Comfortable and energy-efficient speed control of autonomous vehicles on rough pavements using deep reinforcement learning

Yuchuan Du\(^a,b\), Jing Chen\(^a\), Cong Zhao\(^a,*\), Chenglong Liu\(^a,b\), Feixiong Liao\(^c\), Ching-Yao Chan\(^d\)

\(^a\) Key Laboratory of Road and Traffic Engineering of the Ministry of Education, Tongji University, Shanghai 201804, China
\(^b\) Shanghai Engineering Research Center of Urban Infrastructure Renewal, Shanghai 200032, China
\(^c\) Urban Planning and Transportation Group, Eindhoven University of Technology, the Netherlands
\(^d\) California PATH, Institute of Transportation Studies, University of California, Berkeley, CA 94804, USA

Keywords:
Autonomous vehicle
Ride comfort
Energy efficiency
Deep reinforcement learning
Speed control

ABSTRACT

Rough pavements cause ride discomfort and energy inefficiency for road vehicles. Existing methods to address these problems are time-consuming and not adaptive to changing driving conditions on rough pavements. With the development of sensor and communication technologies, crowdsourced road and dynamic traffic information become available for enhancing driving performance, particularly addressing the discomfort and inefficiency issues by controlling driving speeds. This study proposes a speed control framework on rough pavements, envisioning the operation of autonomous vehicles based on the crowdsourced data. We suggest the concept of ‘maximum comfortable speed’ for representing the vertical ride comfort of oncoming roads. A deep reinforcement learning (DRL) algorithm is designed to learn comfortable and energy-efficient speed control strategies. The DRL-based speed control model is trained using real-world rough pavement data in Shanghai, China. The experimental results show that the vertical ride comfort, energy efficiency, and computation efficiency increase by 8.22%, 24.37%, and 94.38%, respectively, compared to an optimization-based speed control model. The results indicate that the proposed framework is effective for real-time speed controls of autonomous vehicles on rough pavements.

1. Introduction

Over the last decade, significant progress has been made in autonomous driving. Technological developments have led to advances in the modeling and analysis of environment perception, decision-making, and vehicle control to support autonomous driving in various scenarios (Chen et al., 2019; Zhao et al., 2021a; Zhao et al., 2021b). A challenging scenario for vehicle operations is driving on rough pavements, where occupants may suffer from bumpy experiences (Du et al., 2018). To improve driving performance on rough pavements, experienced drivers can adjust speeds according to the observed road information and the sense of vibration (Wu et al., 2020a). However, for autonomous vehicles (AVs), the lack of subjective judgment makes it difficult to adjust the speed control strategies. Rapid developments of sensing and communication technologies have brought great potentials for innovative solutions to address this problem.

* Corresponding author.
E-mail address: zhc@tongji.edu.cn (C. Zhao).

https://doi.org/10.1016/j.trc.2021.103489
Received 11 May 2021; Received in revised form 22 October 2021; Accepted 19 November 2021
Available online 8 December 2021
0968-090X/© 2021 Elsevier Ltd. All rights reserved.
For AVs, speed control is critical for achieving better driving performance (Di and Shi, 2021). Existing AV speed control methods can be classified into three main types: rule-based, optimization-based, and learning-based. Rule-based methods usually have strong assumptions about the environment (Zhu et al., 2018) and need much expert knowledge to establish rules (Huang et al., 2019). They cannot be applied to situations that are not well-defined. Optimization-based methods can solve optimal speed control problems in the entire driving trajectory (e.g., dynamic programming) or the prediction horizon (e.g., model predictive control (MPC)) (Hartmann et al., 2019; Ozatay et al., 2014). However, this type of method has limitations in applications. When dealing with a complex system or problem, it demands large computational budgets or even fails in finding the optimal solution (Lin et al., 2020; Zeng and Wang, 2018; Zhu et al., 2020). For the application of driving on rough pavements, although a solution can be found for driving on a specific road, it may not be adaptive to changing driving conditions (Hartmann et al., 2019). Learning-based methods can learn the speed control strategy from massive experience data in a changing environment. Such methods rely on neither predefined rules nor the modeling of complex systems and problems. For example, in deep reinforcement learning (DRL), the DRL agent only needs to select actions according to the state. Simulators are responsible for modeling and state transition (Kiran et al., 2021). Meanwhile, it has been verified that the DRL-based speed control can achieve better driving performance than MPC (Lin et al., 2020; Zhu et al., 2020).

Owing to the characteristics, the application of DRL in intelligent transportation has received much attention in recent years. Researchers have verified that good performance and broad application of DRL can be achieved in various tasks (Mao et al., 2020; Wang and Sun, 2020; Wu et al., 2020b; Zhu et al., 2018). However, DRL is also proved to be unstable, time-consuming, and site-specific. Reward function design and parameter tuning are also challenging (Ye et al., 2019). The potentials of this type of control strategy have not been fully realized yet.

Recently, several studies worked on learning comfortable and energy-efficient driving behaviors using DRL. Since the descriptions of ride comfort are discussed in different studies, we make a distinction in the comfort associated with longitudinal, lateral, and vertical movement respectively. In longitudinal motion planning, Zhu et al. (2020) trained and tested a DRL-based car-following model using real-world driving data. The comfortable speed control strategies are learned by minimizing longitudinal jerk. In lateral motion planning, Wang et al. (2019b) formulated lane change behaviors with continuous actions in DRL. The smooth lane change is trained by minimizing yaw acceleration. As for vertical motion, Ming et al. (2020) controlled the damper force of a semi-active suspension system using DRL. The displacement, speed, and acceleration of sprung and unsprung mass are major control objectives. For energy-efficient vehicle control, researchers mainly applied DRL to hybrid electric vehicles. Qi et al. (2019) proposed a DRL-based vehicle energy management system to learn the optimal power supply strategy. The experimental results show that the system can achieve 16.3% energy savings on a typical commuting trip, compared to binary control strategies. In addition, Guo et al. (2021) proposed an eco-driving algorithm for longitudinal acceleration/deceleration and lane change based on hybrid DRL. The numerical results show that the hybrid DRL approach reduces fuel consumption with an acceptable travel time. However, these studies only considered the conditions at the moment of driving, which has limitations in achieving global driving performance.

Vehicle-to-infrastructure (V2I) systems give a solution for global driving performance enhancement by providing connectivity and communication between mobile sensors on AVs, roadside units, and remote data analytics centers (cloud) (Du et al., 2021; Shi et al., 2020). Though there are some limitations in current V2I systems such as communication, security, and privacy issues (Islam et al., 2018; Liu et al., 2021b; Papageorgiou et al., 2019; Schmidt et al., 2016), researchers have confirmed some advantages of V2I systems such as reduction in vehicle collisions, traffic congestion, travel delay, and energy consumption (Wang et al., 2020a; Wang et al., 2020b), improvement of ride comfort and traffic productivity (Zhu et al., 2020; Yan et al., 2021b), availability of large-scale road information (Liu et al., 2021a) and so on.

With the development of mobile sensors and V2I technologies, it is potential to enhance global driving performance with large-scale road information. In DRL-based speed control, future road information is provided either directly or from the pre-calculated optimal speed profile. For example, Buechel and Knoll (2018) proposed a predictive longitudinal controller for AVs based on deep deterministic policy gradient (DDPG). The controller is trained to select accelerations according to future speed reference values and road grades. The experimental results show a high computation speed and good driving performance. Hartmann et al. (2019) used DDPG to generate speed control strategies based on prior knowledge. The prior knowledge is the time-optimal speed profile along the path. MPC is utilized to compute the speed profile considering vehicle dynamic and terrain characteristics. By setting the prior knowledge in the state, the convergence efficiency is improved. However, for rough pavements, it is challenging to learn model-free speed control strategies, due to irregular road profiles and high-dimensional non-linear suspension systems.

On the other hand, though globally optimal speed profiles can be easily calculated via optimization-based methods, improving vertical ride comfort by speed control is still an urgent problem. First, there is a mismatch between time-continuous evaluation and instantaneous sensation. As recommended in ISO (2631)–1 (1997), long-time vibration measurement and evaluation are representative of vibration exposure. However, it cannot reflect some instantaneous discomfort caused by distresses accurately. Second, the strategies developed based on driving phases lack flexibility. For example, Du et al. (2018) proposed comfortable speed control strategies for three driving phases: uniform, accelerating, and decelerating phases. However, the phases may be interrupted by other vehicles in dynamic traffic. Third, developed strategies are difficult to be extended to unknown rough pavements. For example, Wu et al. (2020a) proposed a speed planning method to optimize ride comfort on specific rough pavements where entire pavements are divided into general segments and rough segments. Meanwhile, it is necessary to investigate the characteristics of vibration generated by different speeds and distresses, which is laborious and impractical. Therefore, vertical ride comfort evaluation, road information processing, and widely applicable strategies are significant to achieve vertical ride comfort.

In view of the above, this study is dedicated to enhancing ride comfort and energy efficiency of AVs on rough pavements. We propose a speed control framework that is composed of road information processing, multi-objective optimization, and implementation. We suggest the notion of ‘maximum comfortable speed’ for representing the vertical ride comfort of oncoming roads.
beyond visual range. We also formulate the comfortable and energy-efficient speed control problem using DRL and MPC. In this way, neither learning the responses of a suspension system nor solving an optimal speed profile is needed; meanwhile, the optimization problem is simplified and several speed control methods can be utilized to enhance the global driving performance. We find that the DRL-based speed control model can achieve better driving performance with much shorter computation times than the one based on MPC. This study contributes to the existing literature by proposing an advanced speed control framework on rough pavements and extending application scenarios to real-world rough pavements.

The remainder of this paper is organized as follows. Section 2 presents a speed control framework on rough pavements. Section 3 proposes a future road information processing method to represent the vertical ride comfort. Section 4 presents the DRL-based comfortable and energy-efficient speed control method for AVs on rough pavements. Section 5 shows details of model training and comparison with the typical optimization-based model. Section 6 summarizes the findings of this paper and suggests a few issues for our future works.

2. The framework of speed control on rough pavements

In this section, a speed control framework for AVs on rough pavements is presented. As we only extend comfortable and energy-efficient speed control to real-world rough pavements, assumptions are given for the feasibility of the framework. First, we assume that all the needed information is detected and integrated accurately without any bias. Specifically, the crowdsourced road information (road profiles, distresses, and roughness) are obtained via advanced multi-vehicle data fusion. Second, the real road alignment and dynamic traffic information (e.g., traffic signals, variable speed limit, traffic flow) along intended driving trajectories are not included in this initial study. Instead, we introduce the dynamic speed limit to simplify modeling. The intended driving trajectory is defined as the safe, comfortable, and efficient trajectory planned on oncoming roads in real time (Lim et al., 2021). The real-time trajectory planning involves aspects such as road geometry, lane-structured roads, traffic regulations, traffic participants, and vehicle physical limitations. Specifically, the intended driving trajectory is changing due to the dynamic movements of surrounding vehicles. Third, AVs have the capability of smooth trajectory tracking (Wang et al., 2019b; Yan et al., 2021b; Ye et al., 2020). The lateral motion of AVs is not discussed in this paper. Only the vertical vibration caused by road profiles and speed variation along intended driving trajectories is considered. Forth, this framework operates after enough road information has been accumulated.

The basis of the framework is road and dynamic traffic information. The information is detected by sensors (e.g., LiDAR, camera, radar) and located by the Global Navigation Satellite System and Inertial Measurement Unit on AVs. Specifically, the road information refers to the road profiles which record the elevation of the road surface with detailed positioning. However, the mobile sensors on an AV only perceive the information within a short range. As shown in Fig. 1, it is expected that a large number of AVs will be empowered to collect the information simultaneously. With advanced communication technologies, the information is crowdsourced from the operation of AV fleets. Then, the multi-vehicle information is integrated to form the crowdsource data. The crowdsourced data has a high sampling frequency and extensive range (Liu et al., 2021a), which supports real-time speed control.

In the framework, the speed control on rough pavements at each time horizon $t_0$ to $t_0 + \Delta T$ is depicted in Fig. 2. $\Delta T$ is the simulation sample time interval. At time $t_0$, the AV at the current location $s_0$ extracts the associated future information, dynamic speed limit and the road profiles of the left and right wheels, from the crowdsourced data with a length of $S$ along the intended driving trajectories. Since the trajectories are updated in the real time, the extracted road profiles change according to the planned trajectories. The framework is dynamic. To process the road profile information efficiently, the oncoming roads are divided into several units. Then, the road profiles on each unit and different speeds are the input of the vehicle dynamic model. We simulate suspension vibration on each unit and use the weighted root square mean acceleration (WRMSA) as a metric to evaluate vertical ride comfort. As the speed with a WRMSA value lower than $0.315 \text{ m/s}^2$ is regarded as a comfortable speed (Du et al., 2018; ISO (2631)-1, 1997), the maximum speed...
that satisfies the standard on each unit are concerned. We suggest the concept of ‘maximum comfortable speed’ for representing the vertical ride comfort of oncoming roads. A sketchy maximum comfortable speed is shown at the bottom of Step 1. Further, the control input (vehicle acceleration) is optimized to achieve the objectives (e.g., driving efficiency, ride comfort, energy efficiency) based on the future vertical ride comfort information, future dynamic speed limit, and current AV features. The AV features include position, speed, and acceleration (a negative value denotes deceleration). After executing the optimized control input, the location of the AV is updated as $s_0 + \Delta S$ on its driving trajectory at the next timestep. This consecutive process is executed until the terminal condition is satisfied.

In summary, the speed control on rough pavements follows the process below.

**Step 1:** Road Information Processing. Future road information is sent to the AV for calculating future vertical ride comfort information (discussed in Section 3).

**Step 2:** Multi-objective Optimization. Future vertical ride comfort information, future dynamic speed limit, and current AV features are sent to the speed control model to optimize the acceleration (discussed in Section 4).

**Step 3:** Implementation. The calculated acceleration is used as a command input to control the AV speed.

Compared to the conventional speed control methods, this framework has the following characteristics. First, the crowdsourced road information has a high updating frequency and extensive range that can support speed control on rough pavements along the intended driving trajectories. Second, the future road information is processed in advance to simplify the multi-objective optimization problem on rough pavements. Third, the future information and current AV features are considered simultaneously, with which the AV concerned can adjust the acceleration in time to enhance control accuracy and global driving performance. Fourth, the framework is adaptive to the changing conditions. The road profile information is changing according to real-time trajectory planning in the framework.

### 3. Road information processing

Rough pavements potentially cause discomfort in the vertical motion. Conventional vertical ride comfort optimization relies on repetitive simulations and accurate modeling, which is time-consuming and difficult to solve. Since fast computation and robustness are significant for real-time speed control, we present a method for road information (road profile) processing to simplify the optimization. The processing starts with dividing the oncoming road into several units. Then, the vertical ride comfort is evaluated at different speeds on each unit using a full-car model. Further, the maximum speed that satisfies the ‘not uncomfortable’ standard on each unit represents the vertical ride comfort information (Du et al., 2018). In this way, the complicated optimization becomes a simple optimal speed control problem. The multi-objective optimization problem in the framework can be solved by many methods such as DRL and MPC. To illustrate, we use the real-world rough pavement data in Shanghai (China) as a case study.
3.1. Data preparation

To apply the proposed framework in the real world, road information of urban local and arterial roads in Shanghai (China) was collected by different vehicles in March and April 2019. The equipment on the vehicles for road information collection as discussed in Section 2 was deployed. We compiled a dataset of 1,202 detection records, including road name, district, the location and type of distresses, road profiles, road roughness, and detection time. Since there are infinite road profile samplings, we assume that AVs drive in the middle of lanes. For each rough pavement, a pair of road profiles is considered for example.

Note that the collected road profiles have a high precision so that they can reflect large and small fluctuations on rough pavements and meet the speed control demand. Compared to the random roads used in previous studies (Ming et al., 2020; Wu et al., 2020a), the road profiles include distresses in the real world like cracks, potholes, manholes, net (alligator cracks), and so on. During the detection, the images of pavements are obtained by the cameras installed on the vehicles (Li et al., 2020). Fig. 3 shows the distress information in the road profiles. The distress in each image is annotated with a bounding box representing its location and type (Du et al., 2021). The corresponding road profiles are shown in the graph on the right.

3.2. Vertical ride comfort evaluation

The vertical ride comfort is closely related to the vibration of the suspension system. The suspension system is responsible for connecting the vehicle body and its wheels. The mathematical model of the suspension system can represent the relationship between road profiles, speed, and vertical vibration (Wu et al., 2020a). The most commonly used model is the quarter-car model, but the model...
is too simple to reflect the entire vibration information of a vehicle. As ride comfort is a subjective sensation of human exposure to vibration, seat vibration should be considered. Thus, a full-car model with the modeling of a driver’s seat and passive suspensions is utilized, as shown in Fig. 4 (refer to Cantisani and Loprencipe (2010) for details). The dynamic equation of the full-car model can be summarized as:

$$M \ddot{Z} + C \dot{Z} + K Z = P$$

$$P = [k_1 Z_{r1}, k_2 Z_{r2}, k_3 Z_{l2}, 0]^{T}$$

$$Z = [z_{r1}, z_{r2}, z_{l2}, z_{l1}, \theta, \alpha, z]^{T}$$

where $M$, $C$, and $K$ are mass matrix, damping matrix, and spring matrix; $\dot{Z}$, $Z$, and $\ddot{Z}$ are the acceleration vector, speed vector, and displacement vector respectively; $I$ is identity matrix; $k_i$ is tire stiffness; $z_{r1}$, $z_{r2}$, and $z_{l2}$ are road profiles of the front left, front right, rear left, and rear right wheel; $z_{l1}$, $z_{u1}$, $z_{u2}$, $z_{u3}$, and $z$ are the front right, front left, rear right, rear left wheel, body, and seat displacement; $\theta$ is the body pitch; $\alpha$ is the body rolling; $I_r$ and $I_p$ are rolling and pitch inertia; $L_1$ is the distance between tires and x-axis; $L_2$ and $L_3$ is the distance from the seat to x-axis and y-axis; $L_r$ and $L_f$ is the distance from rear and front tires to the y-axis.

The vibration information of the seat contains displacement, speed, and acceleration in the time domain. According to ISO (2631–1 (1997)), we only take seat acceleration into account. However, the irregular fluctuations in the time domain make it difficult to establish the relationship between ride comfort and acceleration. Apart from the analyses in the time domain, frequency-domain analyses are commonly used. As the acceleration has the feature of a stationary random signal, the patterns of frequency-domain acceleration are more stable (Liu et al., 2021a). Hence, the time domain data are translated into the frequency domain using power spectral density. Then, as ISO technical committees recommend, the (WRMSA) is used as a metric to evaluate ride comfort in the frequency domain (ISO (2631–1, 1997)). The vibration in the frequency band 0.5 ~ 80 Hz has the largest impact on human sensation. Meanwhile, the effect of each separated band within this range is significantly different. The frequency band is further separated into sections by a 1/3 octave filter (Du et al., 2018). Thus, the WRMSA is calculated with a weighting coefficient assigned to each frequency band as

$$a_v = \sqrt{\frac{1}{23} \sum_{i} \omega_i^2 \int_{u_i}^{l_i} S_v(f) df}$$

where $\omega_i$ is the weighting coefficient for the $i$-th one-third octave band; $u_i$ and $l_i$ are upper and lower limiting frequencies of the one-third octave band respectively; $S_v(f)$ is the vibration acceleration at frequency $f$. Based on ISO (2631–1 (1997)), driving with a WRMSA value less than 0.315 m/s$^2$ is described as ‘not uncomfortable’. When the WRMSA value is larger than 0.8 m/s$^2$, it is uncomfortable. For vertical ride comfort, the speed control objective is to keep the WRMSA value less than 0.315 m/s$^2$. A smaller WRMSA value means a more comfortable vertical motion.

### 3.3 Evaluation unit length

The vertical ride comfort evaluation is based on the continuous suspension responses in the time domain. Generally, the long-time vibration measurement and evaluation is representative of the vibration exposure (ISO (2631–1, 1997)). However, the distribution of small and large fluctuations on rough pavements is irregular. Long-time vibration evaluation results cannot reflect vertical ride comfort accurately. Meanwhile, it is unrealistic to develop a self-adapting time window for each fluctuation. As the conditions on rough pavements are changing, the evaluation method for an AV should be uniform to reduce the burden of computation.

To address such issues, we divide the oncoming road into several evaluation units with a constant unit length. The vehicle is assumed to drive through evaluation units at different speeds according to the conventional vertical ride comfort evaluation method (Wu et al., 2020a). Then, the suspension responses at different speeds on each unit are obtained. The maximum speed on each unit that satisfies the ‘not uncomfortable’ standard is defined as maximum comfortable speed (MCS). MCS represents the vertical ride comfort information of the oncoming road. It is noteworthy that the evaluation unit length has a great impact on evaluation results. To show the impact intuitively, an experiment was executed on Yangshupu Road. The road profiles used is shown in Fig. 5. The evaluation units...
with 50 m, 100 m, and 150 m were tested in the experiment. The red line in Fig. 6 shows the variation of MCS. It is obvious that a short unit length provides high precision evaluation, but also leads to dramatic variations in the values of MCS. With a long unit length, the fluctuations in MCS are much subdued or even negligible.

Although MCS represents the vertical ride comfort information, its form is far from the actual driving speed profile that contains accelerating and decelerating phases in general. As MCS dictates comfortable driving, MCS may lead to sharp changes in acceleration. Therefore, we further propose the use of fitted MCS to simulate autonomous driving and evaluate performance shown as the black curve in Fig. 6. The B-spline interpolation is used to fit the curve between adjacent MCS values. The reason is that B-spline interpolation is widely employed to realize free-form curve tracking and speed planning in real-time high-speed vehicle control (Wang et al., 2020b). Since driving with speed below MCS or fitted MCS is not uncomfortable in the vertical motion, MCS and fitted MCS are considered two forms of future vertical ride comfort information. For convenience, we call them future MCS in discussions below.

To further determine the feasible unit length, we analyze the impact on the rough pavement dataset in Fig. 7. In the real world, the size of large fluctuations like bumps and potholes is considerably shorter than 200 m (Wu et al., 2020a). Meanwhile, the ride comfort evaluation usually focuses on localized road conditions. Thus, we divide the distributions of the unit length in a range of 20 m to 200 m with an incremental interval of 10 m. In Fig. 7, the average speed, jerk, and WRMSA of fitted MCS have a decreasing tendency when the unit length increases from 20 m to 200 m, while the irregular distribution of roughness in road profiles yields fluctuations in the average WRMSA. It is noteworthy that the average WRMSA with a long unit length is slightly larger than 0.315 m/s. A possible reason is that the values of fitted MCS in some segments are higher than MCS as shown in Fig. 6. For longitudinal and vertical ride comfort, the lower average WRMSA and jerk are preferred, and the maximum jerk should not exceed 2.94 m/s² (Du et al., 2018). For vertical ride comfort evaluation, a high average speed represents an accurate evaluation. The AV adjusts its speed in time to adapt to rough pavements, rather than driving at a low speed constantly. Therefore, with all these considerations combined, the recommended evaluation unit length is 60 ~ 70 m for urban rough pavements in the study area. We set the unit length at 60 m for the remaining analyses.

4. DRL-based multi-objective optimization

This section solves the multi-objective optimization problem using DRL. Since the control input (vehicle acceleration) is selected from a continuous space, the deep deterministic policy gradient (DDPG) algorithm is adopted. To balance the control objectives of driving efficiency, ride comfort, and energy consumption, we design a new reward function with the consideration of AV’s speed, jerk, acceleration, vibration, and vehicle-specific power. For ride comfort, instead of simulating the suspension vibration, future MCS is used to train the speed control model. Intuitively, if the speed control requirement for each control step is known, this optimization problem can be solved by existing methods (Buechel and Knoll, 2018; Hartmann et al., 2019). For the real-world rough pavements, however, the changing conditions and long-range information are great challenges for conventional speed control methods, which justifies the use of a DRL-based method below.

4.1. State and action

The DDPG algorithm is designed for learning optimal control strategies in a sequential decision-making setup with a continuous state and action space (Lillicrap et al., 2015). There are two main components of the DDPG algorithm: environment and a learning agent. At timestep \( t \), the agent observes the state \( s(t) \) in the environment. The agent selects the action, longitudinal acceleration \( a(t) \), from the action space \([-3, 3]\) m/s² according to \( s(t) \). To achieve control objectives, the variables in the state should provide enough information for the action selection. Thus, the state is described by the previous acceleration \( a(t - 1) \), current speed \( V(t) \), and future MCS:

\[
s(t) = \begin{bmatrix} a(t - 1), V(t), V_f^1(t), ..., V_f^n(t) \end{bmatrix} | a(t) \in [a_{min}, a_{max}], t \in [1, t_{max}] \tag{7}
\]

Fig. 5. Road profiles of Yangshupu Road.
where $V_i^f(t)$ is the $i$-th future MCS, $i \in [0, n]$; the length of future MCS is a multiple of evaluation unit length recommended in Section 3.3, which is determined by vehicle control demand; $V_0^f(t)$ is the MCS at the current position; $a_{\text{min}}$ and $a_{\text{max}}$ are the minimum and maximum longitudinal acceleration respectively; the maximum simulation timestep $t_{\text{max}} = L/(V_{\text{min}}\Delta T)$. $L$ is the length of the rough pavement. $V_{\text{min}}$ is the lower bound of AV’s speed. To retain the consistency of information in each timestep, the $i$-th future MCS corresponds to information in the spatial domain with a resolution of 1 m.

When the longitudinal acceleration $a(t)$ is given, the AV’s speed $V(t)$ and position $S(t)$ will be updated in the next timestep using a kinematic model:

$$V(t+1) = V(t) + a(t)\Delta T$$

$$S(t+1) = S(t) + \frac{(V(t) + V(t+1))\Delta T}{2}$$

where $\Delta T$ is simulation sample time interval, usually set as 0.1 s.

Fig. 6. Future vertical ride comfort information on Yangshupu road in Shanghai, China: tested evaluation units with (a) 50 m; (b) 100 m; and (c) 150 m.
To obey the traffic rules, the speed is constrained by the dynamic speed limit $V_d$. Meanwhile, the speed adjustment needs flexibility and feasibility in mixed traffic. Thus, the maximum speed is set to $2.24 \text{ m/s (8 km/h)}$ above the dynamic speed limit (Hu et al., 2016). The minimum speed of urban roads is zero. If the speed is out of the range, the episode will be terminated. The speed in the next timestep is defined as:

$$V(t+1) = \begin{cases} 
0, & \text{if } V(t+1) < 0 \\
V(t+1), & \text{if } V(t+1) > V_d(t+1) + 2.24 \\
V_d(t+1) + 2.24, & \text{otherwise} 
\end{cases} \quad (10)$$

### 4.2. Features of reward function

In DRL applications, it is significant to define the reward function appropriately. The main task of a DRL agent is to maximize the reward by selecting the best action at each timestep. Hence, the reward function should be associated with the speed control objectives. For safety, the future dynamic speed limit is calculated based on road alignment and dynamic traffic information. If the speed is larger than the dynamic speed limit, hazardous events such as rollover accidents or collisions may occur. Thus, AVs are supposed to drive within the dynamic speed limit on smooth pavements, but as fast as possible to complete the trip to realize driving efficiency. Whereas, on rough pavements, AVs should further control speed to improve ride comfort. Moreover, the unsuitable combination of acceleration and speed leads to extra energy consumption. These constitute an optimization problem to achieve driving efficiency, ride comfort, and energy efficiency altogether via speed control. Therefore, when AVs are driving with low speed, discomfort, or high energy consumption, a negative reward would be given for punishment. On the contrary, a higher reward will be received if higher driving efficiency, ride comfort, and energy efficiency can be achieved. However, there are no uniform formulations for the reward function, which makes the reward function design non-trivial. The reward function in this section is based on our domain expertise and many experiment trials.

#### 4.2.1. Driving efficiency

Efficient driving refers to driving as fast as possible under the premise of safety while not violating regulated speed limits. The driving efficiency feature aims to minimize the speed deviation from the dynamic speed limit. Through massive experiments, we find that the speed profile converges to the dynamic speed limit quickly when using a linear function in the driving efficiency feature. As the main driving task on rough pavements is to improve ride comfort and keep a relatively high speed, the value of the driving efficiency feature should be similar when the speed is close to the dynamic speed limit. In this way, the agent will try to explore other actions for greater ride comfort. Therefore, the driving efficiency feature is set as a quadratic function. The squared speed deviation is divided by the square of the dynamic speed limit to scale the feature into $[0,1]$. The driving efficiency feature is constructed as:

$$R_d = -\frac{(V(t) - V_d(t))^2}{V_d(t)^2} \quad (11)$$
where \( V_d(t) \) is the dynamic speed limit at timestep \( t \).

### 4.2.2. Vertical ride comfort

As described above, the driving speed directly signifies vertical ride comfort. To confine discomfort, an AV should keep its speed below the current MCS \( V_f^0(t) \). Since the speed in the region \([0, V_f^0(t)]\) does not cause an uncomfortable sensation, the action is not punished, and the vertical ride comfort feature is set as zero. When the speed is out of this region, it should get a penalty. In this way, the agent will get the information that staying in the region \([0, V_f^0(t)]\) is better than going beyond \( V_f^0(t) \). Compared with the driving efficiency feature, \( R_v \) with a constant derivative contributes to faster convergence. The vertical ride comfort feature \( R_v \) is the evaluation of the speed deviation from \( V_f^0(t) \). The feature is divided by \( V_f^0(t) \) to be scaled into \([0,1]\). The vertical ride comfort feature is described as:

\[
R_v = \begin{cases} 
0, & V(t) \leq V_f^0(t) \\
\frac{V_f^0(t) - V(t)}{V_f^0(t)}, & V(t) > V_f^0(t) 
\end{cases}
\]  

### 4.2.3. Longitudinal ride comfort

In longitudinal motion, jerk and acceleration are both associated with ride comfort. Small jerks contribute to a smooth speed trajectory (Du et al., 2018). A small longitudinal acceleration encourages a minimal speed change for drivability and a comfortable driving experience (Hu et al., 2016). Therefore, the longitudinal jerk \( j(t) \) and acceleration \( a(t) \) are utilized to evaluate longitudinal ride comfort. However, the scale of the longitudinal jerk and acceleration have a great difference in values. The longitudinal acceleration ranges from \(-3 \) to \(3 \, \text{m/s}^2\), while the largest value of jerk is \(60 \, \text{m/s}^2\). Due to the changing MCS on rough pavements, the AV should conduct relatively large longitudinal acceleration to adjust the speed in time. To achieve better speed control results, we use two forms of the longitudinal ride comfort feature in longitudinal jerk and acceleration divided by different base values. The longitudinal ride comfort feature is described as:

\[
j(t) = \frac{a(t) - a(t-1)}{\Delta T}
\]

\[
R_l = \frac{j(t)^2}{3600} \cdot \frac{a(t)^2}{90}
\]  

### 4.2.4. Energy efficiency

Energy-efficient driving is to find an appropriate longitudinal acceleration based on the current speed. To cover different types of vehicles and estimate immediate energy efficiency, vehicle-specific power (VSP) is introduced to represent the instantaneous energy consumption with only vehicle dynamics and road conditions (Sun et al., 2021; Zhao et al., 2020). The VSP is defined as instantaneous power per unit mass of a vehicle. The power is generated by the engine to overcome roadway grade, rolling resistance, and aerodynamic drag. Generally, large VSP values correspond to high fuel consumption, and negative values denote ‘fuel cut-off’ events (Jimenez-Palacios, 1998). For convenience, we use the VSP with typical values for all parameters to represent the energy efficiency in this study:

\[
P(t) = V(t)(1.1a(t) + 9.81g(t) + 0.132) + 0.000302V(t)^3
\]

\[
R_e = -\frac{P(t)}{1000}
\]

where \( P(t) \) is the value of VSP at timestep \( t \); \( g(t) \) equals to vehicle vertical rise divided by slope length at timestep \( t \) (percent).

### Table 1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>( V )</td>
<td>( \text{m/s} )</td>
</tr>
<tr>
<td>Dynamic speed limit</td>
<td>( V_f )</td>
<td>( \text{m/s} )</td>
</tr>
<tr>
<td>Maximum comfortable speed</td>
<td>( V_l )</td>
<td>( \text{m/s} )</td>
</tr>
<tr>
<td>Acceleration</td>
<td>( a )</td>
<td>( \text{m/s}^2 )</td>
</tr>
<tr>
<td>Jerk</td>
<td>( j )</td>
<td>( \text{m/s}^2 )</td>
</tr>
<tr>
<td>Road grade</td>
<td>( g )</td>
<td>( \text{percent} )</td>
</tr>
<tr>
<td>Vehicle-specific power</td>
<td>( p )</td>
<td>( \text{KW/ton} )</td>
</tr>
</tbody>
</table>
4.2.5. Immediate reward
For comfortable and energy-efficient speed control on rough pavements, the immediate reward \( r \) is the summation of the above reward items:

\[
r = w_1 R_d + w_2 R_v + w_3 R_l + w_4 R_e
\]

where \( w_1, w_2, w_3, \) and \( w_4 \) are weights. In practical applications, these weights also can be set according to passengers’ preferences. The parameters used in the reward function are summarized in Table 1.

4.3. Simulation setup and DDPG structure
Lillicrap et al. (2015) first proposed the DDPG algorithm and tested it in different environments. One of the environments is a racing game where they controlled the acceleration, braking, and steering of a vehicle successfully. Recently, researchers have extended the application to various autonomous driving scenarios such as car-following (Yan et al., 2021a; Zhu et al., 2020) and lane change (Wang et al., 2019a; Wang et al., 2019b). However, the scenario of driving on real-world rough pavements has not been designed yet, which makes the existing DRL-based speed control strategies immature. Therefore, we establish the environment for autonomous driving on rough pavements. The key components of the environment are rough pavements, dynamic speed limit, AV’s kinematic model, and terminal conditions. We model the rough pavements using future MCS corresponding to the real-world data. For the dynamic speed limit on urban roads, we use a random noise to simulate the impact of random traffic and road alignment on the speed limit. The AV’s kinematic model is described in Eq. (8) and Eq. (9). If the AV’s speed is out of the permitted range as defined in Eq. (10) or the AV’s location is out of this pavement, the episode is terminated. To simulate autonomous driving in this environment, we further propose the simulation setup and modify the DDPG structure.

In the conventional simulation setup, the AV needs to accomplish an entire trip or driving event in an episode (Qi et al., 2019; Zhu et al., 2020). However, the length of a real-world rough pavement is hundreds of meters or even thousands of meters. A great deal of time is required to train a speed control model for driving through the entire rough pavements. To address such an issue, we extract rough pavements from the dataset randomly, set a fixed timestep in each episode, and start the simulation at a random initial position. However, the exploration may be inadequate with the fixed timestep. For example, if an AV starts with a random speed that is extremely high or low, the speed cannot be adjusted to an expected level in a short time or the terminal condition may be satisfied early, which generates bad experiences. Since the DDPG agent is trained using the experiences in the reply buffer, the quality of experiences is quite important for training efficiency. Therefore, we set the initial speed as the current MCS. That is, the agent starts with a relatively good and reasonable state, compared to a random initial speed. In the following timesteps, the AV only needs to adjust its speed slightly to find the optimal control strategy. The experiences with high rewards are easy to obtain during this process. In such
a way, we can simulate speed control on various rough pavements in a relatively short time, and most experiences that the agent undergoes are effective for training.

Fig. 8 shows the structure of the DDPG-based speed control model. The key components and simulation setup described above are used in the model. The model consists of the environment and the agent with an actor-critic structure. Generally, the numbers of layers and neurons in the actor and critic networks are selected according to the difficulty of training. For example, if there are large variables in the state or complex rewards in the DRL model, huge and deep neural networks may be considered. Each neuron in the hidden layer usually uses the ReLU activation function. The final layer in the actor network uses tanh activation function to maps the value in the range [-1,1]. Since the longitudinal acceleration ranges from -1 to 1, we usually use the ReLU activation function. The final layer in the actor network uses tanh activation function to maps the value in the range [-1,1]. Since the longitudinal acceleration ranges from -3 to 3 m/s², the outputs are multiplied by 3 to map this range.

The model is trained based on the interactions between the environment and the agent. The training process is described as follows. For convenience, the timestep is expressed using subscripts in this section. In the beginning, we initialize critic network \( Q(s, a|\theta^Q) \) and actor network \( \mu(s|\theta^\mu) \) with weights \( \theta^Q \) and \( \theta^\mu \). We also initialize target network \( \bar{Q}(s, a|\theta^Q) \) and actor network \( \bar{\mu}(s|\theta^\mu) \) with weights \( \theta^Q \leftarrow \theta^Q \) and \( \theta^\mu \leftarrow \theta^\mu \). The future dynamic speed limit and future MCS values are given to the DDPG agent according to the current location. At each timestep \( t \), the input of the actor network is the state defined in Eq. (7) including the previous acceleration, the current speed, and future MCS; the output is the longitudinal acceleration with a noise: \( a_t = \mu(s_t|\theta^\mu) + \mathcal{N} \). The critic network takes \( s_t \) and \( a_t \) as inputs and outputs \( Q(s_t, a_t) \) to estimate the goodness of taking action \( a_t \) at state \( s_t \). The real-time longitudinal acceleration \( a_t \) is sent to the environment and used to update the state. From timestep \( t \) to timestep \( t+1 \), the longitudinal motion of the AV is regarded as a uniformly accelerated motion. In each learning step, the critic network updates by minimizing the loss function:

\[
L = \frac{1}{N} \sum_{i} (r_i + \gamma \bar{Q}(s_{i+1}, \bar{\mu}(s_{i+1})|\theta^\mu) - Q(s_i, a_i|\theta^Q))^2
\]

(18)

The critic network calculates the gradients \( \nabla_a Q(s, a) \). Then, the gradients are passed to the actor network to update parameters with sampled policy gradients:

\[
\nabla_a J = \frac{1}{N} \sum_{i} \nabla_a Q(s, a|\theta^Q) |_{s \sim \mu(s|\theta^\mu)} \nabla_a \mu(s|\theta^\mu) |_{s}
\]

(19)

The target networks are updated to slowly track the actor and critic networks to improve stability with \( \tau \ll 1 \):

\[
\theta^Q \leftarrow \tau \theta^Q + (1 - \tau) \theta^Q
\]

\[
\theta^\mu \leftarrow \tau \theta^\mu + (1 - \tau) \theta^\mu
\]

(20)

(21)

5. Experiments and results

In this section, we conducted experiments to verify the feasibility and performance of the proposed framework. We established a rough pavement dataset using the detected road information. DDPG models were trained and tested on the rough pavement dataset. Then, we compared the driving performance of a DDPG model with an MPC model on the whole dataset and sampled rough pavements. The MPC was solved by direct single shooting (Hicks and Ray, 1971) and implemented via CasADi (Andersson et al., 2019) in MATLAB 2020a. All the experiments were performed on a computer with i7 CPU @ 2.60 GHz and 12 GB memory. Since the relationship between speed control strategies and results is complicated and not the focus of this paper, we omit the speed controller and powertrain system and assume that strategies are realized without any deviation, which is consistent with the work of Wu et al. (2020a).

Fig. 9. IRI distribution on the rough pavement dataset.
5.1. Rough pavement dataset

Based on the detected road information, we further established a rough pavement dataset for experiments. Road roughness was evaluated using the ‘international roughness index’ (IRI) (Liu et al., 2021a). According to the work of Li et al. (2020), the frequently-used evaluation length of IRI ranges from 10 m to 1 km. As some detected roads are short, the roughness of each 200-meter segment is considered herein. The road roughness of each segment is calculated by multi-vehicle IRI estimation using the algorithm proposed by Liu et al. (2021a). According to the study of road quality evaluation in Shanghai (Zhou et al., 2007), 110 roads with IRI values larger than 6.1 m/km were selected. Since we assume that AVs drive in the middle of lanes and consider a pair of road profiles for each rough pavement for example, we extracted 110 pairs of road profiles of left and right wheels to form the rough pavement dataset. The IRI distribution of the rough pavement dataset is shown in Fig. 9.

5.2. DDPG training and convergence

5.2.1. Network structure and future information

Many factors affect the training performance of DRL models, such as the network structure (Wang et al., 2019b; Ye et al., 2020). In our case, one distinctive factor is the future MCS. The form and length of future MCS both have an impact on the training performance. Specifically, future MCS is a speed profile in the spatial domain. The length of future MCS refers to how long the speed profile of oncoming roads is given. Although both MCS and fitted MCS can be the state variables in the DDPG model, fitted MCS is close to the actual speed profile that may contribute to training. Meanwhile, the length of future MCS should be long enough so that the DDPG model can learn accurate speed control strategies. Thus, a longer length improves driving performance; however, it increases the computational demand and creates difficulty in training. If the length of future MCS is too long, deep networks and a large number of neurons are required.

To find out the best network structure and future MCS information, we trained different DDPG models on the rough pavement dataset. For the structure of the network, 1 ~ 3 hidden layers and 10 ~ 50 neurons in the hidden layers are used according to previous studies (Ye et al., 2020; Zhu et al., 2020). For future MCS, we test the length (60 m, 120 m, and 180 m) and types of future MCS (MCS and fitted MCS). To show the training performance intuitively, we number the model with different hidden layers and the length as listed in Table 2 and further compare the group [(1), (4), (7)], [(2), (5), (8)], and [(7), (8), (9)] with different network structures. To train the models, we assume that passengers assign equal importance to driving efficiency, ride comfort, and energy efficiency. Since each feature in the reward function is into the range [0,1], we set all the weights in the reward function as 1, which is one of the common reward function designs (Zhu et al., 2020). The other hyper-parameters of the DDPG model are listed in Table 3. The maximum timestep in each episode is set as 300 for fast experiments.

Fig. 10 shows the training process clearly with the mean episode reward in the translucent colors and the rolling mean episode reward in solid colors. The mean episode reward is the average reward in an episode. The higher the mean episode reward is, the better learning performance is. The rolling mean episode reward is the average of mean episode rewards with a rolling window of 10 episodes. The rolling mean episode is clear for comparison. To control the variables in experiments, we set MCS as the state and 30 neurons in each hidden layer. Fig. 10 (a) and (b) show that a long length of MCS and deep network contribute to stable convergence and high rewards. However, training with a deep network is time-consuming. Thus, we recommend finding a suitable length of MCS within limited hidden layers. Fig. 10 (c) demonstrates that the convergence of Model (7) is faster and earlier than Model (8), while the convergence of Model (9) is unstable. It illustrates that it is easy to train with a short length of MCS. Consequently, we set the length 60 m considering the stability and efficiency in training.

For the number of neurons in each hidden layer, the training performance of Model (1) with 10, 20, 30, and 50 neurons are compared in Fig. 10 (d). Model (1) with 20, 30, and 50 neurons have converging trends after 100 episodes, while the total reward of Model (1) with 10 neurons is extremely low. Among them, Model (1) with 50 neurons has the highest reward. Further, when 50 neurons are used in Model (7), (8), and (9) (see Fig. 10 (e)), it is obvious that the training performance is improved. It demonstrates that increasing the number of neurons can also improve training performance. Similarly, when the future vertical ride comfort information changes to fitted MCS (see Model (9) in Fig. 10 (f)), there is an improvement in convergence as adding neurons. The reason is that the values of fitted MCS are generated in accordance with driving habits and easy to track. Based on the experimental results above, we set 50-30-20 units in the three hidden layers and 60-meter fitted MCS in the state for stable and efficient training performance.

5.2.2. Trade-off between objectives

As the weights in the reward function determine final driving performance, the weights should be set towards speed control objectives. Nevertheless, the possible weight settings are infinite. As mentioned above, if passengers assign equal importance to the

<table>
<thead>
<tr>
<th>Hidden layers Length of future MCS (m)</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>(1)</td>
<td>(4)</td>
<td>(7)</td>
</tr>
<tr>
<td>120</td>
<td>(2)</td>
<td>(5)</td>
<td>(8)</td>
</tr>
<tr>
<td>180</td>
<td>(3)</td>
<td>(6)</td>
<td>(9)</td>
</tr>
</tbody>
</table>
objectives, all the weights can be set as 1. To match the goal of ride comfort and energy efficiency better, the corresponding weights should be increased. Thus, we trained two DDPG models with different weights (Model (a): $w_1 = w_2 = w_3 = w_4 = 1$, and Model (b): $w_1 = 1, w_2 = w_3 = w_4 = 3$) on the rough pavement dataset as examples. The maximum timestep in each episode is set as 100 to learn the speed control on as many rough pavements as possible in a relatively short training time.

![Fig. 11. The training performance of the DDPG models.](image)

Table 3  
Parameter setting for the DDPG.

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR_A</td>
<td>0.0001</td>
<td>Learning rate of the actor network</td>
</tr>
<tr>
<td>LR_C</td>
<td>0.001</td>
<td>Learning rate of the critic network</td>
</tr>
<tr>
<td>BATCH_SIZE</td>
<td>1624</td>
<td>Number of cases used by stochastic gradient descent update</td>
</tr>
<tr>
<td>MAX_EPISODES</td>
<td>1000</td>
<td>Number of episodes</td>
</tr>
<tr>
<td>DISCOUNT_FACTOR</td>
<td>0.9999</td>
<td>Q-leaning discount factor gamma</td>
</tr>
<tr>
<td>MEMORY_CAPACITY</td>
<td>20,000</td>
<td>Number of cases in reply buffer</td>
</tr>
</tbody>
</table>

Fig. 10. Track of reward with different hyperparameters. The neuron number in hidden layers of (a), (b), (c), and (f) is 30. The nodes in hidden layers of (d) are 10, 20, 30, and 50 neurons. The nodes in hidden layers of (e) are 50 neurons. Models in (a), (b), (c), (d), and (e) use MCS as future information. Models in (f) use fitted MCS.

Fig. 11 shows the reward of the DDPG models with respect to the training episode. As seen, both the DDPG models have a stable training performance. To show the details, Fig. 12 depicts the speed trajectories generated by the models on Zhengli Road. The
Fig. 12. Driving speed of the DDPG models on Zhengli Road. The driving performance is evaluated as follows. The driving time, WRMSA, average absolute longitudinal jerk, and average VSP of Model (a) are 225.2 s, 0.3330 m/s², 0.6147 m/s³, and 1.1177 KW/ton, while those of Model (b) are 243.8 s, 0.3196 m/s², 0.2746 m/s³, and 0.7770 KW/ton.

Fig. 13. Comparison of driving performance on the rough pavement dataset: (a) speed distribution; (b) WRMSA distribution; (c) longitudinal jerk distribution; and (d) VSP distribution.
Fig. 14. Driving speed of the MPC model and the DDPG model on sampled rough pavements: speed profiles on road (a) PR, (b) NSR, (c) CR; (d) ZR, and (e) YR.
increasing of the weights improves ride comfort and energy efficiency, but it also increases driving time. However, driving slowly is potentially hazardous, especially on even segments where vehicles tend to maintain a relatively high speed. In this situation, driving slowly may result in massive traffic jams or even crashes. For autonomous driving on rough pavements, all the driving objectives, driving efficiency, comfort, and energy efficiency, are important. Meanwhile, the detailed trade-off between the objectives is not the mainly contribution of this paper. Therefore, the DDPG model with equal weights is adopted in the following experiments.

5.3. Comparing DDPG with MPC

Recently, model predictive control (MPC) is the most frequently used model for a variety of problems, such as decision making, motion planning, and vehicle control in autonomous driving (Dixit et al., 2020; Hang et al., 2020; Wu et al., 2020; Zhu et al., 2020). In speed control, MPC is used to control AVs’ acceleration for safe, efficient, comfortable, and energy efficient car-following behavior (Li et al., 2015; Takahama and Akasaka, 2018; Zhu et al., 2020). At each timestep, MPC generates a sequence of acceleration by solving an optimal control problem within a prediction horizon, and then only the first acceleration of the sequence is applied (Camacho and Alba, 2013). This process repeats until the terminal conditions satisfied. Major advantages of MPC are capabilities of handling constraints on control inputs (vehicle acceleration) and system states (vehicle speed and position) and performing predictive control (Zhu et al., 2020). Thus, the driving performance of the DDPG model is compared with MPC. For MPC formulation, the kinematic point-mass model in Eq. (8) and Eq. (9) is described in a vector form:

\[ x(t+1) = Ax(t) + Bu(t) \] (20)

Fig. 15. Comparison of driving performance of the MPC model and the DDPG model on sampled rough pavements: (a) driving time; (b) WRMSA; (c) average absolute longitudinal jerk; and (d) average VSP.
where \( t \) is the timestep, \( x(t) = [s(t), v(t)]^T \), \( u(t) = a(t) \), \( A = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix} \) and \( B = \begin{bmatrix} 0.5\Delta T^2 \\ \Delta T \end{bmatrix} \).

The MPC is implemented by optimizing the comfortable and energy-efficient speed control problem. For comparison, the objective function and constraint conditions should refer to the DDPG model. Note that, limited by the MPC formulation, it fails in finding the optimal solution when setting Eq. (12) in the objective function. To achieve similar effects, we use fitted MCS as a reference for speed control. The vertical ride comfort is optimized to minimize deviations between speed and the current MCS. The other objective sub-functions are the same as Eq. (11), Eq. (14), and Eq. (16) in the DDPG model. Similar to the DDPG model, the speed is also constrained by the dynamic speed limit in mixed traffic as defined in Eq. (10) and the longitudinal acceleration is scaled in \([-3,3]\) m/s\(^2\). Therefore, a constrained MPC formulation is defined as follows:

\[
\sum_{t=0}^{N-1} \left[ W_1 \left( \frac{v(t)-v_0(t)}{v_{f}(t)} \right)^2 + W_2 \left( \frac{a(t)}{f_{\text{max}}} \right)^2 + W_3 \left( \frac{v(t)-v_{d}(t)}{v_{f}(t)} \right)^2 + W_4 \frac{P(t)}{1000} \right] 
\]

\[\text{s.t.} \quad x(t+1) = Ax(t) + Bu(t)\]

\[0 < v(t) < v_d + 2.24\]

\[-3 < a(t) < 3\]

where \( N \) is the prediction horizon (\( N=30 \) in this study), \( f_{\text{max}} \) is the maximum value of longitudinal jerk, which is set as 60 m/s\(^3\). \( \alpha^2 \) is a base value and set as 90. The weights are set as \( W_1 = 10, W_2 = 1, W_3 = 1, W_4 = 1 \). \( u = [a(0), a(1), \ldots, a(N-1)] \) is the solved action sequence in each timestep, and only the first action \( a(0) \) is implemented. This process is repeated until the terminal condition arrives. The MPC-based speed control model is referred to as the MPC model below.

5.3.1. Driving performance on the dataset

To compare the performance, we ran the DDPG models with equal weights and the MPC model on 110 rough pavements and set the simulation sample time interval \( \Delta T \) as 0.1 s. Fig. 13 shows the probability of speed, longitudinal jerk, WRMSA, and VSP. The computation times of the DDPG and MPC model are 0.05 h and 0.89 h for the whole rough pavement dataset respectively. The computation efficiency of the DDPG model increases 94.38%.

Fig. 13 (a) shows that the speed controlled by the DDPG model is smaller than that of the MPC. This contributes to improvements of vertical ride comfort. Thus, smaller WRMSA values are presented in Fig. 13 (b). Though the longitudinal jerk in Fig. 13 (c) is larger than that of the MPC, almost all the longitudinal jerks are in the range of \([-2.94, 2.94]\) m/s\(^2\), which meets the longitudinal ride comfort demand (Du et al., 2018). Fig. 13 (d) shows that the DDPG model reduces the high energy consumption. It indicates that the DDPG model can find a better combination of longitudinal speed and acceleration. In summary, the DDPG model improves AV’s driving performance in vertical ride comfort and energy efficiency by 8.22% and 24.37% respectively. With the premise of longitudinal ride comfort, the DDPG model achieves better driving performance in vertical ride comfort and energy efficiency. Note that the improvements are achieved under equal weights in the DDPG model. Certainly, the vertical ride comfort and energy efficiency can be further enhanced by increasing corresponding weights. Thus, the comparison result not only illustrates the adaptability of DDPG model on rough pavements, but also confirms the effectiveness of the DRL-based speed control method.

5.3.2. Driving performance on the sampled roads

To illustrate the comfortable and energy-efficient driving of the DDPG and MPC models, we further sampled five rough pavements with different roughness, including Pingliang Road (PR), North Sichuan Road (NSR), Changlong Road (CR), Zhoujiazui Road (ZR), and Yangshupu Road (YR). The driving performance on the sampled rough pavements are compared in Fig. 14 and Fig. 15. As both the DDPG and MPC models utilize future information, we use the first 2000 m of each sampled road for example. Since the MCS represents maximum speed which satisfies ‘not uncomfortable’ standard, the gentle variation and low value of MCS indicates that the pavement is rough. Fig. 14 shows the MCS variation is large on most rough pavements, while the Yangshupu Road is special with bad road quality due to the gentle variation and low value of MCS.

As shown in Fig. 14, the DDPG model learns a predictive speed control policy after iterations. The policy not only balances multiple objectives, but also finds feasible speed profiles based on future information. In this way, the DDPG model can adjust speeds in advance and avoid rapid speed variation. Meanwhile, Fig. 15 shows that the output of the DDPG model has smaller WRMSA, jerk, and VSP than the MPC model. Specifically, the WRMSA values on the Pingliang Road and the North Sichuan Road of the DDPG model are below 0.315 m/s\(^2\) and satisfy the ‘not uncomfortable’ standard (ISO (2631)–1, 1997). Though the others are still above 0.315 m/s\(^2\), the proportion of passengers who cannot tolerate the vertical vibration is decreased (Du et al., 2018). For the Yangshupu Road, due to the little MCS variation, it is easy to follow the MCS and balance objectives. Intuitively, the driving performance of the DDPG and MPC models on the Yangshupu Road can be similar. However, the results on the Yangshupu Road indicate that there is still a ride comfort and energy efficiency enhancement even on this extremely rough pavement. Overall, the DDPG model demonstrates superiority in comfortable and energy-efficient driving on rough pavements.

By long-time exploration and learning on rough pavements, the DDPG model has the following advantages.

First, the proposed DDPG model runs faster computationally than the MPC as a rolling optimization method. In this case, the computation times of the DDPG model and the MPC model are 1.30 s and 44.07 s, respectively. We also tested the MPC model on the
Yangshupu Road and recorded computation times in Fig. 16. When the prediction horizon reaches around 200 (meaning that the maximum length of future MCS is 377.78 m), it needs 267.57 s to make the speed control strategy for updating AV's action at each timestep of 0.1 s. However, even a long future MCS or road length does not add a large burden to the computation of the DDPG model. It implies that the proposed DDPG model can support real-time speed control for autonomous driving.

Second, the reward or objective function design is more flexible in the DDPG model. For MPC formulation, if the objective function is inappropriate, the optimizer may fail in finding a solution to the constrained optimal control problem. However, the training of the DDPG model is based on the reward function, which can use any form of reward functions such as linear, quadratic, and piecewise models.

Third, the DDPG algorithm is more appropriate for AV speed control on rough pavements. Compared with the MPC model, the proposed DDPG model can achieve higher ride comfort and lower energy consumption. The superiority of the DDPG model is that the model can adapt to different rough pavements and balance different driving requirements.

In the experiments, we assume that passengers assign equal significance to different driving objectives in the DDPG model. The results show that the driving time of the DDPG model is longer than the MPC model. Actually, for driving efficiency, we can increase the corresponding weights and retrain the model with knowledge transfer. However, it is difficult to improve driving efficiency, vertical ride comfort, and energy efficiency at the same time. In other words, the increase of driving speed may cause discomfort and waste energy. Meanwhile, compared to the MPC model, it seems that the DDPG model achieves ride comfort and energy efficiency by deceleration. In fact, comfortable and energy-efficient autonomous driving is an optimization problem with respect to vehicle speed, acceleration, and jerk.

6. Conclusion

With the development of sensor and communication technologies, the crowdsourced road and dynamic traffic information of rough pavements is increasingly available for driving performance enhancement. This paper is dedicated to enhancing the driving efficiency, ride comfort, and energy efficiency on rough pavements, with a targeted operation of AVs. We proposed a speed control framework on rough pavement based on the crowdsourced data. We suggest the concept of the maximum comfortable speed to represent the vertical ride comfort of the oncoming road in this framework. A speed control model based on deep reinforcement learning was designed, trained, and tested with the real-world rough pavements in Shanghai, China. To balance driving efficiency, ride comfort, and energy efficiency requirements, the reward function is designed with the consideration of speed, vertical vibration, longitudinal jerk, longitudinal acceleration, and vehicle-specific power. The experimental results show that on average, 8.22%, 24.37%, and 94.38% improvements are achieved in vertical ride comfort, energy efficiency, and computation efficiency respectively, compared to the MPC-based speed control model. The results indicate that the proposed framework is advantageous and applicable in real-time speed controls on rough pavements.

In our future work, we plan to extend this framework by modifying the vehicle motion simulator. A high-fidelity simulator with

![Fig. 16. Computation times of the MPC model on Yangshupu Road.](image-url)
longitudinal, lateral, and vertical motion can further improve vehicle control accuracy and driving performance. Apart from the longitudinal speed control, an active or semi-active suspension system can also improve vertical ride comfort via damper force control. Meanwhile, AVs can keep relatively high driving speeds by controlling the longitudinal and vertical motion simultaneously. The large-scale and high dimensional road information provides the profile and location of large fluctuations such as bumps and potholes. This allows AVs to adjust the driving trajectories or change lanes to avoid dramatic vertical vibration. Furthermore, the proposed speed control framework will be developed in various driving scenarios of self-driving car and bus. The speed control strategies will further take the dynamically changing movements of surrounding traffic participants into account.

Credit authorship contribution statement

Yuchuan Du: Conceptualization, Investigation, Validation, Writing - review & editing. Jing Chen: Methodology, Experiments Design, Writing - original draft. Cong Zhao: Conceptualization, Investigation, Methodology, Experiments Design, Validation, Writing - review & editing. Chenglong Liu: Validation. Feixiong Liao: Formal Analysis, Writing - review & editing. Ching-Yao Chan: Writing - review & editing.

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

The study was jointly sponsored by the National Natural Science Foundation of China (52102383), the China Postdoctoral Science Foundation (2021M692428), the Innovation Program of Shanghai Municipal Education Commission (2021-01-07-00-07-E00092), and the Shanghai Municipal Science and Technology Major Project (2021SHZDZX0100). The corresponding author was supported by the Shanghai Sailing Program (21YF1449400). The authors would like to thank the anonymous reviewers for their constructive comments.

References


