAV is significantly affected by the macro traffic flow, which exhibits complex temporal and spatial dependencies. To improve the accuracy of traffic speed prediction, the speed of the

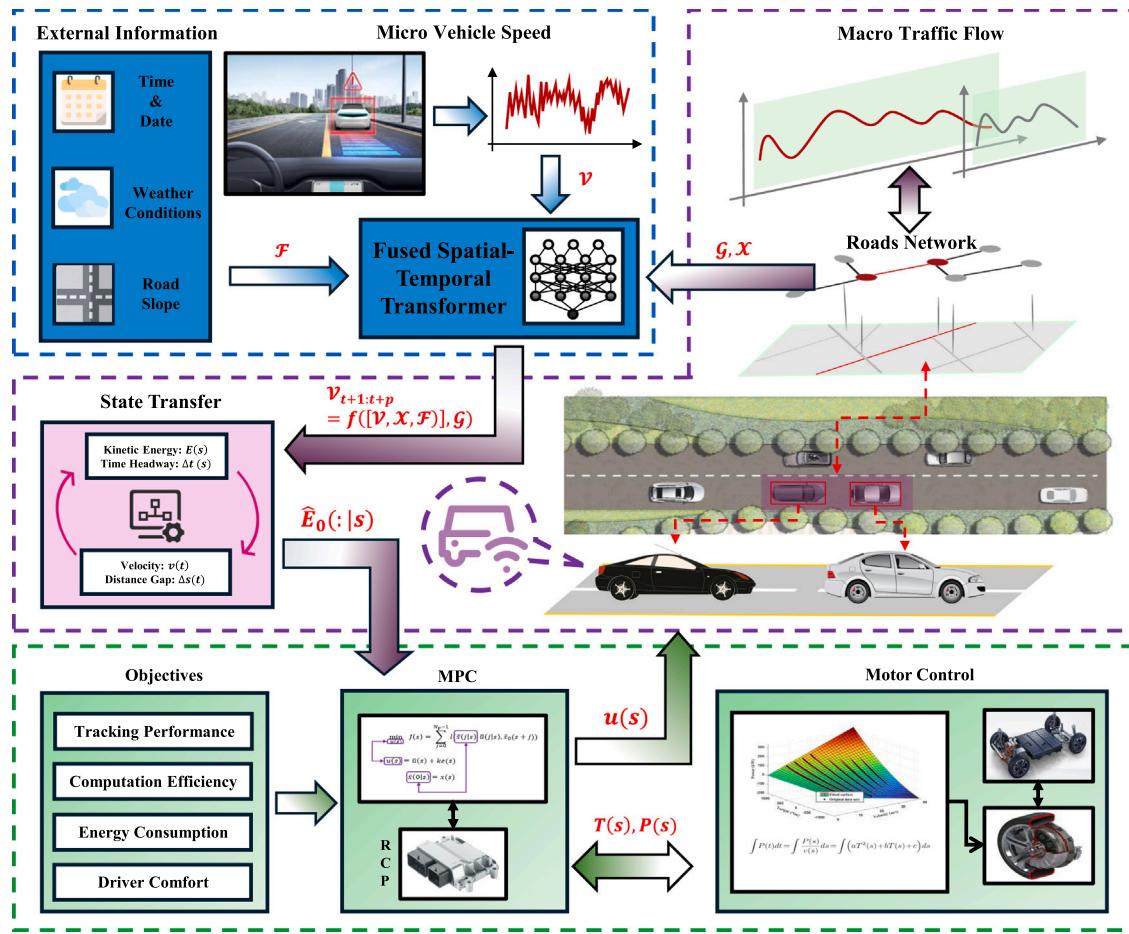


Fig. 3. Schematic framework of the proposed LEC approach.

preceding vehicle and three other types of data are included, namely the topological structure of the road network, macro traffic conditions, and time-related weather factors. To ensure clarity, we can divide long roads, such as expressways, into equal sections. This allows for consideration of traffic conditions at each location. The connections between each section are then treated as nodes. Let  $\mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathcal{A})$  denotes the graph, where  $\mathcal{S} \in \mathbb{R}^N$  is the set of nodes, and  $N$  is the number of nodes;  $\mathcal{E}$  is the edge set;  $\mathcal{A} \in \mathbb{R}^{N \times N}$  is the adjacent matrix of the road network. The macro traffic conditions are denoted as  $\mathcal{X} \in \mathbb{R}^{L \times N \times D}$ , where  $L$  is the length of historical time steps and  $D$  is the number of traffic attribute features, such as average speed, volume, and density. It should be noted that even though the adjacent matrix is symmetric, the traffic conditions have direction, therefore, we use the entry  $a_{i,j}$  in  $\mathcal{A}$  to denote the direction from node  $i$  to node  $j$ . The external factors, such as road slope, date, weather, etc., are defined as  $\mathcal{F} \in \mathbb{R}^{L \times M}$ , where  $M$  is the number of external factors.

The main goal is to find a function  $f(\cdot)$  that predicts the preceding vehicle speed  $\mathcal{V}_{t+1:t+p} = [v_{t+1}, v_{t+2}, \dots, v_{t+p}]$  since  $t$ th time step based on historical  $h$  time steps, where  $p$  denotes the future time steps. Thus, the speed forecasting task can be formulated as follows:

$$\mathcal{V}_{t+1:t+p} = f([\mathcal{V}, \mathcal{X}, \mathcal{F}]_{t-h+1:t}, \mathcal{G}) \quad (8)$$

In this study, a spatial-temporal deep learning model MFST is developed to predict the speed of the preceding vehicle based on micro and macro traffic data. The micro-data includes the historical speed of the preceding vehicle and external factors, and the macro-data includes traffic flow conditions and road network graph data. We consider the external factors as micro-data because they include the road number (i.e. the edge number in the graph) on which the preceding vehicle

was traveling. To address the long-range temporal correlations, a Transformer is introduced, as it has demonstrated superior performance in predicting sequential data. Additionally, recognizing the significance of node information and topological structure, we employ Graph Convolutional Networks (GCN) to capture spatial relationships [34]. As illustrated in Fig. 4, the proposed MFST comprises four modules: the Input module, Temporal Transformer (TT), Spatial Transformer (ST), and Spatial-Temporal Transformer (STT).

(1) The function of the input module is to receive and pre-process data. The historical preceding vehicle speed and external factors are fed into a Long Short-term Memory (LSTM) model, which is then embedded by a fully connected layer. The external factors include the number of road segments where the preceding vehicle. Simultaneously, the macro traffic conditions are input into a GCN layer along with the adjacency matrix, followed by a fully connected layer. Through this module, the temporal and spatial features are preliminary embedded, which can be formulated as follows:

$$\mathbf{H}^{\text{temp}} = \text{Linear}(\text{Concat}(LSTM(\mathcal{P}), LSTM(v))) \quad (9)$$

$$\mathbf{H}^{\text{spat}} = \text{Linear}(GCN(\mathcal{X}, \mathcal{A})) \quad (10)$$

where  $\mathbf{H}^{\text{temp}}$  and  $\mathbf{H}^{\text{spat}}$  are pre-processed temporal and spatial hidden states, respectively.

(2) TT and ST are designed to capture temporal and spatial correlations, respectively. Before being input into the transformer modules, the temporal and spatial hidden states are processed by Positional Encoding to capture the sequence information. Then, both modules apply a multi-head attention mechanism (MAH), feed-forward layer (FF), and Add & Norm layer.

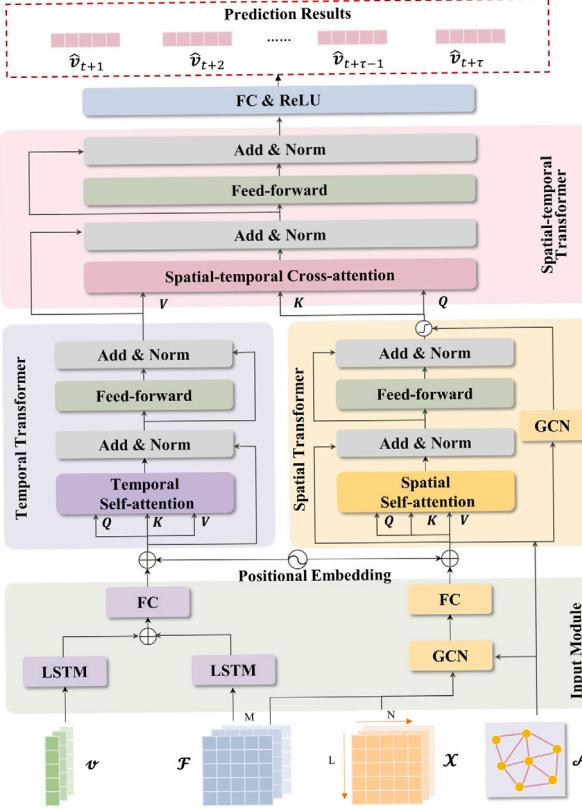


Fig. 4. Framework for prediction model.

The attention mechanism consists of multiple parallel scaled dot-product attention heads. For each head, a linear layer is applied to produce a 3-tuple of keys  $\mathbf{K}$ , queries  $\mathbf{Q}$ , and values  $\mathbf{V}$  for inputs. Then the attention scores can be calculated as (11). In the transformer, Multi-Head Attention (MHA) enables the model to capture different aspects of the input sequence simultaneously, enhancing its ability to learn complex patterns and dependencies.

$$\text{Atten} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_K}}\right)\mathbf{V} \quad (11)$$

$$\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{Atten}_1, \dots, \text{Atten}_i)W^O \quad (12)$$

where  $d_K$  is the dimension of the key,  $\text{Atten}_i$  is the  $i$ th attention mechanism in MHA, and  $W^O$  is the learnable parameter for the output layer.

The FF layer consists of two fully connected linear layers with a ReLU activation in between. Add & Norm (AN) layer functions as the connection, which applies Layer normalization and residual connections. Therefore, the procedure can be formulated as follows:

$$\text{FFN}(\mathbf{H}_i) = \text{ReLU}(\text{Linear}(\mathbf{H}_i)W^F + b_i^F) \quad (13)$$

$$\text{AN}(\mathbf{H}_i) = \text{LayerNorm}(\mathbf{H}_i + \text{Sublayer}(\mathbf{H}_i)) \quad (14)$$

where  $\mathbf{H}_i$  is the output of a certain hidden layer, weights  $W^F$  and biases  $b_i^F$  are learnable parameters,  $\text{LayerNorm}(\cdot)$  denotes the layer normalization procedure, and  $\text{Sublayer}(\cdot)$  denotes the attention mechanism or FF layer.

Therefore, the procedure of Transformer in TT and ST can be formulated as follows:

$$\mathbf{H}_{in} = \text{PE}(\mathbf{H}) \quad (15)$$

$$\mathbf{H}_{mid} = \text{AN}(\text{MHA}(\mathbf{H}_{in}, \mathbf{H}_{in}, \mathbf{H}_{in})) \quad (16)$$

$$\mathbf{H}_{out} = \text{AN}(\text{FFN}(\mathbf{H}_{mid})) \quad (17)$$

To balance the spatial features learned by the transformer and the GCN, an additional gate mechanism proposed by [35] is applied in ST before output, which can be formulated as follows:

$$\zeta = \text{sigmoid}(\text{Linear}(\mathbf{H}_{out}) + \text{Linear}(\text{GCN}(\mathbf{H}_{in}, \mathcal{A}))) \quad (18)$$

$$\mathbf{H}_{out} = \zeta \mathbf{H}_{out} + (1 - \zeta) \text{GCN}(\mathbf{H}_{in}) \quad (19)$$

(3) STT is designed to merge spatial and temporal characteristics. The structure of STT is similar to TT except that the self-attention mechanism is substituted by the cross-attention mechanism. Since the dimension of spatial hidden states is different from that of temporal hidden states, we first extract the target macro states from the spatial hidden states according to the road number. Specifically, for the road segment  $q$  from node  $i$  and  $j$ , we select the  $i$ th row and  $j$ th column of hidden states and add them together, then input them with temporal hidden states into the cross-attention mechanism. The formulations are shown as follows:

$$\mathbf{H}_q^{spat} = \text{Reshape}(\mathbf{H}[i]^{spat} + \mathbf{H}[j]^{spat}{}^T) \quad (20)$$

$$\mathbf{H}_{mid}^{fuse} = \text{AN}(\text{MHA}(\mathbf{H}^{temp}, \mathbf{H}_q^{spat}, \mathbf{H}_q^{spat})) \quad (21)$$

where  $\mathbf{H}_q^{spat}$  denotes the spatial hidden states for road  $q$ ,  $\mathbf{H}^{temp}$  denotes the hidden states generated by TT, respectively.  $\text{Reshape}(\cdot)$  denotes the function to rearrange the dimension to be the same as  $\mathbf{H}^{temp}$ .

Finally, the hidden states generated by STT are input to the output layer, which consists of fully connected layers with ReLU activation functions, then we can obtain the multiple-step predicted speed for preceding vehicles.

### 3.2. Robust learning-in-the-loop MPC

We first discretize and reformulate the dynamics (7), taking into account system disturbance  $d(s) \in \mathbb{D}$  for further introduction of the robust MPC as follows:

$$x(s+1) = Ax(s) + Bu(s) + Y(s) + d(s) \quad (22)$$

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 - \frac{2C_d}{m}\Delta s \end{bmatrix}, \quad B = \begin{bmatrix} \Delta s & 0 \\ 0 & 2\frac{\eta_f}{r}\Delta s \end{bmatrix},$$

$$Y(s) = \begin{bmatrix} -1/\left(\sqrt{2\hat{E}_0(s)/m_0}\right)\Delta s \\ -mg\left(\sin\theta(s) + C_f \cos\theta(s)\right)\Delta s \end{bmatrix}.$$

where  $s = 0, 1, \dots, \bar{s}$ ,  $\Delta s \in \mathbb{R}_{>0}$  denotes the sampling distance interval, and  $\bar{s}\Delta s$  represents the total length of the mission. The state vector  $x(s)$  is determined as  $x(s) = [\Delta t(s) \ E(s)]^T$ , and  $x(s) \in \mathbb{X}$  represents the state constraint as:

$$\mathbb{X} = \left\{ x(s) \mid \Delta t_{\min} \leq \Delta t(s) \leq \Delta t_{\max}, \frac{1}{2}mV_{\min}^2 \leq E(s) \leq \frac{1}{2}mV_{\max}^2 \right\} \quad (23)$$

where  $\{V_{\min}, V_{\max}\}$  are the speed limit of CAV. Similarly,  $u(s) = [\xi(s) \ T(s)]^T$  denotes the control input, where  $\xi(s)$  is a fictitious input designed for system convexification and  $T(s)$  represents the input torque. The control input  $u(s) \in \mathbb{U}$  is subject to the following constraint:

$$\mathbb{U} = \left\{ u(s) \mid T_{\min} \leq T(s) \leq T_{\max} \right\} \quad (24)$$

where  $\{T_{\min}, T_{\max}\}$  are the bounds on the maximal braking/driving torque

Next, the system disturbance  $d(s)$  is specified, where  $d_{[1,1]}(s)$  denotes the mismatch between  $1/\left(\sqrt{2E_0(s)/m_0}\right)\Delta s$  and  $1/\left(\sqrt{2\hat{E}_0(s)/m_0}\right)\Delta s$ , and  $d_{[2,1]}(s)$  accounts for the modeling uncertainties caused by parameter induction (e.g.  $C_d$ ,  $C_f$ , etc.). Following this,  $\mathbb{D}$  is defined as  $\mathbb{D} = \{d(s) \in \mathbb{R}^2 \mid \|d(s)\|_\infty \leq \bar{d} \in \mathbb{R}_{>0}\}$ , which is a compact convex set.

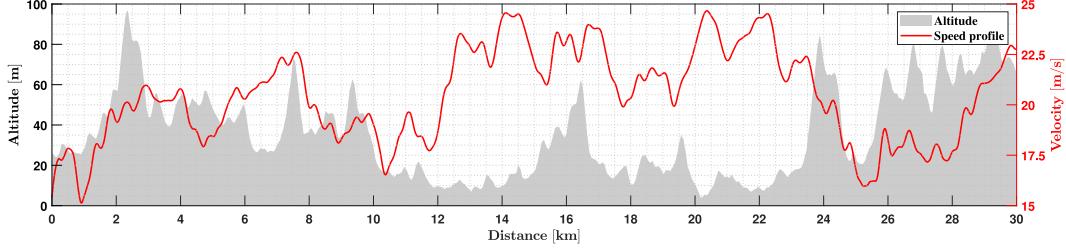


Fig. 5. Velocity profile and road slope angle.

Next, the nominal system corresponding to the actual dynamics (22) is presented as follows:

$$\bar{x}(s+1) = A\bar{x}(s) + B\bar{u}(s) + Y(s) \quad (25)$$

where  $\bar{x}(s) = [\Delta\bar{t}(s) \ \bar{E}(s)]^\top$  denotes the nominal state and  $\bar{u}(s) = [\bar{\xi}(s) \ \bar{T}(s)]^\top$  denotes the nominal control input, respectively. Herein, we define the discrepancy  $e(s)$  between the actual state  $x(s)$  and the nominal state  $\bar{x}(s)$  as

$$e(s) = x(s) - \bar{x}(s), \quad (26)$$

and the control law can be constructed by

$$u(s) = \bar{u}(s) + K e(s). \quad (27)$$

Based on (22), (25), (26) and (27), the dynamics of  $e(s)$  can be governed by:

$$e(s+1) = (A + BK)e(s) + d(s). \quad (28)$$

Then,  $\mathbb{E}$  accounts for the robust invariant set of  $e(s)$ , which can be obtained by:

$$\mathbb{E} = \bigoplus_{s=0}^{\infty} (A + BK)^s \mathbb{D} \quad (29)$$

Therefore, if the nominal state  $\bar{x}(s)$  and the nominal input  $\bar{u}(s)$  satisfy  $\bar{x}(s) \in \mathbb{X} \ominus \mathbb{E}$ ,

$$\bar{u}(s) \in \mathbb{U} \ominus K\mathbb{E}, \quad (30)$$

the satisfaction of  $x(s) \in \mathbb{X}$  and  $u(s) \in \mathbb{U}$  can be guaranteed. Additionally, the tube size is designed according to the varying forecasted speed of the preceding vehicle, aiming to achieve less conservative results. Before introducing the design process, we first present the concept of the reachable range of the state  $E_0(s) \in \{\underline{E}(\hat{E}_0(s)), \bar{E}(\hat{E}_0(s))\}$  over  $N_p$  steps, where  $N_p$  is the length of the predictive horizon. These ranges can be determined by applying minimum and maximum input torque to the system dynamic (22) over the prediction horizon, as shown below:

$$\underline{E}(\hat{E}_0(s)) = (1 - \frac{2C_{d,0}}{m_0} \Delta s)^{N_p-1} \hat{E}_0(s) + \sum_{j=0}^{N_p-2} (1 - \frac{2C_{d,0}}{m_0})^j \left( 2\frac{\eta_{t,0}}{r_0} T_{0,\min} - m_0 g (\sin \theta(s+j) + C_f \cos \theta(s+j)) \right) \Delta s \quad (31)$$

and

$$\bar{E}(\hat{E}_0(s)) = (1 - \frac{2C_{d,0}}{m_0} \Delta s)^{N_p-1} \hat{E}_0(s) + \sum_{j=0}^{N_p-2} (1 - \frac{2C_{d,0}}{m_0})^j \left( 2\frac{\eta_{t,0}}{r_0} T_{0,\max} - m_0 g (\sin \theta(s+j) + C_f \cos \theta(s+j)) \right) \Delta s. \quad (32)$$

Following this, the tube size can be precomputed offline as follows:

For the sake of the brevity, the tightened sets are rewritten into

$$\bar{x}(s) \in \bar{\mathbb{X}}(k), \quad \bar{u}(s) \in \bar{\mathbb{U}} \quad (33)$$

#### Algorithm 1 Offline design of the tube size

- 1: Divide  $\{E_{0,\min}, E_{0,\max}\}$  into  $n$  parts and define  $\delta = \frac{E_{0,\max} - E_{0,\min}}{n}$ ;
- 2: Each subset of  $\{E_{0,\min}, E_{0,\max}\}$  can be represented by  $\{\underline{E}(E_{0,\min} + k\delta), \bar{E}(E_{0,\min} + k\delta)\}, k \in \{0, 1, 2, \dots, n-1\}$ ;
- 3: Calculate  $\bar{\mathbb{X}}(k) = \mathbb{X}(k) \ominus \mathbb{E}$  for each subset through (22) to (30);

In summary, the local MPC problem  $\mathcal{P}(s)$  at step  $s$  is formulated as follows:

$$\mathcal{P}(s) :$$

$$\min_{\bar{u}} J(s) = \sum_{j=0}^{N_p-1} l(\bar{x}(j|s), \bar{u}(j|s), \hat{x}_0(s+j)) \quad (34a)$$

s.t.

$$\bar{x}(j+1|s) = A\bar{x}(j|s) + B\bar{u}(j|s) + Y(j|s) \quad (34b)$$

$$\bar{x}(j|s) \in \bar{\mathbb{X}}(k) \quad (34c)$$

$$\bar{u}(j|s) \in \bar{\mathbb{U}} \quad (34d)$$

$$(\bar{x}(j|s), \bar{u}(j|s)) \in \bar{\mathbb{X}}(k) \times \bar{\mathbb{U}} \quad (34e)$$

$$\bar{x}(0|s) = x(s) \quad (34f)$$

where the stage cost function  $l(\bar{x}(j|s), \bar{u}(j|s), \hat{x}_0(s+j))$  in (34a) can be specified:

$$l = w_1 |\Delta\bar{t}(j|s) - \Delta| + w_2 |\bar{E}(j|s) - \frac{m}{m_0} \hat{E}_0(s+j)| + w_3 |T(j|s) - h(j|s)| + w_4 P(j|s) + w_5 \xi(j|s) \quad (35)$$

$w_1 |\Delta\bar{t}(j|s) - \Delta|$  and  $w_2 |\bar{E}(j|s) - \frac{m}{m_0} \hat{E}_0(j|s)|$  are configured to preserve the car-following performance;  $w_3 |T(j|s) - h(j|s)|$  is set to improve passenger comfort, where  $h(j|s)$  denotes the equilibrium input torque as

$$h(j|s) = \frac{r}{\eta_t} \left( \frac{2C_d}{m} \hat{E}_0(j|s) + mg (\sin \theta(s+j) + C_f \cos \theta(s+j)) \right) \quad (36)$$

$w_4 P(j|s)$  penalizes energy consumption, where  $P(j|s)$  represents the power of in-wheel motor based on (2); The final term  $w_5 \xi(j|s)$  ensures the validity of the control solution.

Dynamic Eq. (34b) represents the nominal system introduced in (25); Constraints (34c) and (34d) are robust state and input constraints, corresponding to the tube size design; (34e) denotes a coupled constraint which enforces system convexity as follows:

$$\bar{\mathbb{X}}(k) \times \bar{\mathbb{U}} = \left\{ (\bar{E}(j|s), \bar{\xi}(j|s)) \mid \bar{\xi}(j|s) \geq \frac{1}{\sqrt{2\hat{E}(s+j)/m}} \right\} \quad (37)$$

The initial condition of the MPC is specified by (34f) which employs the actual state. Overall, the proposed robust control approach is summarized in Algorithm 2.

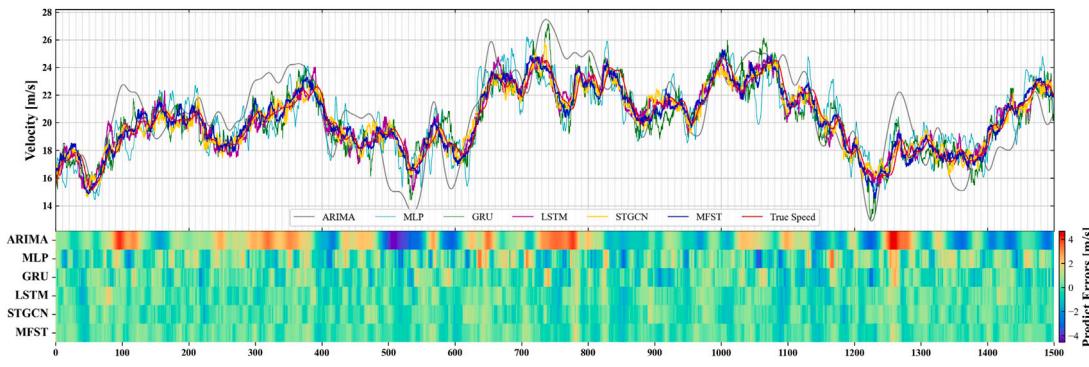
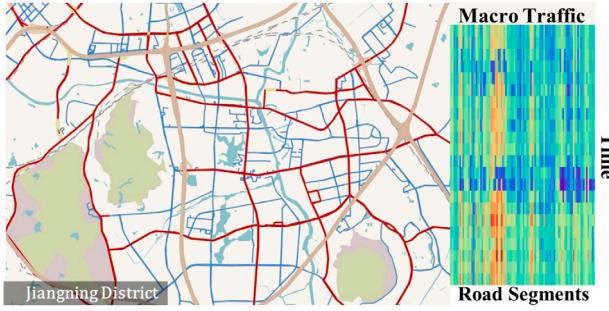


Fig. 6. Prediction results and predicted errors.



**Fig. 7.** Map of the study area and macro traffic speed data: The data was collected every second from 150 road segments over one month. The subplot on the right illustrates the spatiotemporal distribution of traffic speed, with the horizontal axis representing individual road segments, the vertical axis representing time, and the color gradient from blue to red indicating an increase in speed from low to high.

**Algorithm 2** The robust learning-in-the-loop MPC approach for ego vehicle

#### Offline:

- Find the control gain  $K$  via pole-placement method;
- Select suitable weighting parameters  $w_1, w_2, w_3, w_4, w_5$  and the prediction horizon  $N_p$  for the OCP  $\mathcal{P}$ ;
- Calculate  $\tilde{\mathbb{X}}(k), k \in \{0, 1, 2, \dots, n-1\}$  via Algorithm 1;

#### Online:

- while**  $0 \leq s < \bar{s}$  **do**
- Measure current states  $x_0(s)$  and  $x(s)$ ;
- Receive  $\hat{x}_0(:|s)$  from the preceding vehicle;
- Check to which subset  $\{\underline{E}(E_{0,\min} + k\delta), \bar{E}(E_{0,\min} + k\delta)\} \hat{E}_0(s)\}$  belongs;
- Invoke corresponding  $\tilde{\mathbb{X}}(k)$ ;
- Set the initial condition (34f) of the MPC problem  $\mathcal{P}(s)$  based on the measured values  $x_0(s)$  and  $x(s)$ ;
- Generate  $\hat{x}(:|s)$  through the speed predictor;
- Solve OCP  $\mathcal{P}(s)$  and obtain  $\tilde{u}(:|s)$  and  $\tilde{x}(:|s)$ ;
- Apply the control action  $u(s) = \tilde{u}(0|s) + Ke(s)$  with  $e(s) = x(s) - \tilde{x}(0|s)$  to the actual system (22);
- $s \leftarrow s + 1$ ;
- end while**

## 4. Simulation

In this section, we showcase the performance of the LEC strategy, focusing on key aspects such as prediction accuracy, energy consumption, and computational efficiency. Numerical simulations are conducted to highlight the advantages of the proposed strategy compared to several state-of-the-art algorithms, which are introduced in Section 4.1.

**Table 1**  
Simulation parameters.

Description	Symbols	Values
Ego vehicle mass	$m$	2000 kg
Motor transmission efficiency	$\eta_t$	0.95
Tire radius	$r$	0.36 m
Air-drag coefficient	$C_d$	0.28
Tire rolling resistance coefficient	$C_f$	0.015
Gravity constant	$g$	9.8 N/kg
Motor fitting parameters	$a; b; c$	0.01078; 22.23; 67.63
Maximum torque	$T_{\max}$	1225 N m
Minimal torque	$T_{\min}$	-1225 N m
Predictive horizon Length	$N_p$	10
Sampling distance interval	$\Delta s$	2 m
Desired time headway	$\Delta t$	1 s
Minimum time headway	$\Delta t_{\min}$	0.5 s
Maximum time headway	$\Delta t_{\max}$	1.5 s

**Table 2**  
Results of prediction performance.

	MSE	MAE	RMSE	R2
ARIMA	74.96	8.03	8.66	0.06
MLP	30.26	4.79	5.50	0.15
GRU	22.85	4.04	4.78	0.17
LSTM	10.41	2.73	3.23	0.34
STGCN	7.84	2.38	2.80	0.40
MFST	<b>4.98</b>	<b>1.89</b>	<b>2.23</b>	<b>0.51</b>

Relevant parameters for simulations and the on-road experiment are summarized in Table 1.

### 4.1. Benchmarks

Comparisons with corresponding algorithms are in terms of prediction accuracy, computation efficiency, and energy consumption, which are introduced as follows: Regarding prediction accuracy, we apply several baseline models for comparison, including ARIMA, MLP, LSTM [36], GRU [37], and STGCN [38]. Considering the computation time of the LEC strategy, we first set a nonlinear car-following approach [39] as a benchmark and extend the comparison with a state-of-the-art real-time algorithm [40]. Concerning energy consumption, the proposed LEC algorithm initially compares with an Eco-driving benchmark [41] with a feedback mechanism and further compares with an IDM-based method [42].

### 4.2. Prediction accuracy

The data for this study was collected from the road network in Jiangning District, Nanjing, China, during the period of June 2020, as shown in Fig. 7. The macro traffic data are collected by vehicle monitoring and tracking systems, which are counted as 1-second granularity and interpolated at 0.5 km intervals. The adjacent matrix is

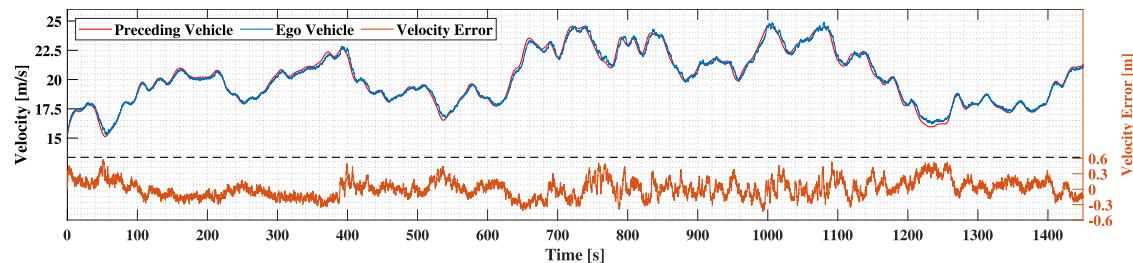


Fig. 8. Velocity and velocity error.

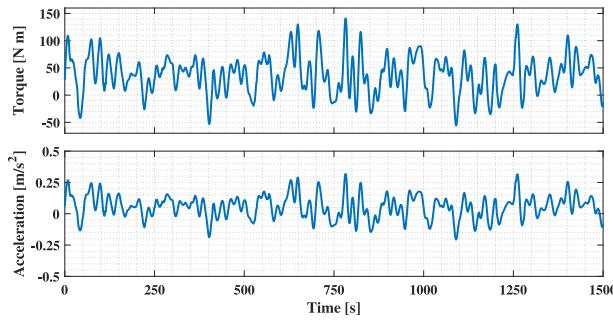


Fig. 9. Input torque and acceleration.

generated according to the road network within the study area. The micro-level vehicle speed data was obtained from multiple instances of vehicle-following recordings. Weather data collected from Weather Underground is also processed to match the dimensions of the micro and macro traffic data.

In this study, each time step represents 1 s, and 10-second speed data are input as historical information to predict the future two-step speed. We divided the dataset into training and test sets in a ratio of 8 to 2 and used it for model training. To validate the advantage of MFST, the prediction performance is compared based on various error metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-square ( $R^2$ ). The results are summarized in Table 2. It can be seen that our proposed method obtains the best prediction performance. As car-following control requires finer time granularity, 0.1 s in this study, we applied linear interpolation and sliding median method to process the predicted speed. To further validate our predictor, we predict the preceding vehicle speed recorded from a 30 km trip, and the raw speed data and road slope are shown in Fig. 5. The predicted results are shown in Fig. 6, in which the lines illustrate the predicted results of the proposed model along with that of the baseline models and ground truth, and the heatmap represents the predicted errors (i.e. the difference between predicted and true values). It is observed that our model presents a better temporal pattern and fewer errors among all selected models.

#### 4.3. Car-following performance

The car-following performance of the proposed strategy is investigated in this subsection, and the results are demonstrated in Fig. 8. Specifically, the red line denotes the speed profile of the preceding vehicle (The same as depicted in Fig. 5), while the blue line illustrates the tracking performance of the ego vehicle. Considering the data provided above, the tracking error, indicated in orange, does not exceed  $\pm 0.6$  m/s at its maximum. Additionally, the input torque and corresponding acceleration of the ego vehicle are presented in Fig. 9. By employing the control objective  $w_3|T(j|s) - h(j|s)|$ , the acceleration remains within  $\pm 0.31$  m/s<sup>2</sup>, aligning with the ‘excellent acceleration’ criterion outlined in [43]. Additionally, the effectiveness of robust

control is also studied. In Fig. 11, it can be observed that if a nominal MPC without the tightened constraint (34c) is taken into account, the safety constraint is violated at the position 576 m, which could cause a collision with its preceding vehicle. Note that the scope of safety constraint is changing because we present the distance headway instead of the constant time headway here.

#### 4.4. Energy consumption

The energy consumption of the proposed strategy is studied in this subsection, and its advantages are further emphasized through a comparison with two additional methods. Simulation results are shown in Fig. 10, which employs a feedback control mechanism [41] in orange and another IDM-based strategy [42] in yellow. In detail, owing to the feedback mechanism, the benchmark method places a higher emphasis on car-following performance, thereby at the expense of energy consumption. On the other hand, the IDM strategy is based on imitating the behavior of human drivers, favoring gentle speed changes. Compared to the benchmark, it exhibits greater energy efficiency; however, it is still inferior to the proposed LEC strategy. This is attributed to maximizing the penalty weight assigned to energy consumption in the optimization objective of the proposed method while satisfying all constraints. Furthermore, the total energy consumption among the three strategies is summarized, where the LEC strategy saves 9.31% of energy in comparison to the IDM strategy, and a 13.63% energy savings to the feedback control strategy (FBC). In detail, energy consumption can be categorized into four types: powertrain loss, tire friction loss, air-drag loss, and potential energy loss. The proposed strategy primarily saves energy in powertrain loss by reducing braking and keeping input torque and vehicle speed in the high-efficiency range.

#### 4.5. Computation efficiency

This subsection investigates the real-time computation capabilities of the proposed strategy. To underscore the computational efficiency of the LEC algorithm, a nonlinear benchmark [39] and a recently published real-time ecological ACC (REACC) algorithm [40] are set for comparison. Numerical simulation results are illustrated in Fig. 12, where LEC is in blue, the nonlinear benchmark is in yellow and REACC is in orange. Specifically, the average computation time of the proposed strategy is 0.006 s per step, with a maximum value of 0.01 s. Taking into account the length of the spatial sampling interval  $\Delta s = 2$  m and  $V_{\max} = 40$  m/s, the time required for traversing this distance is 0.05 s, meeting the real-time requirement. In comparison, the average time cost of the nonlinear benchmark is 0.491 s and the average computation time of the REACC is 0.081 s. The proposed LEC strategy achieves a computation speed 10 times faster than that of REACC, showcasing significantly enhanced real-time computational capabilities.

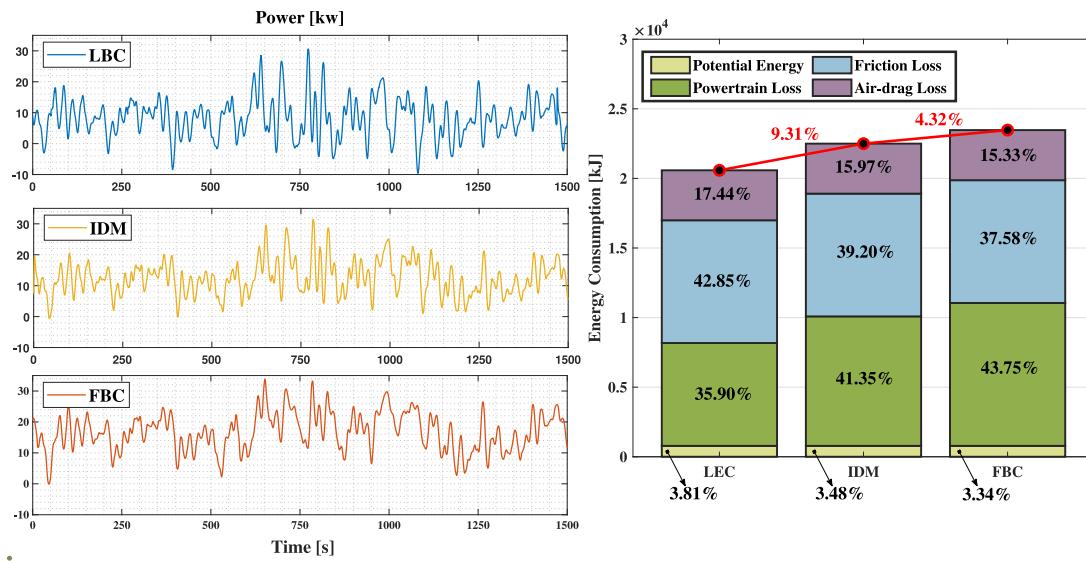


Fig. 10. Power and energy consumption.

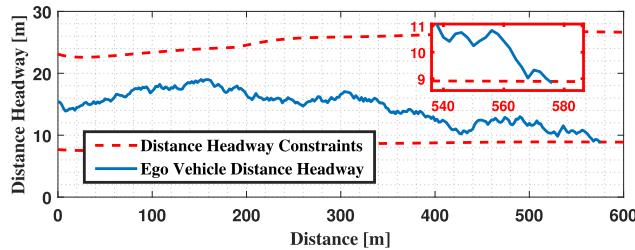


Fig. 11. Car-following performance without robust control.

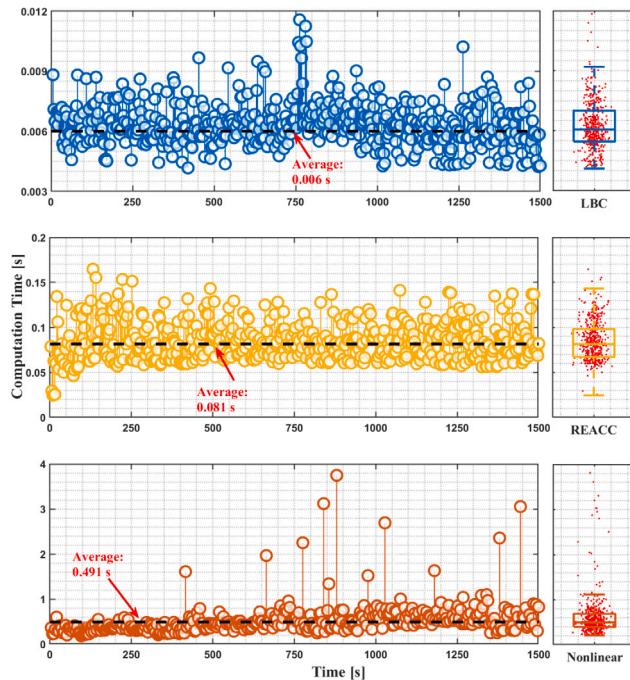


Fig. 12. Computation load of different methods: Top-LEC strategy; Middle-REACC algorithm; Bottom-nonlinear benchmark.

## 5. On-road vehicle experiment

In order to highlight the practical applicability of the proposed strategy, the on-road experiment is conducted. Due to traffic regulations, the ego vehicle only tracks the preceding vehicle for a distance of 3.5 km (Speed profile of the preceding vehicle is introduced in Section 4).

As shown in Fig. 13, a modified CAV based on a Chery BEV eQ1 [4] is employed for this experiment. Through integrating the equipped LIDAR, camera, and Millimeter-wave Radar (MWR), the surrounding environmental data can be fused and then transformed into the spatial domain. Further combined with the navigation module consisting of Global Positioning System (GPS) and Inertial Navigation System (INS), the reference trajectory and motion constraints are generated and applied to the proposed MPC-based controller. Subsequently, the input commands calculated by the built-in LEC algorithm are transmitted to the actuators of the tested CAV, including brake-by-wire and steering-by-wire (enclosed by the purple box in Fig. 13). The aforementioned experimental modules are built and connected together based on the Robotics Operating System (ROS), which is operating on Ubuntu 16.04 LTS.

The experimental results are presented in Fig. 14, recording the velocity changes, velocity error and spacing distance of the test vehicles. It is noted that the spacing distance (from the rear of the preceding vehicle to the front of the ego vehicle) increases gradually due to our control target of maintaining a constant time headway rather than a constant spacing distance. As the preceding vehicle accelerates, this spacing distance also increases.

## 6. Conclusion and future work

This paper addresses a car-following problem for connected and autonomous electric vehicles. Through a spatial-temporal transformer, micro and macro traffic information are fused for preceding vehicle speed prediction. Resulting errors are handled by a robust learning-in-the-loop MPC method in the control layer. Meanwhile, the receding horizon optimization problem is reformulated into a convex form, transforming from the time domain to the spatial domain, which improves the computation efficiency and enables real-time implementation. The advantages of the LEC strategy are highlighted in comparison with several state-of-the-art algorithms in terms of prediction accuracy, energy consumption, and computational efficiency. Moreover, an on-road vehicle experiment is conducted to confirm that the proposed strategy has the capability for practical applications.

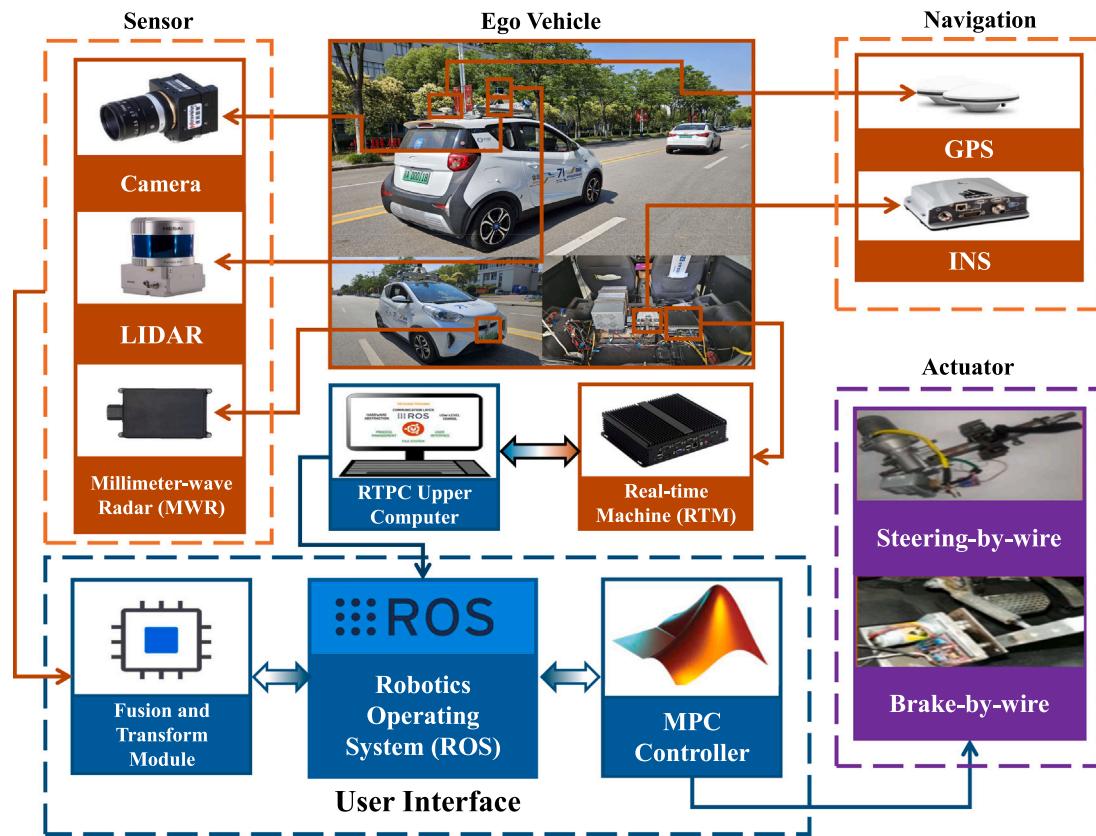


Fig. 13. On-road vehicle experiment.

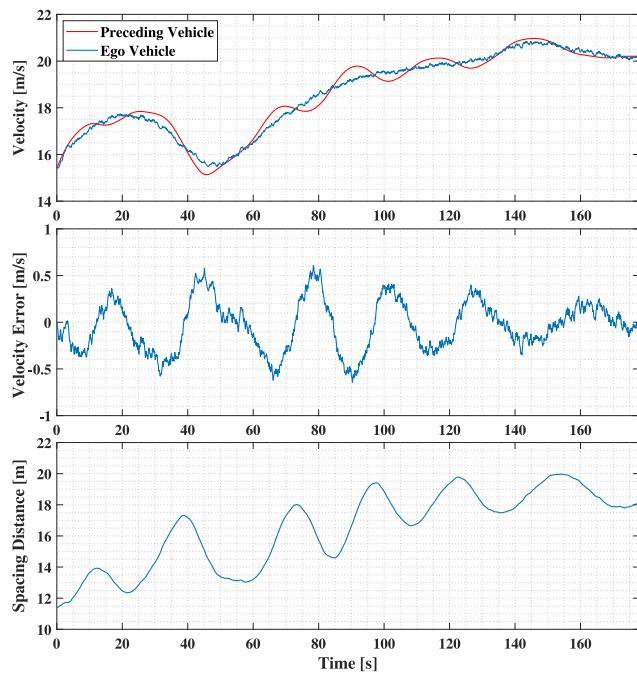


Fig. 14. The results of the VIL experiment: Top-Velocity changes; Middle-Velocity error; Bottom-Spacing distance.

In the future, the LEC strategy will be extended to urban scenarios with traffic lights. Additionally, turning and overtaking behaviors of the vehicle will be studied with consideration of lateral motion models.

#### CRediT authorship contribution statement

**Hao Sun:** Writing – original draft, Software, Methodology, Conceptualization. **Shuang Li:** Writing – review & editing, Software, Methodology, Data curation. **Bingbing Li:** Visualization, Validation, Resources. **Mingyang Chen:** Writing – review & editing, Visualization, Data curation. **Sunan Zhang:** Writing – review & editing, Visualization, Data curation. **Weichao Zhuang:** Writing – review & editing, Supervision, Resources, Formal analysis. **Guodong Yin:** Writing – review & editing, Supervision, Resources, Project administration. **Boli Chen:** Writing – review & editing, Supervision, Methodology, Funding acquisition.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Acknowledgments

This work has been supported by The Royal Society Grant, UK, IES\R2\212041 and the State Key Laboratory of Intelligent Green Vehicle and Mobility, China under Project No. KFZ2405.

#### Data availability

Data will be made available on request.

## References

- [1] Hill MK. Understanding environmental pollution. Cambridge University Press; 2020.
- [2] Kumar RR, Alok K. Adoption of electric vehicle: A literature review and prospects for sustainability. *J Clean Prod* 2020;253:119911.
- [3] Sierchula W, Bakker S, Maat K, Van Wee B. The influence of financial incentives and other socio-economic factors on electric vehicle adoption. *Energy Policy* 2014;68:183–94.
- [4] Li B, Zhuang W, Zhang H, Sun H, Liu H, Zhang J, Yin G, Chen B. Traffic-aware ecological cruising control for connected electric vehicle. *IEEE Trans Transp Electrific* 2023.
- [5] Li L, Wang X, Song J. Fuel consumption optimization for smart hybrid electric vehicle during a car-following process. *Mech Syst Signal Process* 2017;87:17–29.
- [6] Pan C, Huang A, Wang J, Chen L, Liang J, Zhou W, Wang L, Yang J. Energy-optimal adaptive cruise control strategy for electric vehicles based on model predictive control. *Energy* 2022;241:122793.
- [7] Pan X, Chen B, Evangelou SA. Optimal vehicle following strategy for joint velocity and energy management control of series hybrid electric vehicles. *IFAC-Pap* 2020;53(2):14161–6.
- [8] Yu S, Pan X, Georgiou A, Chen B, Jaimoukha IM, Evangelou SA. Robust model predictive control framework for energy-optimal adaptive cruise control of battery electric vehicles. In: 2022 European control conference. ECC, IEEE; 2022, p. 1728–33.
- [9] Hu B, Zhang S, Liu B. A hybrid algorithm combining data-driven and simulation-based reinforcement learning approaches to energy management of hybrid electric vehicles. *IEEE Trans Transp Electrification* 2023;10(1):1257–73.
- [10] Jia D, Chen H, Zheng Z, Watling D, Connors R, Gao J, Li Y. An enhanced predictive cruise control system design with data-driven traffic prediction. *IEEE Trans Intell Transp Syst* 2021;23(7):8170–83.
- [11] Sun C, Leng J, Sun F. A fast optimal speed planning system in arterial roads for intelligent and connected vehicles. *IEEE Internet Things J* 2022;9(20):20295–307.
- [12] Zhang S, Zhuang W, Li B, Li K, Xia T, Hu B. Integration of planning and deep reinforcement learning in speed and lane change decision-making for highway autonomous driving. *IEEE Trans Transp Electrific* 2024.
- [13] Amini MH, Kargarian A, Karabasoglu O. ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation. *Electr Power Syst Res* 2016;140:378–90.
- [14] 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(6):5309–20.
- [15] Jiang B, Fei Y. Vehicle speed prediction by two-level data driven models in vehicular networks. *IEEE Trans Intell Transp Syst* 2016;18(7):1793–801.
- [16] Shin J, Sunwoo M. Vehicle speed prediction using a Markov chain with speed constraints. *IEEE Trans Intell Transp Syst* 2018;20(9):3201–11.
- [17] Sun C, Hu X, Moura SJ, Sun F. Velocity predictors for predictive energy management in hybrid electric vehicles. *IEEE Trans Control Syst Technol* 2014;23(3):1197–204.
- [18] Sun C, Sun F, He H. Investigating adaptive-ECMS with velocity forecast ability for hybrid electric vehicles. *Appl Energy* 2017;185:1644–53.
- [19] Ye F, Hao P, Qi X, Wu G, Boriboonsomsin K, Barth MJ. Prediction-based eco-approach and departure at signalized intersections with speed forecasting on preceding vehicles. *IEEE Trans Intell Transp Syst* 2018;20(4):1378–89.
- [20] Lin X, Wang Z, Wu J. Energy management strategy based on velocity prediction using back propagation neural network for a plug-in fuel cell electric vehicle. *Int J Energy Res* 2021;45(2):2629–43.
- [21] Liu J, Chen Y, Zhan J, Shang F. An on-line energy management strategy based on trip condition prediction for commuter plug-in hybrid electric vehicles. *IEEE Trans Veh Technol* 2018;67(5):3767–81.
- [22] Yan M, Li M, He H, Peng J. Deep learning for vehicle speed prediction. *Energy Procedia* 2018;152:618–23.
- [23] Cheng Z, Chow M-Y, Jung D, Jeon J. A big data based deep learning approach for vehicle speed prediction. In: 2017 IEEE 26th international symposium on industrial electronics. ISIE, IEEE; 2017, p. 389–94.
- [24] Zhang Z, He H, Guo J, Han R. Velocity prediction and profile optimization based real-time energy management strategy for plug-in hybrid electric buses. *Appl Energy* 2020;280:116001.
- [25] Han S, Zhang F, Xi J, Ren Y, Xu S. Short-term vehicle speed prediction based on convolutional bidirectional lstm networks. In: 2019 IEEE intelligent transportation systems conference. ITSC, IEEE; 2019, p. 4055–60.
- [26] Gao B, Wan K, Chen Q, Wang Z, Li R, Jiang Y, Mei R, Luo Y, Li K. A review and outlook on predictive cruise control of vehicles and typical applications under cloud control system. *Mach Intell Res* 2023;20(5):614–39.
- [27] Liu C-Z, Li L, Chen X, Yong J-W, Cheng S, Dong H-L. An innovative adaptive cruise control method based on mixed  $H_2/H_\infty$  out-of-sequence measurement observer. *IEEE Trans Intell Transp Syst* 2021;23(6):5602–14.
- [28] Schmid R, Colaneri P. Mixed  $H_2 - H_\infty$  control for automated highway driving. *Mechatronics* 2019;57:63–72.
- [29] Naeem HMY, Bhatti AI, Mahmood A. Longitudinal cruise control of a car using sliding mode approach. In: 2019 international conference on electrical, communication, and computer engineering. IECCE, IEEE; 2019, p. 1–5.
- [30] Ren D, Zhang J, Zhang J. Sliding mode control for vehicle following with parametric uncertainty. *Electr Mach Control* 2010;14(1):73–8.
- [31] Zhang Y, Huang Y, Chen Z, Li G, Liu Y. A novel learning-based model predictive control strategy for plug-in hybrid electric vehicle. *IEEE Trans Transp Electrific* 2021;8(1):23–35.
- [32] Hou Z, Chu L, Guo Z, Hu J, Jiang J, Yang J, Chen Z, Zhang Y. A learning-and-tube-based robust model predictive control strategy for plug-in hybrid electric vehicle. *IEEE Trans Intell Veh* 2023.
- [33] Mayne DQ, Raković SV, Findeisen R, Allgöwer F. Robust output feedback model predictive control of constrained linear systems. *Automatica* 2006;42(7):1217–22.
- [34] Bruna J, Zaremba W, Szlam A, LeCun Y. Spectral networks and locally connected networks on graphs. 2014, arXiv:1312.6203.
- [35] Xu M, Dai W, Liu C, Gao X, Lin W, Qi G-J, Xiong H. Spatial-temporal transformer networks for traffic flow forecasting. 2020, arXiv preprint arXiv:2001.02908.
- [36] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8):1735–80.
- [37] Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. 2014, arXiv preprint arXiv:1406.1078.
- [38] Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. 2017, arXiv preprint arXiv:1709.04875.
- [39] Lopes DR, Evangelou SA. Energy savings from an eco-cooperative adaptive cruise control: a BEV platoon investigation. In: 2019 18th European control conference. ECC, IEEE; 2019, p. 4160–7.
- [40] Yu S, Pan X, Georgiou A, Chen B, Jaimoukha IM, Evangelou SA. A real-time robust ecological-adaptive cruise control strategy for battery electric vehicles. *IEEE Trans Transp Electrific* 2023. <http://dx.doi.org/10.1109/TTE.2023.3340670>, 1–1.
- [41] Li B, Zhuang W, Tang M, Li J. Hierarchical optimal control of a group of electric vehicles for maximizing energy efficiency and battery life. In: 2022 6th CAA international conference on vehicular control and intelligence. CVCI, 2022, p. 1–6. <http://dx.doi.org/10.1109/CVCI56766.2022.9964759>.
- [42] Treiber M, Hennecke A, Helbing D. Congested traffic states in empirical observations and microscopic simulations. *Phys Rev E* 2000;62(2):1805.
- [43] de Winkel KN, Irmak T, Happee R, Shyrokau B. Standards for passenger comfort in automated vehicles: Acceleration and jerk. *Appl Ergon* 2023;106:103881.