Journal of Systems Engineering and Electronics
Vol. 33, No. 4, August 2022, pp.812 – 826

Predictive cruise control for heavy trucks based on slope information under cloud control system

LI Shuyan[1], WAN Keke[1], GAO Bolin[2,*], LI Rui[1], WANG Yue[2], and LI Keqiang[2]

1. College of Engineering, China Agricultural University, Beijing 100083, China;
2. School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China

Abstract: With the advantage of fast calculation and map resources on cloud control system (CCS), cloud-based predictive cruise control (CPCC) for heavy trucks has great potential to improve energy efficiency, which is significant to achieve the goal of national carbon neutrality. However, most investigations focus on the on-board predictive cruise control (PCC) system, lack of research on CPCC architecture under CCS. Besides, the current PCC algorithms have the problems of a single control target and high computational complexity, which hinders the improvement of the control effect. In this paper, a layered architecture based on CCS is proposed to effectively address the real-time computing of CPCC system and the deployment of its algorithm on vehicle-cloud. In addition, based on the dynamic programming principle and the proposed road point segmentation method (RPSM), a PCC algorithm is designed to optimize the speed and gear of heavy trucks with slope information. Simulation results show that the CPCC system can adaptively control vehicle driving through the slope prediction, with fuel-saving rate of 6.17% in comparison with the constant cruise control. Also, compared with other similar algorithms, the PCC algorithm can make the engine operate more in the efficient zone by cooperatively optimizing the gear and speed. Moreover, the RPSM algorithm can reconfigure the road in advance, with a 91% roadpoint reduction rate, significantly reducing algorithm complexity. Therefore, this study has essential research significance for the economic driving of heavy trucks and the promotion of the CPCC system.

Keywords: predictive cruise control (PCC), cloud control system (CCS), layered architecture, road point segmentation method (RPSM), economic driving.

DOI: 10.23919/JSEE.2022.000081

1. Introduction

With the rapid development of the mobile communication technology and cloud computing, the research on cloud control system (CCS) and its applications has become an important research scope [1–5]. In the field of intelligent and connected vehicles (ICV), the cloud-based predictive cruise control (CPCC) system can obtain the road and traffic information by cloud platform and perform predictive control for vehicles by wireless communication. This control method can significantly improve the driving to be safe and fuel-efficient, which has essential research implications for achieving national carbon neutrality targets as well [6]. For heavy trucks, the predictive cruise control (PCC) algorithm based on slope information is an important means of improving fuel efficiency and reducing transport costs [7,8]. However, in the current on-board PCC algorithm, the problems of poor real-time computing, difficulty in map updating and high application cost seriously hinder its popularization and application. Since the CCS has super computing power and data resources, the research of PCC based on CCS is an effective method to solve the above problems [9].

Most research on the CCSs focuses on intelligent agriculture, automated factories, intelligent transport system (ITS), ICV, etc. In the early days, these applications were derived from research on the methods of network control systems. Later, with the development of artificial intelligence and cloud computing, the research on CCS has made rapid progress in various fields [10]. In the field of intelligent agriculture, Khattab et al. [11] proposed a precision agriculture cloud control architecture for intelligent control of crops based on a cloud platform of the Internet of Things. The proposed three-tier architecture has certain performance advantages, but no practical validation of its effectiveness has been carried out. Et-taibi et al. [12] proposed an intelligent irrigation CCS by taking smart agriculture as a cyber-physical system (CPS), and verified the feasibility of the system through experiments. In the field of automated factories, Hu et al. [13] pro-

Manuscript received March 01, 2022.
*Corresponding author.
This work was supported by the National Key Research and Development Program (2021YFB2501003), the Key Research and Development Program of Guangdong Province (2019B090912001) and the China Postdoctoral Science Foundation (2020M680531).

posed a CCS architecture for smart robot factories based on edge computing and cloud service theory, which significantly improved productivity. Chen et al. [14] explained the definition, architecture and application cases of intelligent factories based on CPS, providing a new solution for the optimization design of intelligent factories. In the field of ITS and ICV, Xia et al. [15] designed an intelligent traffic cyber-physical control system based on CCS principle. The simulation results showed that the system improved the dynamic performance of the traffic control system. Based on the CPS theory, Li et al. [6] proposed China's architecture of CCS for ICVs and named it as vehicle-road-cloud integrated control system, which is dedicated to bridging the two major areas of ITS and ICVs, with the aim of accelerating the development of intelligent vehicles in China. The above research includes not only innovative exploration of CPS in various fields, but also specific applications in distinctive sub-fields, but the architecture of CPCC system under CCS is less well studied.

The control methods of the predictive cruise control algorithm with the use of slope information can be divided into three categories: optimal control [16–18], model predictive control (MPC) [19–21] and instantaneous control [22,23]. Dynamic programming (DP) is based on optimal control theory and can always generate the most fuel-efficient results in handling PCC planning problems. Hellstrom et al. [24,25] proposed a look-ahead control algorithm based on the DP, which uses the road slope information ahead of the vehicle to carry out the minimum fuel-saving control for heavy diesel trucks. Cong et al. [26] used the road topology information learned in advance to construct the optimal shifting strategy based on DP, aiming to achieve the goal of minimizing fuel for heavy trucks. Compared with optimal control, MPC can improve the robustness of the system through rolling time domain optimization, which has been widely studied in PCC. In [27,28], based on the road slope information, the MPC control problem was constructed and solved with fuel economy as the objective and the simulation results show that the developed system can significantly reduce vehicle fuel consumption. However, the optimal control algorithm and MPC have high computational complexity and poor real-time performance in solving optimal control problems. Therefore, some researchers try to use the instantaneous control method to design the fuel-saving algorithm. For example, Xu et al. [29] designed a feedback controller with fuel economy priority only based on the current road slope information, which greatly reduces the calculation time. However, instantaneous control only optimizes the vehicle power system according to the current driving conditions, which has no predictive ability and is worse than optimal control and MPC in fuel-saving [30].

Although the existing research has made great contributions to the design of the PCC algorithm based on slope information, they have some deficiencies. On the one hand, the PCC algorithm studied by most scholars only considers the single optimization of vehicle speed or gear, and few people simultaneously optimize the two control objectives to meet the dynamic and economic requirements of vehicles under different driving conditions, which is particularly important for heavy trucks when driving on the slope. On the other hand, most researchers optimize the computational complexity of the PCC algorithm by adding constraints or simplifying objective functions. However, few people reduce the dimension of the algorithm from the perspective of road segmentation.

Given the above, under the ICV CCS, this paper proposes a layered architecture for the CPCC system. Besides, based on the dynamic programming principle and the road point segmentation method (RPSM), a novel PCC algorithm is presented. In comparison with the existing studies, two distinctive contributions of this work are as follows:

(i) A layered architecture for the CPCC system. This architecture can realize the convenient acquisition of maps and the collaborative management of ICV groups, which is conducive to the industrialization promotion of the PCC algorithm.

(ii) A novel PCC algorithm. This algorithm can use the predicted slope information to optimize both speed and gear, considering simultaneously the vehicle's economy and dynamics as well as reducing the algorithm's computational complexity by RPSM.

This paper is organized as follows. In Section 2, the system architecture of CPCC is designed. In Section 3, the vehicle model is constructed. In Section 4, the algorithm and principle of the CPCC system are presented. In Section 5, the simulation and analysis are provided. Finally, the conclusions are summarized in Section 6.

2. System architecture of CPCC

The general architecture of the ICV CCS has been proposed for many years [6], but the architecture is not invariably applied for different applications, which needs to be adapted and redesigned. In this chapter, the general architecture of the ICV CCS is introduced. Then, based on the general architecture, the layered architecture for the CPCC system is designed.

2.1 General architecture of the ICV CCS

This general architecture, also known as the vehicle-road-cloud integrated control system, is an integrated control system with a cloud control platform (CCP) as the core and oriented to ICVs and transportation [6,9]. This architecture is composed of CCP, roadside infrastructures, communication network, ICVs and related supporting platform. Its overall architecture is shown in Fig. 1.

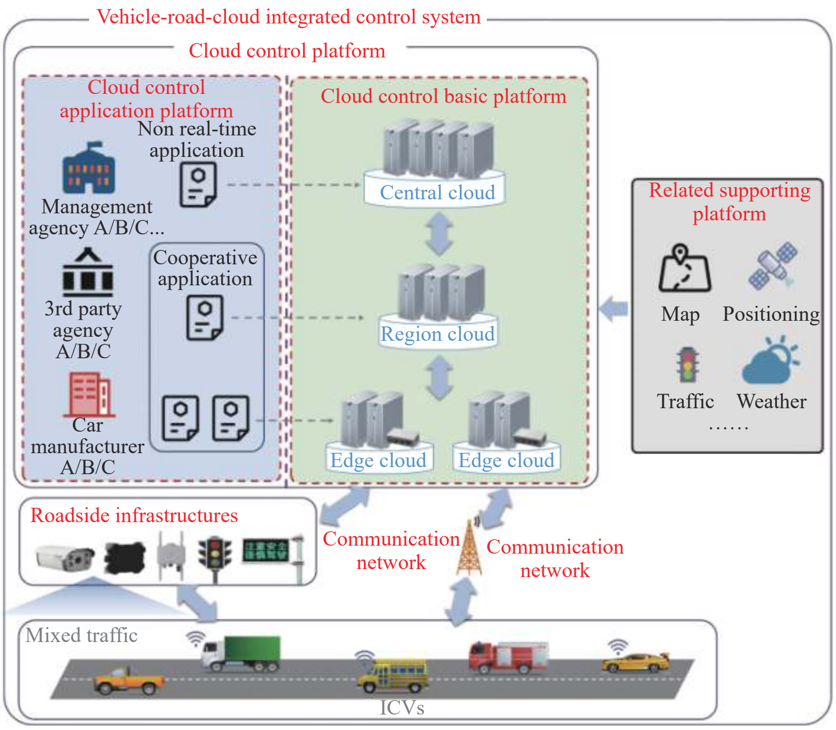


Fig. 1 Architecture of CCS for ICVs [6]

The CCP aims to build a real-time computing environment and integrate data of vehicle-road-cloud for cooperative application in performance optimization of ICVs and traffic, which is divided into cloud control basic platform and cloud control application platform. The basic platform provides real-time traffic data and an application environment for collaborative applications. To better support the service needs of different applications, the CCP is divided into edge cloud, region cloud and central cloud. Edge cloud usually runs real-time collaborative applications, regional cloud runs near-real-time collaborative applications, central cloud services in the country, and runs non-real-time applications. The three tiers of clouds are coordinated with each other, with a gradual decrease in real-time service from the bottom to the top and a sequential increase in service volume. In the application platform, there are management agencies, 3rd party agencies and carmanufacturers depending on the needs of each institution for the applications. The ICVs in traffic interact with the cloud via the communication network or roadside communication units. The roadside sensing unit senses road traffic data in real-time and uploads it to the edge cloud for application processing. The related supporting platform provide data support such as maps and traffic for the CCP.

PCC requires fast calculations and road map data in real-time. With the advantages of CCS, its control effects will be further enhanced by placing it in the ICV CCS. The PCC algorithm is arranged on the edge cloud to quickly complete the real-time solution of the control command by calling the map according to the signal of the vehicle cruise. The regional cloud can predict the traffic status information in an area, and achieve regional macro traffic control. The central cloud is mainly devoted to non-real-time application analysis and vehicle management tasks for PCC, which is used to manage vehicle configuration parameters and PCC operational data. Through the big data analysis of the running data for a period, some applications can be realized, such as the continuous optimization of the PCC algorithm, the analysis of the user's driving habits, and the fuel-saving statistics. By taking PCC as an example to introduce the functions and advantages of CCS at all levels, it can show the research significance of the cloud-based predictive cruise system and its architecture. Therefore, under the ICV CCS, the design of layered architecture for the CPCC system is described in Subsection 2.2.

2.2 Design of layered architecture for CPCC system

The realization of the CPCC system requires reasonable analysis of cloud and vehicle functions to ensure effective use of their respective advantages and realize efficient and safe operation of the system. In order to reduce the hardware cost of the on-board Telematics BOX (T-BOX), the vehicle platform mainly parses the control commands sent from the cloud and monitors the cruising state of the vehicle based on the set state transition conditions, so as to send and receive control commands and safely shift cruise modes. The layered architecture of the CPCC system is shown in Fig. 2 below.

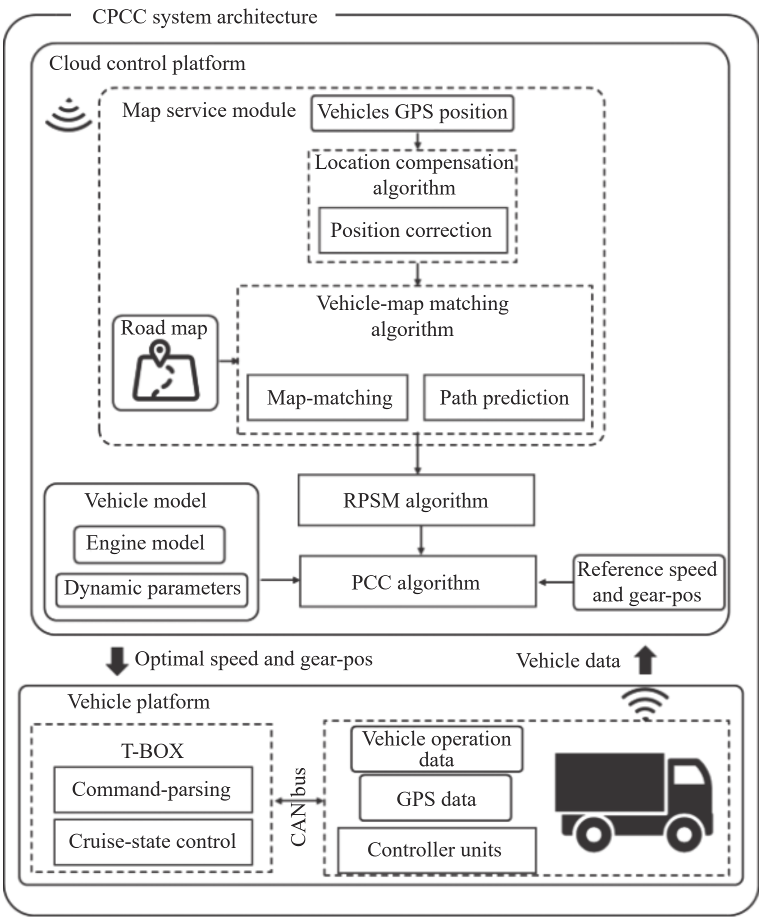


Fig. 2 Schematic diagram of layered architecture

The CPCC system is mainly composed of Map service module, RPSM algorithm, PCC algorithm, T-BOX, and controller units, as shown in Fig. 2. The CCP is equipped with the engine model and dynamic parameters of the controlled vehicle, and the parameters of the vehicle such as GPS position, reference speed and gear are obtained in real-time through the edge cloud. These data are stored on the cloud control basic platform, pending the use of the PCC algorithm. The cloud control application platform provides a fast-computing environment for the PCC algorithm, and obtains the data of the basic platform according to the needs. The vehicle platform is mainly composed of a T-BOX and controller unit for the analysis and control of cloud control instructions. The T-BOX connects internally with the controller area network (CAN) bus and externally with the CCS through wireless communication. The command parsing module is used to locate and extract the control sequence and send it to the control units for execution. Cruise state control is based on the set state transition logic to control the cruise state with security as the goal. In addition, the vehicle operation data is uploaded to the cloud by the T-BOX for statistical analysis of the data by the application platform.

The cruise state control logic in the T-BOX identifies the GPS signal quality, network status, data format, vehicle speed and cruise switch signal to determine whether the condition of state transfer is met and ensure the safe operation of the system. The transformation logic diagram is shown in Fig.3. The conditions are described in Table 1.

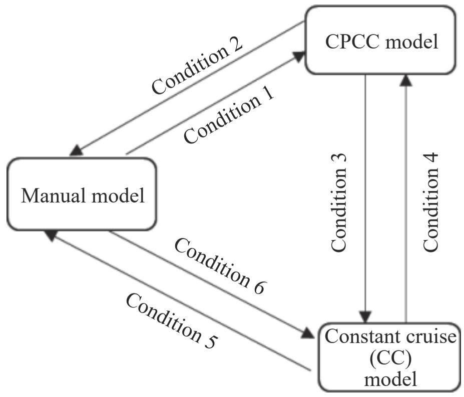


Fig. 3 Transformation logic diagram

Table 1 Table of conditions

Condition series	Value of relevant signals
Condition 1	CPCC switch=1 and data format=1 and network status=1 and GPS signal quality=1 and vehicle speed$\in [v_{\min}, v_{\max}]$
Condition 2	CC switch=0
Condition 3	CPCC switch=0 or data format=0 or network status=0 or GPS signal quality=0 or vehicle speed$\notin [v_{\min}, v_{\max}]$
Condition 4	Same as Condition 1
Condition 5	Same as Condition 2
Condition 6	CC switch=1

3. Vehicle model

The vehicle model mainly includes engine model, transmission model and longitudinal dynamics model. This section introduces the vehicle model required by the PCC algorithm below, and simplifies the model partly. The truck prototype studied in this paper is shown in Fig. 4. The main parameters of the truck are listed in Table 2. Since some parameters involve enterprise confidentiality, only typical parameters related to the proposed algorithm are published, and others are given only the reference range.

Fig. 4 Prototype of the investigated truck

Table 2 Parameters of the truck

Item	Parameter	Symbol	Value
Engine	Engine speed range/(r/min)	ω_e	[700 2 100]
	Engine torque range/Nm	T_e	[−160 2 549]
	Moment of inertia /kgm^2	J_e	[2.5 3.5]
Driveline	Final drive ratio	i_0	3.7
	Transmission ratio	i_g	[12.26,9.56,7.36,5.8, 4.52,3.53,2.71,2.12, 1.62,1.29,1,0.78]
	Final drive efficiency	η_0	0.95
	Transmission efficiency	η_g	[0.966,0.966,0.969,0.971, 0.975,0.977,0.985,0.988, 0.988,0.989,0.989,0.99]
Longitudinal force	Vehicle curb mass/kg	m	49 000
	Effective tire radius/m	r_w	0.459
	Frontal area/m^2	A	[5.0 10.0]
	Air drag coefficient	C_D	[0.5 1.0]
	Rolling resistance coefficient	f	[0.010 0.018]

3.1 Engine model

An engine model is constructed according to the universal characteristic data of a certain type of diesel engine, which includes three columns of data: engine speed, torque, and fuel consumption rate. Based on the linear interpolation principle (throttle opening approximately equals torque percentage), engine MAP data is generated into two functions of engine fuel consumption rate m_f and output torque T_e, as shown in (1).

$$\left\{\begin{array}{l} \dot{m}_f = \dot{m}_f(\omega_e, \alpha) \\ T_e = T_e(\omega_e, \alpha) \end{array}\right. \tag{1}$$

where ω_e is the output speed of the engine, and α is the throttle opening. Equation (1) shows that the three variables ω_e, α, T_e have a functional relationship and can be transformed into $\alpha = \text{throttle}(\omega_e, T_e)$ by deformation. Then, (1) can be expressed as

$$\dot{m}_f = \dot{m}_f(\omega_e, \text{throttle}(\omega_e, T_e)). \tag{2}$$

Equation (2) can be used to calculate fuel consumption through the speed and the torque of engine.

3.2 Driveline model

The power transfer route of the vehicle is shown in Fig. 5. The speed and torque of the engine are transmitted to the driving wheel after passing through the driveline. Because the clutch, transmission and other driveline components are rigid coupling, torsional deformation can be ignored.

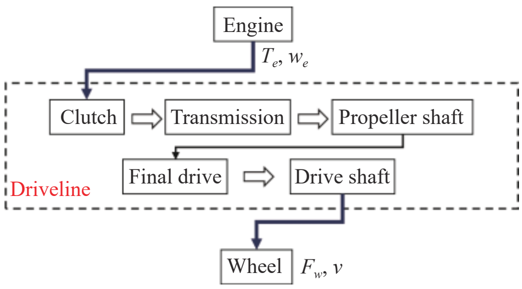


Fig. 5 Power transfer route of the vehicle

According to [25] and [31], in order to facilitate the analysis and calculation of the driveline model, the total moment of inertia of driveline can be equivalent to the tire, which is written as J_t. Therefore, when the transmission gear is set, the engine torque is transferred to the wheels through (3) and (4).

$$J_e\dot{\omega}_e = T_e - T_c, \tag{3}$$

$$T_w = i\eta T_c, \tag{4}$$

where T_c is the torque transmitted to the clutch, and T_w is the torque transmitted to the tire. $i = i_g i_0$ and $\eta = \eta_g \eta_0$ are respectively the total transmission ratio and efficiency of the driveline.

Then, the traction force transferred to the wheels F_w can be expressed as (5).

$$J_t\dot{\omega}_w = T_w - r_w F_w \tag{5}$$

where $\dot{\omega}_w$ is the angular acceleration of the engine crankshaft.

Further, the engine speed is converted into the driving speed by the driveline, as shown in (6).

$$v = \frac{\omega_e \cdot r_w \cdot 2\pi}{i \cdot 60} \tag{6}$$

3.3 Longitudinal dynamics model

The force analysis of the vehicle driving is shown in Fig. 6. Based on Newton's second law and the driveline model, the vehicle driving balance equation can be expressed as (7) [31].

$$m\frac{\mathrm{d}v}{\mathrm{d}t} = F_{\mathrm{w}} - mgf\cos\theta - \frac{C_{\mathrm{D}}A}{21.15}v^2 - mg\sin\theta. \tag{7}$$

Substitute (3)–(6) into (7) to obtain the longitudinal dynamics model of the vehicle, as shown in (8).

$$\frac{\mathrm{d}v}{\mathrm{d}t} = \frac{r_w}{J_t + m{r_w}^2 + i^2\eta J_e}\cdot \left[\left(i\eta T_e - r_w\left(\frac{C_{\mathrm{D}}A}{21.15}v^2 + mgf\cos\theta + mg\sin\theta\right)\right)\right]. \tag{8}$$

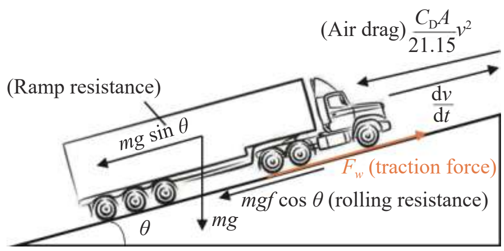


Fig. 6 Force analysis diagram of vehicle driving

4. Algorithm and principle of CPCC system

This section introduces the method of using DP to build the PCC algorithm, and proposes an RPSM algorithm to reduce the complexity of the DP algorithm from the perspective of road data preprocessing, which is to realize fast computation of the CPCC system. The PCC algorithm based on DP mainly studies cost function terms, constraint conditions and the dimension reduction optimization of DP.

4.1 PCC algorithm based on DP

The PCC algorithm is to solve the optimal control sequence of gear position and speed by using the information on road slope ahead of vehicle. Then, predictive cruise control of vehicle is realized by executing predictive control sequences.

4.1.1 Cost function

Since the state space of the PCC algorithm is divided into stages based on the distance domain, the cost function equation is calculated by discrete integration [25]. The cost function includes five items, namely fuel consumption, deviation of cruise reference speed, speed change rate, gear change penalty and throttle opening comparison, as shown in (9).

$$J = \sum_{k=1}^{N} w_1 \cdot \dot{m}_{fk} \cdot \frac{2\Delta s(k)}{v_k + v_{k+1}} + w_2 \cdot |(v_{\mathrm{ref}} - v_k)| + w_3 \cdot |(v_k - v_{k+1})| + w_4 \cdot \left|(g_{tk} - g_{t(k+1)})\right| + w_5 \cdot (\alpha_{ck} - \alpha_{pk}) \tag{9}$$

where $\dot{m}_{fk}$ is the fuel consumption rate of the stage k.

The variables α_{pk} and α_{ck} represent the throttle opening required by state transfer at stage k, in the PCC and CC control respectively.

The penalty factor of fuel consumption w_1 is used to ensure fuel economy of vehicle driving; the penalty factor of speed deviation w_2 is used to restrict the planned speed as close to the reference speed as possible, so as to maintain speed without excessive deviation from reference speed; the penalty factor of speed change rate w_3 is used to constrain the speed change of the two stages, and reduce rapid acceleration and deceleration to save fuel consumption; the penalty factor of gear penalty w_4 is used to reduce the phenomenon of gear jumping in the changing transmission at the front and rear stages; the penalty factor of throttle opening w_5 reduces the phenomenon of large throttle opening as far as possible by comparing CC throttle opening.

4.1.2 Constraints

In order to avoid vehicle speed deviating too much from the reference speed, the speed boundary of the state space is set as follows:

$$v_{\min} \leqslant v_k \leqslant v_{\max}. \tag{10}$$

According to the universal characteristic diagram of the engine, the efficient working area of the engine tends to be within certain engine speed and torque ranges, so there are the following constraints:

$$\begin{cases} \omega_{\min} \leqslant \omega_k \leqslant \omega_{\max} \\ T_{e\min} \leqslant T_{ek} \leqslant T_{e\max} \end{cases}. \tag{11}$$

According to the economic driving principle, rapid acceleration and deceleration will have a great impact on the fuel consumption of the vehicle, so there is an acceleration constraint.

$$a_{\min} \leqslant \frac{\mathrm{d}v}{\mathrm{d}t} = \frac{v}{3.6}\frac{\mathrm{d}v}{\mathrm{d}s} \leqslant a_{\max} \tag{12}$$

where s is the position of driving.

In the state space of DP, there are the lowest and highest cruise gears. Moreover, only the specific transmission gears are available under constraints of engine speed and vehicle speed. Therefore, the gear constraints as follows are indispensable.

$$\begin{cases} G_{\min} \leqslant G_k \leqslant G_{\max} \\ G_{v_k} = \{G \mid \omega_{\min} \leqslant \omega(v_k, G) \leqslant \omega_{\max}\} \end{cases} \tag{13}$$

4.1.3 Determination of optimal control sequence

In order to solve the optimal solution of the PCC problem through DP, the state space of DP needs to be established based on control variables firstly. Then, the state sequence that minimizes the cost function is the optimal control sequence of predictive driving on the road ahead. The state space and calculation process of the PCC algorithm based on DP are shown in Fig. 7.

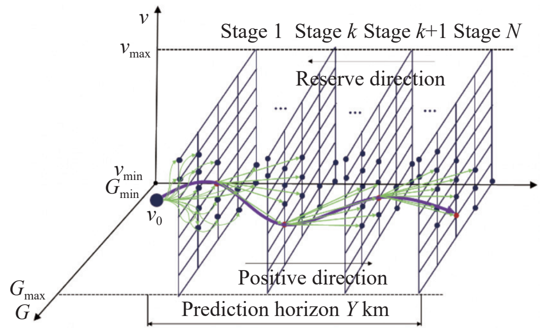


Fig. 7 State space and computational logic diagram of the PCC algorithm

Fig. 7 shows the state space constructed by speed and gear, and only part of the state points meets the requirements under constraints to reduce unnecessary calculations. By calculating (9) in the reverse direction, an optimal speed and gear sequence of the prediction horizon is finally obtained from the positive direction.

The stages of DP are divided according to the distance between the road points. Therefore, if the distance is too small when the prediction distance is fixed, the complexity calculation of DP is larger and the real-time performance is worse. Therefore, the pretreatment of road points plays an important role in the dimension reduction optimization of DP.

4.2 Dimensional reduction optimization of DP by RPSM

The solution complexity of DP in the CPCC system is mainly determined by prediction distance, the number of available gears,the constraint range and the dispersion interval of velocity. The computation time of DP will affect the iterative control precision of the system. The principle of the RPSM algorithm is to reduce the number of stages of DP without changing the predicted distance. For example, if the prediction distance is 2 km and the original road point is 20 m an interval, the PCC algorithm needs 100 stages of state space to be solved by DP once. If 2 km is divided into 10 stages according to certain rules, the calculation time can be reduced by 10 times, greatly reducing the closed-loop following control time of the system. The algorithm flow chart of RPSM is shown in Fig. 8.

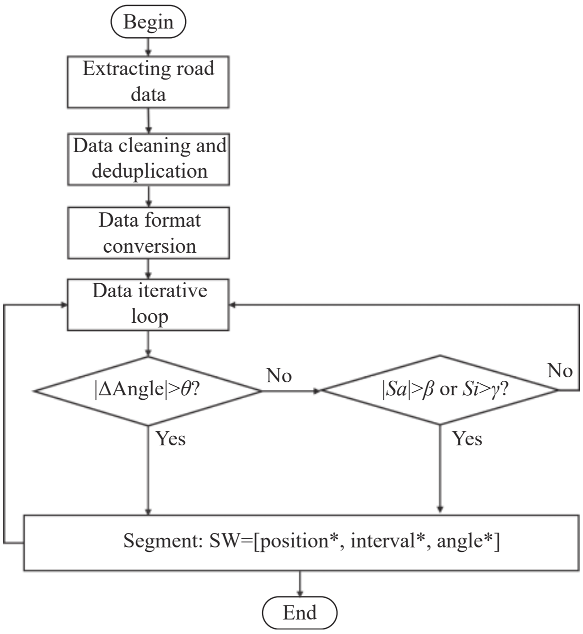


Fig. 8 Algorithm flow chart of RPSM

The RPSM is a pre-processing algorithm for road information, which outputs information on slope and road point intervals after segmentation and is used as input to the PCC algorithm. Firstly, the road information extracted from the map server is cleaned and reprocessed, and its format is transformed into the interface format required by the RPSM algorithm. Secondly, according to the sequence of road points, an iterative calculation is carried out based on three segmentation conditions. The condition 1 of $|\Delta\text{Angel}| > \theta$ is a slope constraint. In the iterative process of the road point, if the slope changes of the two original road points exceed θ, the segmentation is carried out at that point. The condition 2 of $|Sa| > \beta$ is a cumulative slope constraint to determine whether the cumulative slope value Sa meets the segmentation requirement when condition 1 is not satisfied. The condition 3 of $Si > \gamma$ is the constraint of road point spacing, indicating that after one segmentation, when the cumulative distance of the original road point exceeds γ, the segmentation is performed. In the iterative segmentation process, the first step is to judge the condition 1, and then 2 and 3, until the road sequence loop is completed. Finally, the segmented road-point sequence is output according to the format required by the PCC algorithm.

4.3 Control method and process of CPCC system

The CPCC system adopts the control method of iterative update and rolling distance domain optimization. The optimal speed and gear are recalculated according to a certain distance threshold r, when the CPCC mode is started. The detailed control process is shown in Fig. 9.

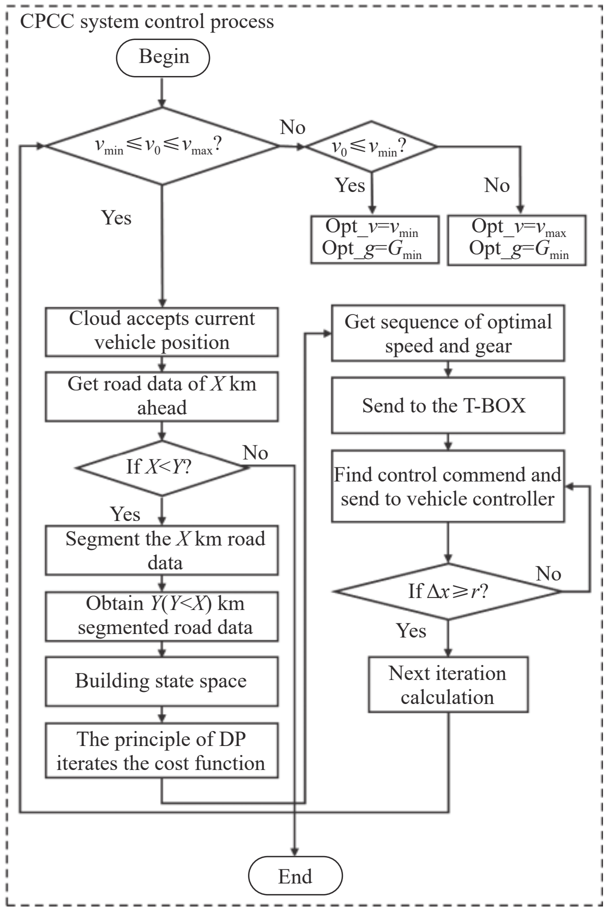


Fig. 9 Control process and method of CPCC system

Step 1 When the cloud platform detects that each signal of the vehicle and T-BOX meet the requirements for entering the CPCC system (see Section 3 vehicle platform introduction for details), the CPCC mode will be activated to replace the CC mode.

Step 2 If the current initial speed $v_0 \leqslant v_{max}$, the recommended speed and gear are planned according to the minimum set value. Conversely, if the initial speed is greater than the maximum, the set maximum value of gear and speed is output.

Step 3 When the condition of $v_{min} \leqslant v_0 \leqslant v_{max}$ is met, the cloud accepts the location information of the vehicle, and extracts the map data of X km in front of the current road from the map server (long distance).

Step 4 When the obtained road data range X is more than the optimization range Y, it means that the map data is missing and the CPCC mode should be exited, otherwise it runs and enters Step 5.

Step 5 The X km road is segmented by the RPSM algorithm. Then, based on the current position of the vehicle, the segmented road of Y km is obtained to optimize the optimal control strategy in the current prediction range.

Step 6 The state space of DP is constructed based on speed, gear position and predicted distance.

Step 7 Based on the iterative calculation principle of DP, the cost function is calculated from back to front within the range of constraints, and the minimum cost path is finally obtained. Then, the optimal speed and gear sequence is acquired.

Step 8 Based on the real-time position of the vehicle, the corresponding control command of speed and gear are found in the optimal control sequence and sent to the vehicle controller units.

Step 9 Judge whether the driving range of the vehicle exceeds r m in one iteration cycle. If not, the optimization control instruction is still sent according to Step 6. If so, the next iteration is performed and return to Step 2.

5. Simulation and analysis

The simulation experiment is divided into two parts. First, an actual road is used to verify the segmentation effect of RPSM. Then, the simulation model of the CPCC system, which is built based on Matlab/Simulink and Trucksim, is simulated by using the same road.

5.1 Effect verification of the RPSM algorithm

The simulation road selected is 36 km from the Yiyuan server to the Zhuge server on the G22 Expressway in Shandong, China. This road has a significant change in slope and elevation, which meets the validation needs. The aerial view of this road is shown in Fig. 10.

Fig. 10 Aerial view of the simulated road

By pre-processing this road offline based on the RPSM algorithm, the segmentation effect is shown in Fig. 11. It shows the original and segmentary road points in the view of road elevation map, with local splitting effects shown at certain key locations on the road. It is obvious that the segmented road-points basically retain the characteristic information of the original road-points in the locations of slope top and slope bottom. From the overall perspective, the original road with a total of 1 731 points are reasonably divided into 155, with a road-point reduction rate of 91%, shown in Fig. 12.

The advantages of the RPSM algorithm are that it not only greatly reduces the number of road segments, but also preserves the road characteristic points well. Thus, it can prolong the road prediction distance of CPCC system, without increasing time dimension. Furthermore, by combining the algorithm with the CPCC system, the computing power of the CCP can be greatly freed up and the number of PCC-equipped vehicles managed by CCS can be further broadened.

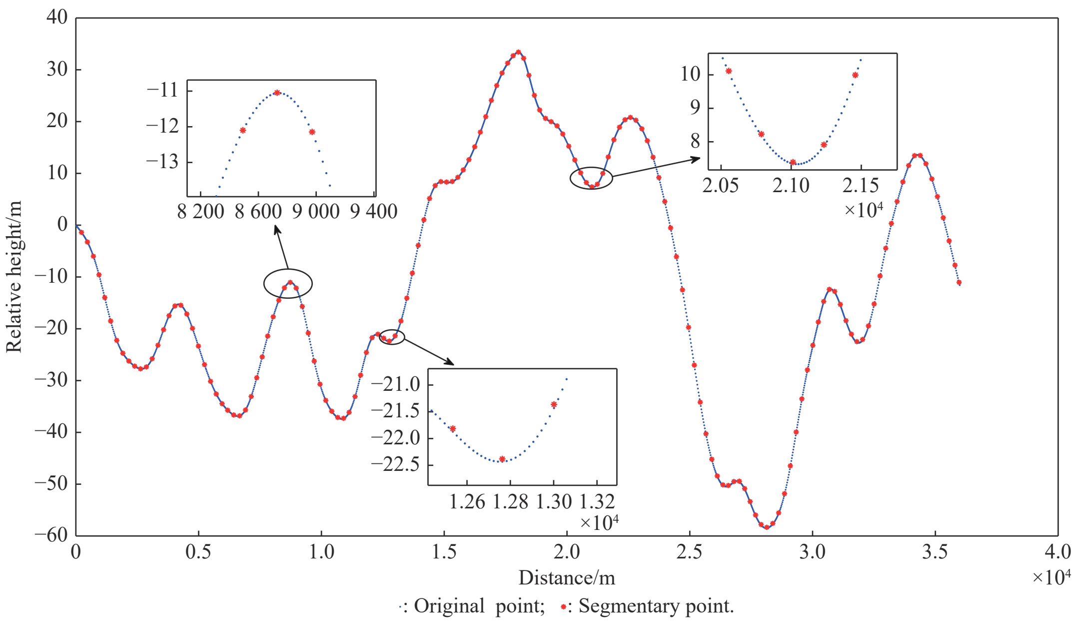


Fig. 11 Segmentation effect of the RPSM algorithm

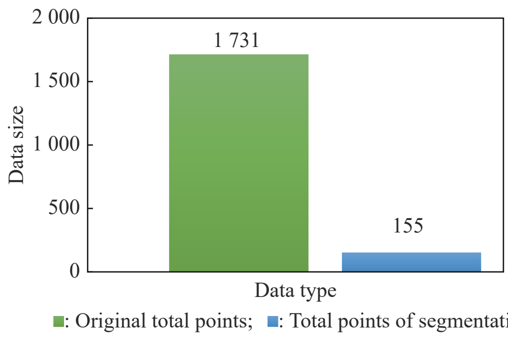


Fig. 12 Histogram of data volume comparison (roadpoint reduction rate of 91%)

5.2 Simulation analysis of CPCC system

The CPCC system is a combination of the PCC and RPSM algorithms based on the CCS. The same experimental road as shown in Subsection 5.1 is used to verify its control effectiveness.

5.2.1 Simulation model

The simulation model is shown in Fig. 13, including four modules. Module 1 is the map update module, which contains the original map data and the RPSM algorithm, capable of providing segmented map data to the PCC algorithm. Module 2 is a PCC algorithm solver that uses predicted slope information to solve for the optimal gear and speed based on the DP principle. In practice, Module 2 is located on the edge cloud in the CCP and caters to the real-time calculation and the delivery of commands. Module 3 is used to simulate the control instruction parsing function of the vehicle-side T-BOX, which can extract control commands in a predictive control sequence based on the vehicle's travel position. Module 4 is a Trucksim S-function, which simulates the driving process of vehicles on the actual road by configuring vehicle parameters and road information. The four modules are connected to form a closed loop to simulate the calculation and control process of the CPCC system under the layered architecture.

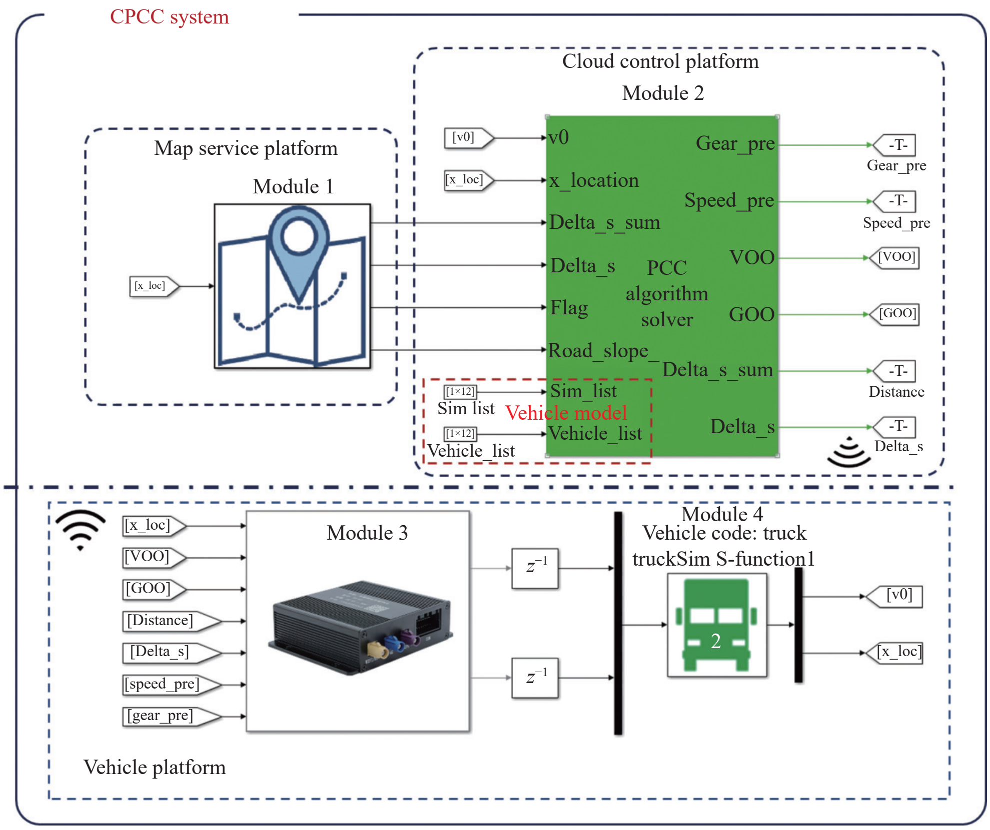


Fig. 13 Simulation model of the CPCC system

5.2.2 Parameter setting

The debugging of cost function weight is roughly divided into three steps. The first step is to analyze the trend impact of each parameter on fuel consumption, and find out several cost items that have an obvious impact on fuel consumption. The second step is to determine the approximate boundary range of each parameter by the method of controlling the variables. The third step is to debug the weight with fuel consumption, gear change frequency and speed change rate as the evaluation objectives. According to the above debugging method, weight settings are shown in Table 3.

Table 3 Weight parameter of the cost function

Weight factor	value
w_1	1
w_2	0.025
w_3	0.01
w_4	0.02
w_5	1

Simulation parameter settings of the CPCC system are shown in Table 4.

Table 4 Simulation parameters of the CPCC system

Parameter	Symbol	Value
Predictive horizon/km	Y	3
Segmentation horizon/km	X	9
Iteration distance/m	r	200
Discrete interval /(m/s)	Δv	0.2
Reference velocity /(km/h)	v_{ref}	70
Engine min rpm /(r/min)	$\omega_{\min}$	1 000
Engine max rpm /(r/min)	$\omega_{\max}$	1 800
Min gear position	$G_{\min}$	10
Max gear position	$G_{\max}$	12
Maximum acceleration /(m/s^2)	$a_{\max}$	0.4
Minimum acceleration /(m/s^2)	$a_{\min}$	−0.4

5.2.3 Results analysis

Through the co-simulation of Trucksim and Simulink, the heavy truck equipped with the CPCC system is simulated and verified on the real road. The CC mode and the classic DP algorithm for a velocity search (DP-V) are respectively as the comparative algorithm (the DP-V is the self-defined abbreviation of the classic DP-based speed search algorithm in [32]). The reference speed, truck model and road condition are the same as those of the CPCC system. The general diagram of simulation results is shown in Fig. 14, which shows the simulation results for the full road. Fig. 15(a) and Fig. 15(b) show the results of typical sections of 0−10 km and 25−32 km captured from Fig. 14, in order to make comparative analysis of planning effect in detail.

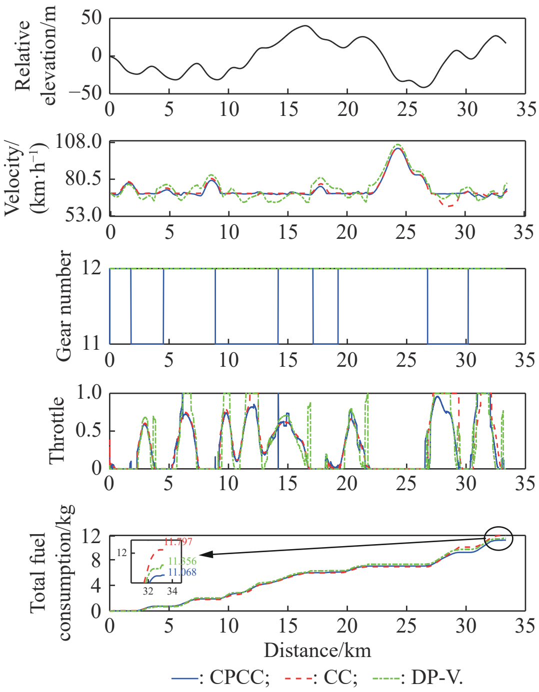
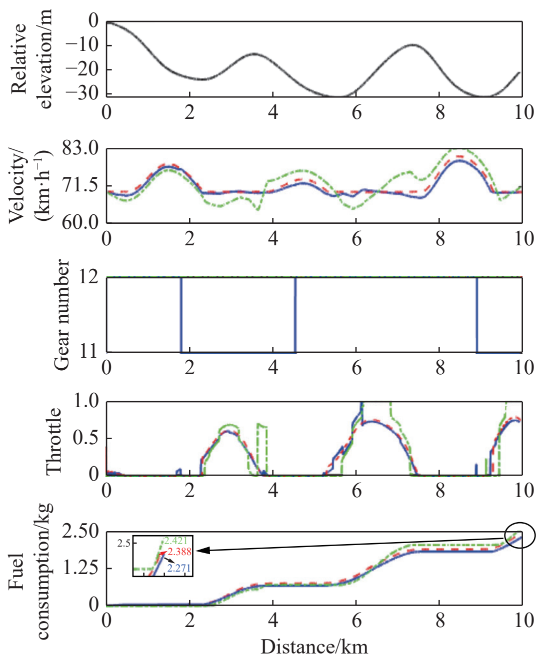
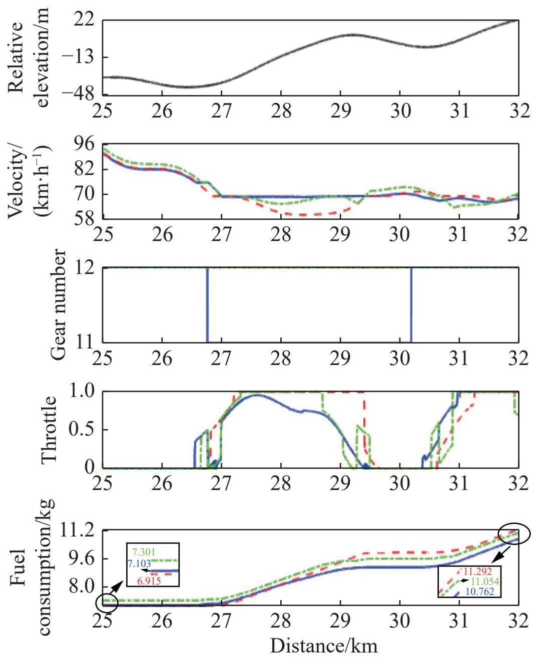


Fig. 14 General diagram of simulation results in the entire section



(a) Diagram of typical Section 1



(b) Diagram of typical Section 2

—: CPCC; - - -: CC; -·-·-: DP-V.

Fig. 15 Simulation result sub-diagram in typical section

As shown in Fig. 14, compared with the CC and DP-V, the CPCC system can adjust the vehicle speed moderately and downshift gear predictively. Obviously distinct from the DP-V, the speed curve planned by the PCC algorithm is relatively much smoother, which demonstrates better driving stability. Moreover, the PCC algorithm also ensures the dynamic performance of ramp driving by adjusting optimal gear, which is particularly important for heavy trucks. The DP-V algorithm can achieve the purpose of fuel-saving by optimizing the speed, with the fuel-saving rate of 3.74 % compared with CC. However, since the DP-V does not consider the gear optimization, the full throttle opening often occurs, which will cause insufficient combustion of fuel and increased contamination. This is also an important reason why the fuel-saving rate of DP-V is lower than that of PCC. Overall, under the condition of approximately equal driving time, the PCC algorithm shows higher ramp adaptability than CC and DP-V and obtains the fuel-saving rate of 6.17% in comparison with CC. Specific simulation results are shown in Table 5.

Table 5 Simulation results of the various algorithms

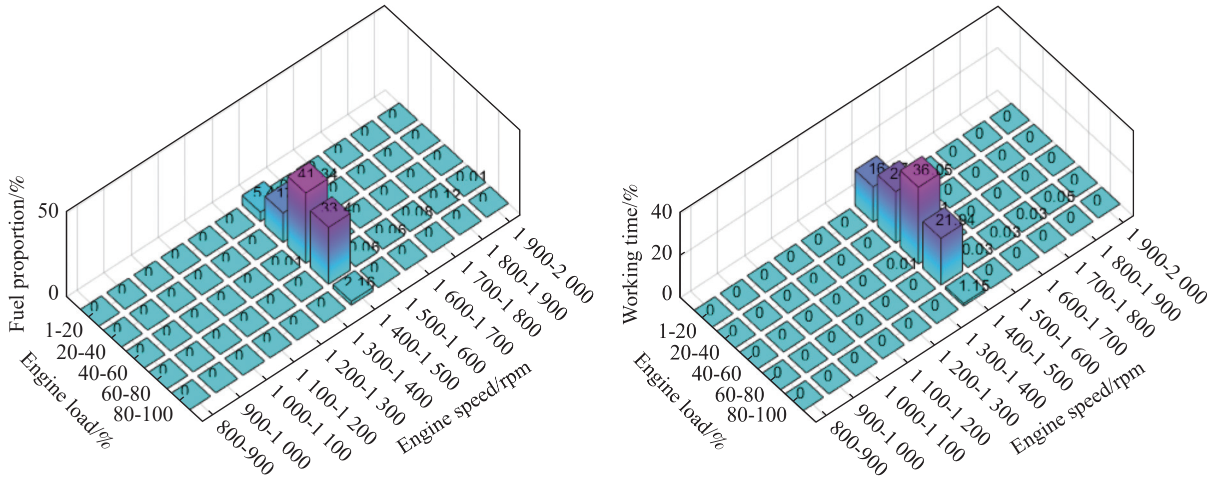
Type	Distance/km	Time/s	Fuel consumption/kg	Fuel-saving rate/%
CC	34	1 648.3	11.797	–
DP-V	34	1 655.4	11.356	3.74
CPCC	34	1 658.8	11.068	6.17

In Fig. 15(a), when the truck is going down the long slope at 0.5 km and 7.5 km, the speed is reduced in advance to avoid a high-speed impact. During the uphill and downhill journey from 1.9 km to 4.6 km, the truck maintains power and auxiliary braking by downshifting and maintaining gear position on the 11th gear. Through the adjustment of speed and gear, the fuel-saving rate of the first 10 km section is 4.9 % in comparison with the CC. In addition, although the speed regulation range of the DP-V is more obvious than PCC, the fuel saving of the DP-V is not better than the CC. On the contrary, the PCC algorithm shows good performance by collaborative optimization of gear and vehicle speed.

As can be seen from Fig. 15(b), when the vehicle climbs a long uphill at 26.5 km, it downshifts gear at the bottom of the slope in advance and climbs at a uniform speed, which avoids the energy loss caused by shifting in the middle of the ramp and conforms to the fuel-saving driving law. In addition, the throttle opening of CPCC is significantly lower than that of CC, and the fuel-saving rate of whole-course reaches 14.3 %, which fully reflects the importance of gear prediction. By analyzing the speed curve, it can be found that the speed adjustment ability of the three algorithms is limited under the condition of long uphill. However, the PCC algorithm improves both dynamic performance and economy through predictive gear adjustment, which is an obvious merit.

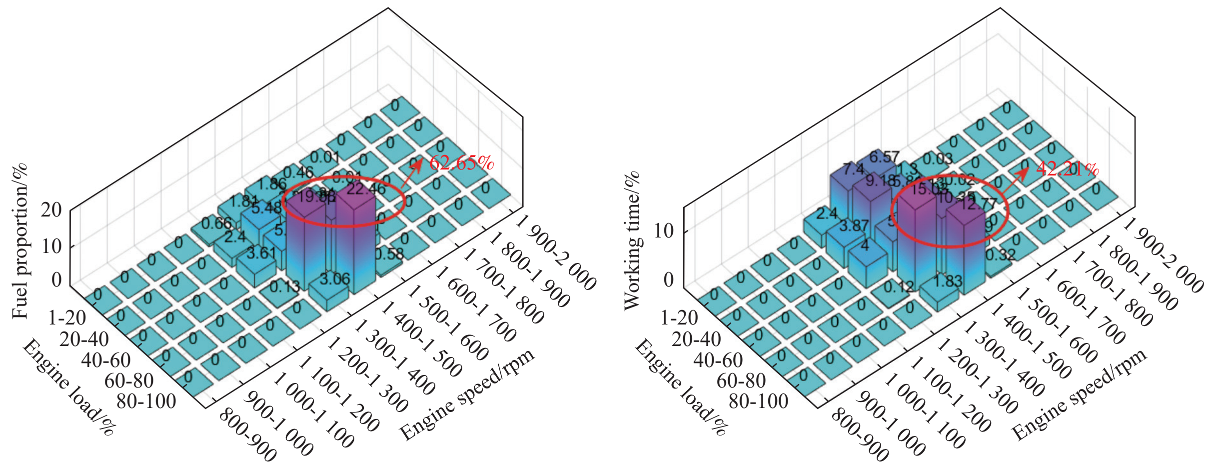
The curve analysis of the simulation results is carried out above, and the mechanism analysis of the control effect will be carried out below. Through calculating statistically of the engine speed, torque, and fuel consumption rate in the CC, DP-V and CPCC modes, the engine working interval distribution under the three modes is obtained as shown in Figs. 16−18.

Comparing Fig. 16 and Fig. 18, it can be found that the fuel proportion and working time of engine load in the range of 40%–80% in the CC mode are more than 74% and 57%, respectively, which are significantly higher than those in the CPCC mode. In addition, in the CC mode, the working time of the engine throttle opening close to 100% is significantly higher than that in the CPCC mode, and it occurs mostly in heavy load climbing conditions. This situation will cause the engine to work badly and the combustion is insufficient, which is not conducive to prolonging the working life of the engine and its accessories. In DP-V mode seen as Fig. 17, the working range of the engine is obviously moved to the high efficiency region (the highly efficient speed range is [1 200 1 400]) than in the CC mode, which is an important reason for its fuel saving compared with the CC mode. However, the working time and fuel proportion of engine load in the 60%−100% range are 42.21% and 62.65 % respectively, which are significantly higher than those of the CPCC mode. Moreover, in this case, although the speed range is biased towards the efficient region, the efficient load range of the engine (the highly efficient engine load range is [40% 60%]) is significantly smaller than that of the CPCC mode.



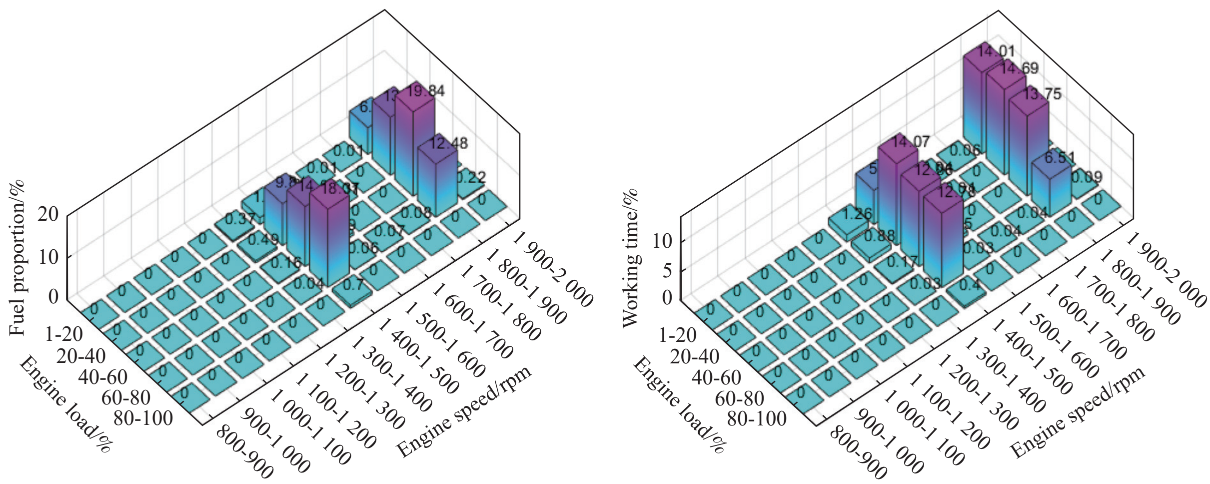
(a) Distribution of fuel proportion (b) Distribution of working time

Fig. 16 Interval distribution of engine working in the CC mode



(a) Distribution of fuel proportion (b) Distribution of working time

Fig. 17 Interval distribution of engine working in the DP-V mode



(a) Distribution of fuel proportion (b) Distribution of working time

Fig. 18 Interval distribution of engine working in the CPCC mode

Based on the above analysis, it can be fully explained that the PCC algorithm has a stronger ability to adjust the engine to move to the high efficiency region through the collaborative optimization of the gear and speed than the CC and DP-V modes, thereby reducing the fuel consumption and improving the comprehensive performance of the heavy truck.

6. Conclusions

In this paper, a layered architecture of the CPCC system based on the ICV CCS is proposed to reasonably arrange the PCC algorithm on the vehicle platform and CCP. In order to simultaneously meet the dynamic and fuel-saving requirements of heavy trucks when driving on ramps, a PCC algorithm for the co-optimization of vehicle speed and gear has been proposed based on DP. Furthermore, from the perspective of road point segmentation, an RPSM algorithm is proposed to reduce the dimension of the DP algorithm to diminish the computational complexity of the PCC algorithm. The simulation results show that the RPSM algorithm reduces the original road points of 1 731 to 155, with a road-point reduction rate of 91%. Also, the segmented road retains the road feature information well, which can reduce the calculation error of the PCC algorithm and ensure the accuracy. For the simulation of the CPCC system, it can be found that the proposed PCC algorithm can adaptively adjust the speed and gear position according to the change of slope, with a fuel-saving rate of 6.17% in contrast to the CC mode, which can satisfy the dynamic and economic performance of heavy trucks under different ramp conditions. Also, compared with the similar algorithm of DP-V, the PCC algorithm can make the engine operate more in the efficient zone by cooperatively optimizing the gear and speed. Therefore, the CPCC system and architecture proposed in this paper can fulfill the fast computability of the algorithm and has a high fuel saving capability, which has a strong prospect for the industrial application of the ICV CCS.

Due to the limitations of the simulation model, this paper does not verify the cruise state transition logic in the CPCC system. In the future, the proposed system will be deployed on the cloud control platform, and the state transition logic will be embedded in T-BOX to complete the real vehicle verification. In addition, a comparative experiment between the CPCC and human driver will be added to fully verify the effectiveness of the algorithm.

References

[1] XIA Y Q. Cloud control systems. IEEE/CAA Journal of Automatica Sinica, 2015, 2(2): 134–142.

[2] CHU W B, WUNIRI Q, DU X P, et al. Cloud control system architectures, technologies and applications on intelligent and connected vehicles: a review. Chinese Journal of Mechanical Engineering, 2021, 34(1): 1–23.

[3] YANG S C, ZHANG Z J, CAO R, et al. Implementation for a cloud battery management system based on the CHAIN framework. Energy and AI, 2021, 5: 100088.

[4] XU X L, WU J X, YANG G, et al. Low-power task scheduling algorithm for large-scale cloud data centers. Journal of Systems Engineering and Electronics, 2013, 24(5): 870–878.

[5] XU X L, ZHANG Q T, MOU Y Q, et al. Server load prediction algorithm based on CM-MC for cloud systems. Journal of Systems Engineering and Electronics, 2018, 29(5): 1069–1078.

[6] LI K Q, LI J W, CHANG X Y, et al. Principles and typical applications of cloud control system for intelligent and connected vehicles. Journal of Automotive Safety and Energy, 2020, 11(3): 261–275. (in Chinese)

[7] HELLSTROM E. Explicit use of road topography for model predictive cruise control in heavy trucks. Linkoping: Linkoping University, 2005.

[8] LATTEMANN F, NEISS K, TERWEN S, et al. The predictive cruise control–a system to reduce fuel consumption of heavy duty trucks. SAE Transactions, 2004, 113(2): 139–146.

[9] LI K Q, CHANG X Y, LI J W. Cloud control system for intelligent and connected vehicles and its application. Automotive Engineering, 2020, 42(12): 1595–1605. (in Chinese)

[10] XIA Y Q. From networked control systems to cloud control systems. Proc. of the 31st Chinese Control Conference, 2012: 5878–5883.

[11] KHATTAB A, ABDELGAWAD A, YELMARTHI K. Design and implementation of a cloud-based IoT scheme for precision agriculture. Proc. of the 28th International Conference on Microelectronics, 2016: 201–204.

[12] ET-TAIBI B, ABID M R, BOUMHIDI I, et al. Smart agriculture as a cyber physical system: a real-world deployment. Proc. of the 4th International Conference on Intelligent Computing in Data Sciences, 2020: 1–7.

[13] HU L, MIAO Y M, WU G X, et al. iRobot-Factory: an intelligent robot factory based on cognitive manufacturing and edge computing. Future Generation Computer Systems, 2019, 90: 569–577.

[14] CHEN G G, WANG P, FENG B, et al. The framework design of smart factory in discrete manufacturing industry based on cyber-physical system. International Journal of Computer Integrated Manufacturing, 2020, 33(1): 79–101.

[15] XIA Y Q, YAN C, WANG X J, et al. Intelligent transportation cyber-physical cloud control systems. Acta Automatica Sinica, 2019, 45(1): 132–142.

[16] OZATAY E, ONORI S, WOLLAEGER J, et al. Cloud-based velocity profile optimization for everyday driving: a dynamic-programming-based solution. IEEE Trans. on Intelligent Transportation Systems, 2014, 15(6): 2491–2505.

[17] HAMEDNIA A, MURGOVSKI N, FREDRIKSSON J. Predictive velocity control in a hilly terrain over a long look-ahead horizon. IFAC-PapersOnLine, 2018, 51(31): 485–492.

[18] SCHWARZKOPF A B, LEIPNIK R B. Control of highway vehicles for minimum fuel consumption over varying terrain. Transportation Research, 1977, 11(4): 279–286.

[19] LIN Y C, HSU H C, KUO I C. An eco-cruising control systems using nonlinear predictive control approach. Proc. of the International Automatic Control Conference, 2016: 56–94.

[20] WEIBMANN A, GORGES D, LIN X H. Energy-optimal adaptive cruise control based on model predictive control. IFAC-PapersOnLine, 2017, 50(1): 12563–12568.

[21] GUO L L, GAO B Z, GAO Y, et al. Optimal energy management for HEVs in eco-driving applications using bi-level MPC. IEEE Trans. on Intelligent Transportation Systems, 2016, 18(8): 2153–2162.
[22] KUMAR R, IVANTYSYNOVA M. An instantaneous optimization based power management strategy to reduce fuel consumption in hydraulic hybrids. International Journal of Fluid Power, 2011, 12(2): 15–25.
[23] HUANG Y J, YIN C L, ZHANG J W. Design of an energy management strategy for parallel hybrid electric vehicles using a logic threshold and instantaneous optimization method. International Journal of Automotive Technology, 2009, 10(4): 513–521.
[24] HELLSTROM E, ASLUND J, NIELSEN L. Design of an efficient algorithm for fuel-optimal look-ahead control. Control Engineering Practice, 2010, 18(11): 1318–1327.
[25] HELLSTROM E, IVARSSON M, NIELSEN L. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. IFAC Proceedings Volumes, 2007, 40(10): 439–446.
[26] CONG X Y, ZHANG Y G, WANG C P, et al. Look-ahead gear-shifting strategy on ramps for heavy trucks with automated mechanical transmission. Advances in Mechanical Engineering, 2019, 11(1): 1–13.
[27] HELLSTROM E, FROBERG A, NIELSEN L. A real-time fuel-optimal cruise controller for heavy trucks using road topography information. Proc. of the SAE World Congress & Exhibition, 2006: 1–8.
[28] LIN Y C, NGUYEN H L T, BALAS V E, et al. Adaptive prediction-based control for an ecological cruise control system on curved and hilly roads. Journal of Intelligent & Fuzzy Systems, 2020, 38(5): 6129–6144.
[29] XU S B, LI S E, CHENG B, et al. Instantaneous feedback control for a fuel-prioritized vehicle cruising system on highways with a varying slope. IEEE Trans. on Intelligent Transportation Systems, 2016, 18(5): 1210–1220.
[30] QU L H. Ecological cruising control strategy design of electric vehicle based on prediction of road slopes. Nanjing: Southeast University, 2019. (in Chinese)
[31] GILLESPIE T D. Fundamentals of vehicle dynamics. New York: SAE International, 2021.
[32] ZHANG J, JIN H. Optimized calculation of the economic speed profile for slope driving: based on iterative dynamic programming. IEEE Trans. on Intelligent Transportation Systems, 2022, 23(4): 3313–3323.

Biographies

LI Shuyan was born in 1972. She is currently an associate professor at the College of Engineering, China Agricultural University. Her research interests are intelligent and connected vehicle, predictive cruise control, and intelligent testing of agricultural machinery and equipment.
E-mail: lishuyan@cau.edu.cn

WAN Keke was born in 1998. He received his B.E. degree from Henan University of Engineering. He is pursuing his master's degree in vehicle engineering at the College of Engineering, China Agricultural University, Beijing, China. He is engaged in research work in the School of Vehicle and Mobility, Tsinghua University. His research interests include cloud-based predictive cruise control, vehicle-road-cloud collaborative control and cloud control architecture.
E-mail: wankeke@cau.edu.cn

GAO Bolin was born in 1986. He is now an associate research professor at the School of Vehicle and Mobility, Tsinghua University. His research interests include the theoretical research and engineering application of the dynamic design and control of intelligent and connected vehicles, especially about the collaborative perception and tracking method in cloud control system, intelligent predictive cruise control system on commercial trucks with cloud control mode, as well as the test and evaluation of intelligent vehicle driving system.
E-mail: gaobolin@tsinghua.edu.cn

LI Rui was born in 1997. He is currently a master student at the College of Engineering, China Agricultural University, Beijing, China. He is engaged in research work in the School of Vehicle and Mobility, Tsinghua University. His research interest is adaptive cruise control.
E-mail: lirui@cau.edu.cn

WANG Yue was born in 1988. He received his Ph.D. degree in vehicle engineering from Jilin University, Jilin, China, in 2020. He is currently an assistant research fellow with the School of Vehicle and Mobility, Tsinghua University, Beijing, China. His research interests include energy management, clean energy vehicles, intelligent vehicles, and eco-driving.
E-mail: wyue@mail.tsinghua.edu.cn

LI Keqiang was born in 1963. He is an academician of the Chinese Academy of Engineering, and a professor at the School of Vehicle and Mobility, Tsinghua University. He is also the Director of the State Key Laboratory of Automotive Safety and Energy and a Senior Member of SAE-China. His research interests include connected and intelligent vehicles, vehicle dynamics and control.
E-mail: likq@mail.tsinghua.edu.cn