Energy Conversion and Management: X 20 (2023) 100414

Contents lists available at ScienceDirect

Energy Conversion and Management: X

journal homepage: www.sciencedirect.com/journal/energy-conversion-and-management-x

Data-driven bi-level predictive energy management strategy for fuel cell buses with algorithmics fusion

Menglin Li [a], Haoran Liu [a], Mei Yan [a,*,1], Jingda Wu [b], Lisheng Jin [a], Hongwen He [c,1]

[a] School of Vehicle and Energy, Yanshan University, Qinhuangdao 066004, China
[b] School of Mechanical and Aerospace Engineering, Nanyang Technological University 308232, Singapore
[c] National Engineering Laboratory for Electric Vehicles, School of Mechanical Engineering, Beijing Institute of Technology 100081, China

ARTICLE INFO

Keywords:
Energy management
Multi-prediction horizons
Fuel cell buses
Data-driven
Reinforcement learning

ABSTRACT

This paper aims to answer how to effectively integrate the data-driven method into the traditional predictive energy management algorithm rather than replacing it outright. Given the challenge of selecting an appropriate prediction horizon for predictive energy management, this study seeks to bridge traditional predictive energy management with machine learning approaches, thereby presenting a novel bi-level predictive energy management strategy for fuel cell buses with multi-prediction horizons. In the upper layer, the core parameter, prediction horizon, of the traditional model predictive control energy management framework is optimized using two distinct data-driven methods. The first method employs deep learning to establish a mapping relationship between the vehicle states and the optimal prediction horizon through deep neural networks. The second method utilizes reinforcement learning to obtain the best prediction horizon under varying vehicle states through intelligent agent exploration. In the lower level, predictive energy management is performed on fuel cell buses based on optimization levels. Finally, the proposed strategy is validated using test data from actual fuel cell buses. The results demonstrate that two data-driven methods, based on the optimal ΔSoC approximation and the deep reinforcement learning, can select the appropriate prediction horizon more conducive to energy saving according to the vehicle states. Regarding energy consumption, the multi-horizon predictive energy management based on deep reinforcement learning exhibits a remarkable reduction in energy consumption by 7.62 %, 4.55 %, 4.60 %, and 7.80 %, when compared with the predictive energy management employing fixed prediction horizons of 5 s, 10 s, 15 s, and 20 s, respectively. Furthermore, it outperforms the multi-horizon predictive energy management approach based on the optimal ΔSoC approximation by 3.59 %.

Introduction

Background

As an essential carrier of hydrogen energy application, the development of fuel cell buses (FCBs) is a crucial part of realizing energy structure reform [1] and the application of sustainable energy [2]. Therefore, the development of FCBs has important practical significance for countries committed to reducing carbon emissions [3]. Due to the hysteresis of the fuel cell (FC) power output response, to adapt to complex vehicle driving conditions, an additional energy source that can provide transient response power is usually added to the fuel cell system [4], such as lithium batteries or supercapacitors [5]. The combination of multiple power sources makes the operating mode of the power system more diverse [6]. Therefore, effective energy conversion and management have become the key to fuel cell systems to improve energy efficiency, extend component life, and enhance adaptability to operating conditions.

Literature review

The research on energy management strategies for fuel cell vehicles has been enhanced with the attraction of fuel cell vehicles in recent years. In addition, the multi-source powertrain structure makes it easy to migrate the traditional energy management strategy from hybrid electric vehicles to fuel cell vehicles. At present, the energy management strategies (EMSs) of fuel cell hybrid electric vehicles mainly include

* Corresponding author.
E-mail address: yanmei@ysu.edu.cn (M. Yan).
[1] The two authors have the same contribution to this study.

https://doi.org/10.1016/j.ecmx.2023.100414
Received 2 March 2023; Received in revised form 22 May 2023; Accepted 22 June 2023
Available online 28 June 2023

Nomenclature

DB-PEMS	data-driven bi-level predictive EMS
DNN	deep neural network
DP	dynamic programming
DQN	deep Q-network
DRL	deep reinforcement learning
EMS	energy management strategy
FC	fuel cel
FCB	fuel cell bus
F_t	driving force
F_f	rolling resistance
F_w	air resistance
F_i	ramp resistance
F_j	acceleration resistance
T_{tq}	driving torque
i_0	transmission ratio of the final reduction
η_T	mechanical efficiency of the drive train
r	wheel radius
G	vehicle gravity
E_{PS}	energy consumption of power system
E_{bat}	energy provided by battery
f	rolling resistance coefficient
α	road slope
C_D	air resistance coefficient
A	windward area
v	vehicle speed
δ	rotating mass conversion coefficient
m	vehicle mass
V_{bat}	output voltage of the battery
E_{bat}	open circuit voltage of the battery
V_{pol}	voltage drop caused by polarization
I_{bat}	battery current
R_{bat}	battery ohmic resistance
C_{pol}	battery polarization resistance
R_{pol}	battery polarization capacitance
P_{bat_load}	load power of the battery
u	control variable
C_{bat}	initial capacity of the battery
T_m	torque of the drive motor
n_m	rotational speed of the drive motor
E_{bat}	energy consumption of the battery
ΔP_{fc}	change in output power of FCS
χ	weight coefficients of equivalent hydrogen energy consumption
ζ	weight coefficients of the fluctuation degree of power output of the FCS
φ	conversion coefficient of electricity consumption into hydrogen consumption
f_{DNN_SoC}	DNN-based optimal ΔSoC predictor
X	input of DNN-based optimal ΔSoC predictor
Y	output of DNN-based optimal ΔSoC predictor
hp	prediction horizon
hp^*	optimal prediction horizon
f_{BiLSTM_NN}	BiLSTM-based speed predictor
Input	input of BiLSTM-based speed predictor
Output	output of BiLSTM-based speed predictor
FCS	fuel cell system
LSTM	Long Short Term Memory
MAE	mean absolute error
MPC	model predictive control
PEMFC	proton exchange membrane fuel cell
PEMS	predictive energy management strategy
SoC	State of charge
SoC_DP	SoC obtained via DP
SoC_ΔSoC	SoC obtained by accumulating the predicted ΔSoC
P	resistance power
P_m	output power of the drive motor
P_{req}	demand power of the power system
P_{bat}	energy conversion efficiency of the battery
P_{fc}	output power of the battery
η_{fc}	efficiency of the FCS
η_{bat}	output power of the FCS
η_{DC}	DC/DC converter efficiency
η_m	drive motor efficiency
SOC_{int}	initial SOC
E_{fc}	energy provided by fuel cell system
V_{cell}	voltage of individual PEMFC
E_0	ideal voltage source
I_{fc}	output current of FC
R_{act}	resistance of activation loss
R_{ohmic}	resistance of ohmic loss
R_{con}	resistance of concentration loss
V_{act}	activation loss voltage
V_{ohmic}	ohmic loss voltage
V_{con}	concentration loss voltage
V_{stack}	voltage of PEMFC stack
N_{cell}	number of fuel cell cells
P_{fc_load}	required power of FCS
P_{fc_out}	total output power of FCS
P_{fc_aux}	required power of the FCS accessories
V_{ec}	thermal voltage
x	state variable
w	disturbance variable
J	optimization fuction
E_{fc}	energy consumption of the FCS
s	driving distance
a	Vehicle acceleration
P	caculated demand power of vehicle
Q	action-value function of DRL
R	reward function value
sta_t	state variables of DRL
act_t	action variables of DRL
θ	weight parameter of the DQN network
β	weight coefficient of the DQN network
Act	action set of DRL
S	state set of DRL
p_a	accelerator pedal signal
p_d	deceleration pedal signal

rule-based EMSs, which are subdivided into thermostat control [7], power follow command [8], filter control [9], state machine control [10], fuzzy control [11–13], synovial control [9], etc. Optimization-based EMSs are divided into linear programming (LP) [14], dynamic programming (DP) [15], convex optimization [16], Pontryagin minimum principle [17,18], equivalent consumption minimization strategy [19] and model predictive control (MPC) [20], etc. The advantages and disadvantages of the main algorithms are shown in Fig. 1.

Get benefits from the continuous progress of computer computing power and prediction methods based on artificial intelligence, the prediction accuracy of the future states of vehicles in complex environments is getting higher [1]. In conjunction with the real-time and robustness advantages of MPC, the optimal local solution of EMS based on MPC is approaching the optimal global one. The system's performance is

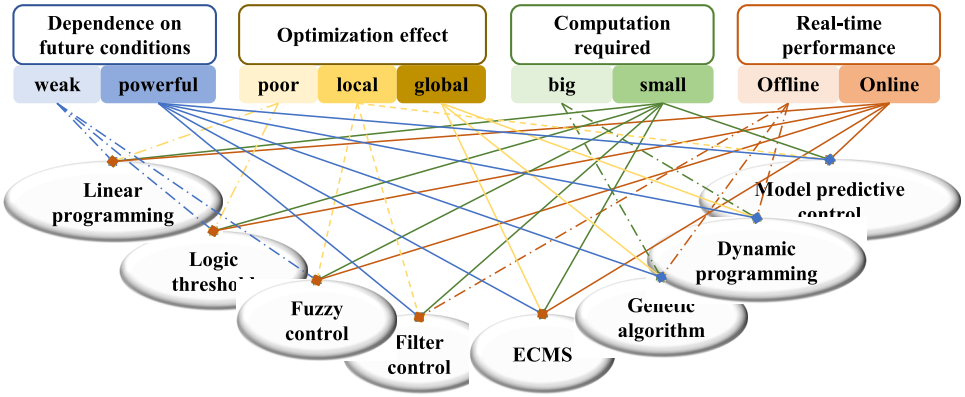


Fig. 1. Advantages and disadvantages of EMSs commonly used.

contingent on accurately predicting future operating conditions, including speed and power demand, as this is a prerequisite for developing effective energy management strategies. Precise predictions of future power demand can increase strategic time allowance, thus achieving energy conservation and optimizing the lifespan of power sources. Consequently, accurately predicting driving conditions remains a hot and challenging issue.

The accuracy and adaptability of future working conditions need improvement due to early limitations in information acquisition and data mining technology capabilities. C Sun et al. explored various prediction methods, including exponential prediction, Markov chain prediction, and neural network prediction, and discussed the impact of tuning parameters on prediction effectiveness [21]. Kolmanovsky Ilya et al. analyzed the impact of deterministic models, such as autoregressive moving averages and nonlinear autoregressive models, on speed prediction accuracy [22]. Khajepour et al. proposed an average-based working condition predictor [23]. Vatanparvar et al. trained a novel context-aware NARX model based on the historical behaviour of real drivers, recent driving reactions, and average route speed to enable long-term estimation [24]. Ravey et al. achieved speed working condition prediction in various driving scenarios through real-time recognition of transition probability matrices [25]. Yang et al. combined much actual driving cycle data and established stochastic driving behaviours as a probability transition matrix for vehicle torque demands [26]. Guo et al. designed a vision-based variable field of view speed predictor based on K-means and radial basis neural networks [27]. Considering the driving route information, Tang et al. used extreme learning machines to establish a speed working condition predictor [28]. With the rapid development of information mining technology and intelligent transportation system information acquisition technology, the acquisition of multi-source traffic information has become possible. Machine learning algorithms represented by deep learning provide a foundation for the deep mining of complex multi-source data. Mi C C et al. predicted speed and battery power based on the information interaction of connected vehicles [29]. Guanetti et al. considered the uncertainty of traffic signal timing at road intersections and used data-driven methods based on empirical sample data to formulate speed planning [30]. Egardt et al. considered the terrain, speed, powertrain efficiency, traffic lights, intersections, acceleration and braking effects in predicting speed [31]. Based on historical working condition information, Lv et al. used deep learning algorithms to predict traffic flow characteristics [32]. By integrating V2V and V2I information, He H et al. used particle swarm methods to optimize the initial values of extreme learning machines to improve speed prediction accuracy [33]. Zhang et al. proposed a speed profile prediction method based on a specific physical network system architecture and verified the accuracy of speed distribution prediction through hybrid electric vehicles [34]. The addition of multi-source information has improved the accuracy of short-term working condition prediction, and the current speed prediction accuracy has exceeded 90 %, providing a good starting point for energy management strategy development. Although deep learning-based prediction methods have significantly improved the accuracy of speed working condition prediction, in data-driven working condition prediction methods, the connection between prediction and energy allocation strategies is relatively independent, and the relationship between the two has been weakened. A summary of prediction model in predictive energy management is shown in Table 1.

To achieve better predictive energy management, the energy-saving effects of different plans are usually tested within different prediction horizons. This helps determine a fixed prediction time domain for energy management. Liu et al. compared the energy-saving effects of varying prediction methods with horizons of 1–10 s and found that the 10 s Long Short Term Memory (LSTM) prediction model was favoured for energy management strategies [21]. Liu et al. optimized BP network parameters using the genetic algorithm method, determined the optimum prediction horizon to be 10 s and verified the energy management strategy's effectiveness under the horizon [35]. Liu et al. proposed a

Table 1
A summary of prediction model in predictive energy management.

Model approach	Advantages	Disadvantages
Exponentially Varying [20,22]	Easy to implement.	The prediction of velocity does not conform to the vehicle's kinematic characteristics.
Auto-Regressive Moving Average [21]	Easy to implement. A good basis.	Significant changes in the external environment often lead to a sizable deviation from predicted outcomes.
Nonlinear Auto-Regressive with external input [22,23]	Easy to implement; Not require certainty models.	The selection of constants has a significant impact on the degree of data smoothing and should not be too small; Only suitable for short-term prediction.
Markov Chain model [24,25,36]	Reflect the transition characteristics of the state well.	Existing cumulative errors are unsuitable for the system's medium to long-term prediction.
Neural Network [27,32,34]	Better ability in learning and predicting; Better generalization ability.	Slow convergence speed; Relying on a large amount of training data.
Deep Neural Network [36,37]	Strong ability in learning, and predicting; More accurate prediction; Strong generalization ability.	Slow convergence speed; Relying on a large amount of training data.

data-driven online velocity prediction method for short-term speed work and determined the optimum prediction time to be 5 s for a speed of 1.27 km/s [36]. Jinquan et al. compared short-term working condition prediction effects under a deep neural network model for 5 s, 10 s, and 15 s intervals and determined that the best prediction time domain under an economic-driving professional system was 10 s [37]. It is evident that as the prediction horizon extends, the accuracy of future speed conditions gradually decreases. Thus, while energy management strategies in the model predictive control framework exhibit robustness when prediction accuracy falls below a certain threshold, the energy-saving effect of such strategies will be considerably diminished even if the predicted working condition trend is correct.

The energy management strategy with a fixed prediction horizon cannot dynamically adjust based on information such as vehicle status and operating conditions. Therefore, it is unable to fully maximize optimization potential. However, drivers can typically predict future driving conditions within a limited horizon while driving, with limitations such as road visibility and vehicle speed. Suppose the EMS can dynamically select the current prediction horizon and automatically adjust it based on the vehicle's status and surrounding environment. In that case, the energy-saving potential of the vehicle can be maximized. Determining the optimal prediction time domain becomes challenging under the complex vehicle and traffic conditions, as it depends on various factors like the prediction method, vehicle status, road surface, and traffic conditions. Balancing the length of future power trends and prediction accuracy through any algorithm is challenging when relying on the accuracy of future operating condition predictions for energy management strategies.

Hence, this paper proposes a data-driven bi-level predictive EMS (DB-PEMS), which combines traditional predictive energy management and data-driven machine learning for FCBs with multi-prediction horizons. The data-driven strategy is employed in the upper layer to select the optimal prediction horizon based on vehicle states. Two efficient techniques for multi-prediction horizons are proposed. The first approach is a modular method grounded on the optimal Δ state of charge (SoC) approximation. This approach establishes a mapping between vehicle states and optimal ΔSoC via a deep neural network (DNN), enabling horizon selection. The second approach is an end-to-end method based on Deep Reinforcement Learning (DRL). This approach leverages an offline trained DRL agent to optimize prediction horizons and achieve multi-horizon speed predictions. The former harnesses the data-mining skills of DNNs, while the latter taps into the environment exploration capability of DRL. In the lower layer, an MPC-based Energy Management System receives prediction horizons from the upper layer and then conducts speed prediction, rolling optimization, and feedback. The research framework is illustrated in Fig. 2.

Contributions

(1) Data-driven bi-level PEMS, data-driven approaches integrating into the traditional prediction management strategy to optimize the prediction horizon, is proposed for FCBs.

(2) By exploring the mapping relationship between the vehicle status and the optimal ΔSoC, an optimization model based on deep learning for ΔSoC approximation is established, which guides the reference trajectory of SoC in MPC and enhances the energy-saving potential of PEMS.

(3) DRL-based mechanism has been established to select the optimal prediction horizon by enabling interactive learning and exploration between DRL agents and PEMS environments, thereby improving the energy-saving of FCBs.

Organization

The organization of this paper is as follows. The vehicle model of FCB is presented in Section II, and Section III introduces the methodology of DB-PEMS for FCBs. The results and discussions are described in Section IV, and Section V draws the conclusions.

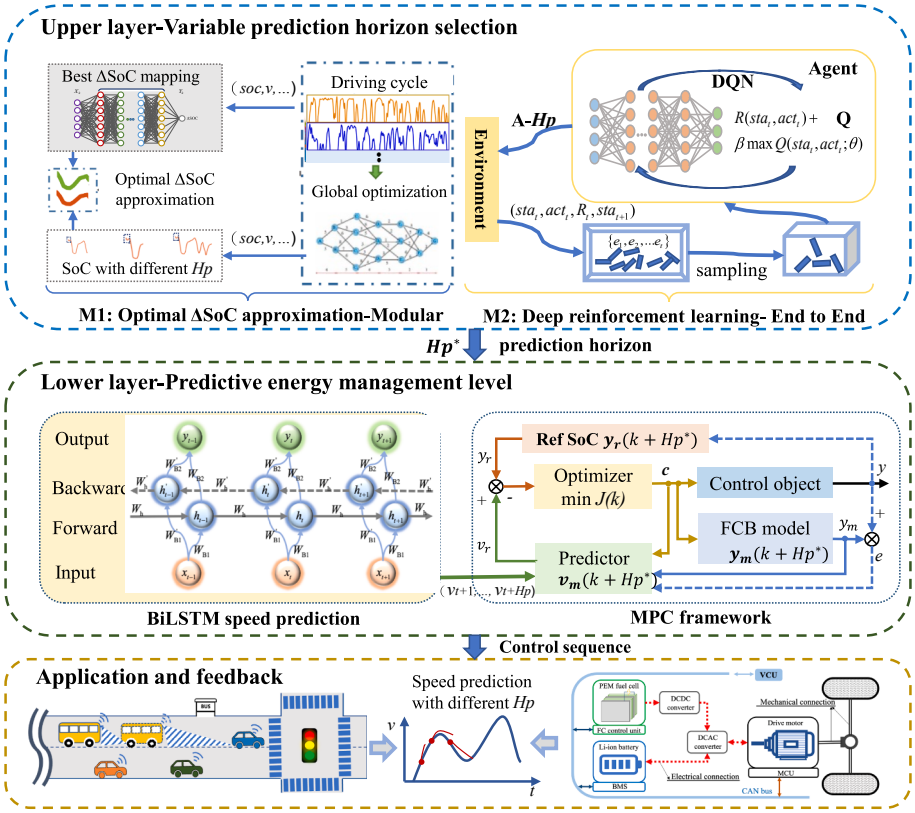


Fig. 2. DB-PEMS with multi-prediction horizons for FCBs.

Vehicle model

The FCB produced by Yutong Bus Company is studied as the research object. The power system mainly includes a fuel cell system (FCS), DC/DC converter, lithium iron phosphate battery pack, battery management system, drive motor and drive motor controller, main gear, etc. The actual FCB and the powertrain structure are shown in Fig. 3.

Longitudinal dynamic modelling

The vehicle must overcome the rolling, air, ramp, and acceleration resistance when driving. That is, the vehicle's driving force should be balanced with the resistance. Therefore, the vehicle driving equation can be expressed as Eq. (1) [28].

$$F_t = F_f + F_w + F_i + F_j$$

$$F_t = Gf\cos\alpha + \frac{1}{2}C_D A\rho v^2 + G\sin\alpha + \delta m\frac{dv}{dt} \tag{1}$$

The resistance power experienced by the vehicle while driving is

$$P = -F_t \cdot v \tag{2}$$

When the vehicle is running, the driving force and the total resistance should be balanced. Also, the motor output power and the total resistance power should be balanced. The full resistance power includes the power generated by the rolling, air, ramp, and acceleration resistance and the mechanical wear power generated during the power transmission process. Mechanical transmission power loss is expressed in terms of η_T. The vehicle's driving power balance equation is described as Eq. (3).

$$\begin{aligned} P_{req} &= -P = P_m\eta_T \\ P_{req} &= \frac{v}{3600}\left(Gf + \frac{C_D A\rho v^2}{2} + Gi + \delta m\frac{dv}{dt}\right) \end{aligned} \tag{3}$$

The onboard energy sources of FCB include a fuel cell system and a power battery. The power balance relationship of FCB is shown in Eq. (4).

$$P_{req}(t) = \left(P_{bat}(t)\cdot\eta_{bat} + P_{fc}(t)\cdot\eta_{fc}\cdot\eta_{DC}\right)\cdot\eta_m \tag{4}$$

The energy consumption is expressed as Eq. (5) [38].

$$\begin{aligned} E_{PS} &= E_{fc} + E_{bat} \\ E_{fc} &= \frac{1}{3.6\times10^6}\left(\int\left(P_{FC}(t)\cdot\eta_{FC}\cdot\eta_{DC}\cdot\eta_m\right)dt \right. \\ E_{bat} &= \left(SOC_{int} - SOC_{final}\right)C_{bat} \end{aligned} \tag{5}$$

Although the impact of driving data on energy consumption is not directly reflected in energy consumption calculations, it is nonetheless a crucial factor in determining the energy demand of the vehicle power system. The energy required for vehicle driving E_{PS} is calculated by the longitudinal dynamic equation of the vehicle, that is, by integrating Eq. (3) to obtain the energy consumption demand. Specifically, the speed and acceleration of vehicles are important components of driving data and key parameters for calculating power demand. Therefore, when optimizing energy management, the consideration of driving data and variable prediction horizon is aimed at improving the accuracy of speed prediction and more accurately calculating future demand power. In this way can more effective and accurate optimization strategies be developed, power output be allocated reasonably, and energy efficiency be improved.

Power system modelling

The power system of FCB mainly includes fuel cell system, battery pack and its management system, DC/DC converter, drive motor, and DC/AC inverter. The fuel cell system, battery, DC/DC converter, and drive motor are modeled based on the test data in the laboratory [39].

Fuel cell system

The fuel cell system used consists of proton exchange membrane fuel cell (PEMFC) stack and its accessory. A circuit model is used to model the PEMFC, as shown in Fig. 4. It includes an ideal voltage source and three series resistors, which represent the irreversible losses caused by activation loss, ohmic loss, and concentration loss during the operation of the PEMFC.

The individual output voltage of the PEMFC stack can be expressed as Eq. (6), and the total voltage of the PEMFC stack is the product of individual voltage and quantity, expressed as Eq. (7) [40].

$$V_{cell} = E_0 - I_{fc}R_{act} - I_{fc}R_{ohmic} - I_{fc}R_{con} = E_0 - V_{act} - V_{ohmic} - V_{con} \tag{6}$$

$$V_{stack} = N_{cell}\cdot V_{cell} \tag{7}$$

In standard circuits, the resistance value is fixed, but in the equivalent circuit of PEMFC stack, the resistance value is related to the current, so the equivalent circuit model is nonlinear.

The fuel cell system (FCS) efficiency η_{fc} is shown as Eq. (8).

$$\eta_{fc} = \frac{P_{fc\_load}}{N_{cell}V_{ec}I_{fc}} = \frac{P_{fc\_out} - P_{fc\_aux}}{N_{cell}V_{ec}I_{fc}} = \frac{V_{stack}I_{fc} - P_{fc\_aux}}{N_{cell}V_{ec}I_{fc}} \tag{8}$$

The corresponding relationship between the efficiency and output power of the FCS, the hydrogen consumption and output power are shown in Fig. 5 based on laboratory test data [39].

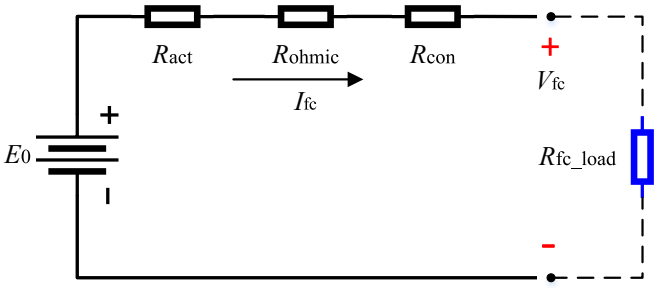
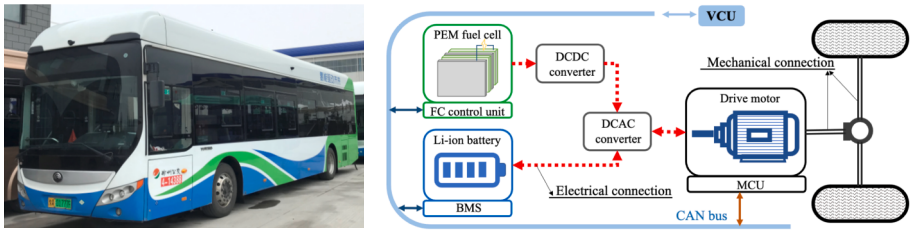


Fig. 4. Equivalent circuit of PEMFC stack.



(a) The fuel cell bus (b) The powertrain structure

Fig. 3. FCB and the structure of the powertrain.

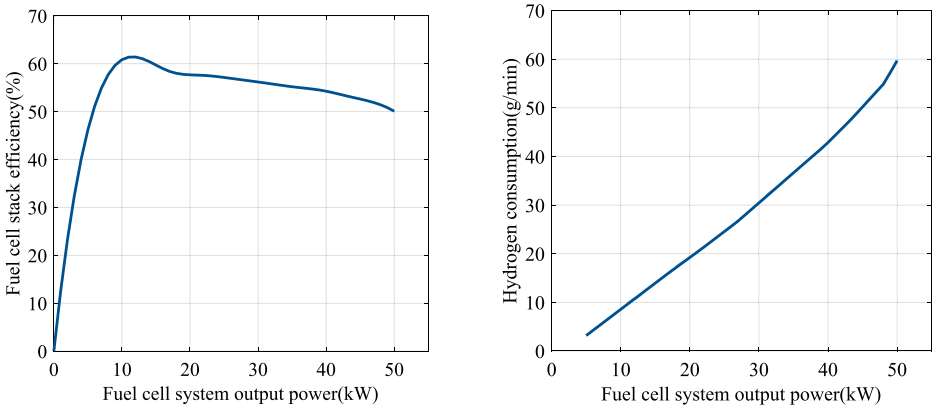


Fig. 5. Relationship between the efficiency and output power, the hydrogen consumption and output power.

Battery

The equivalent circuit of Thevenin model is used to model the battery, as shown in Fig. 6. According to Kirchhoff's law of voltage and current, the relationship between the equivalent circuit of the Thevenin battery model can be expressed as Eq. (9) [41].

$$\begin{aligned} V_{\mathrm{bat}} &= E_{\mathrm{bat}} - V_{\mathrm{pol}} - I_{bat}R_{bat} \\ V_{\mathrm{pol}} &= \frac{1}{C_{\mathrm{pol}}}I_{bat} - \frac{1}{C_{\mathrm{pol}}R_{\mathrm{pol}}}V_{\mathrm{pol}} \\ I_{bat} &= \frac{V_{\mathrm{bat}} - V_{\mathrm{pol}} - \sqrt{\left(V_{\mathrm{bat}} - V_{\mathrm{pol}}\right)^2 - 4R_{\mathrm{bat}}P_{\mathrm{bat\_load}}}}{2R_{\mathrm{bat}}} \\ SOC &= SOC_{int} - \frac{\int I_{\mathrm{bat}}\mathrm{d}t}{C_{\mathrm{bat}}} \end{aligned} \tag{9}$$

The charge and discharge internal resistance and voltage characteristics of the battery are shown in Fig. 7.

DC/DC converter

The DC/DC converter can boost the output voltage of the FCS and maintain consistency with the bus voltage, while also having the function of voltage regulation. As the FCS only outputs electricity externally and cannot recover electricity, a unidirectional DCDC converter is chosen to regulate the voltage of the FCS. To reduce computational costs and maintain model accuracy, the efficiency of the DC/DC converter is obtained by looking up tables at the corresponding current and output power. The efficiency relationship of the DC/DC converter is shown in Fig. 8 [39].

Drive motor

The driving motor is modeled using a quasi-steady state method and the expression is shown in Eq. (10). The efficiency of the drive motor is shown in Fig. 9.

$$P_{\mathrm{m}} = \begin{cases} \dfrac{T_{\mathrm{m}}n_{\mathrm{m}}}{\eta_{\mathrm{m}}(T_{\mathrm{m}}, n_{\mathrm{m}})} T_{\mathrm{m}} \geqslant 0 \\ T_{\mathrm{m}}n_{\mathrm{m}}\eta_{\mathrm{m}}(T_{\mathrm{m}}, n_{\mathrm{m}}) T_{\mathrm{m}} < 0 \end{cases} \tag{10}$$

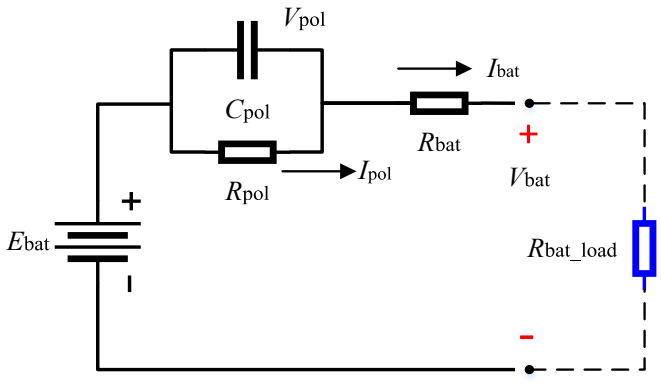


Fig. 6. Equivalent circuit of the Thevenin battery model.

Predictive energy management strategy with multi-horizons

For PEMS, the higher the accuracy of the predicted speed, the more conducive it is to improving the optimization effect of EMS. However, the accuracy of velocity prediction is not only related to the complexity of the driving cycles themselves, but also influenced by the length of the prediction horizon. Therefore, the PEMS with multi-horizons is proposed to improve the accuracy of velocity prediction and thus enhance the overall energy-saving effect. We propose two data-driven methods with algorithmics fusion to achieve variable multi-horizons for vehicle speed prediction, one based on optimal ΔSoC approximation and the other based on DRL. The former summarises the prediction horizon optimization strategy from offline data. The latter adopts a model-free manner to learn the optimal prediction horizon from the interactive learning of DRL.

Multi-horizons prediction method based on the optimal ΔSoC approximation

In the PEMS, the state variable SoC in the rolling optimization is usually constrained by the final SoC value or the local optimal SoC trajectory in the prediction horizon length. Neither of these two state variable constraints can make the rolling optimization result sufficiently close to the optimal global solution. On the premise that the global driving cycles are known, DP is used to obtain the trend of the optimal SoC trajectory under different speeds, which can better reflect the change of the optimal SoC under different conditions, that is, the optimal ΔSoC. Furthermore, suppose the state variable SoC in the rolling optimization process can follow the optimal ΔSoC at each moment. In that case, the rolling optimization results will also tend to be closer to the optimal global solution. Due to the fact that the actual driving process is real-time and online, it is impossible to know the global operating conditions in advance. Therefore, we propose to use deep learning methods to learn the optimal SOC trajectory corresponding to different speed operating conditions. So, a multi-prediction horizons EMS based on the optimal ΔSoC approximation is proposed. The optimal ΔSoC approximation is achieved by using DNN fitting.

Since the SoC trajectory solved in different prediction horizons is different from the optimal SoC trajectory. It means that the ΔSoC in different prediction horizons is different from the optimal ΔSoC. The prediction horizon corresponding to the ΔSoC closest to the optimal ΔSoC is selected as the best prediction horizon. The research framework

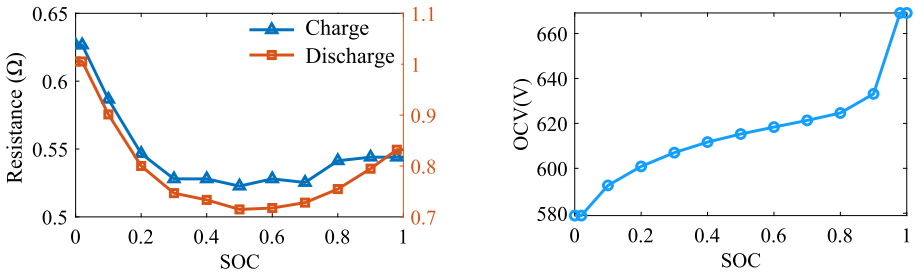


Fig. 7. Internal resistance and voltage characteristics of the battery.

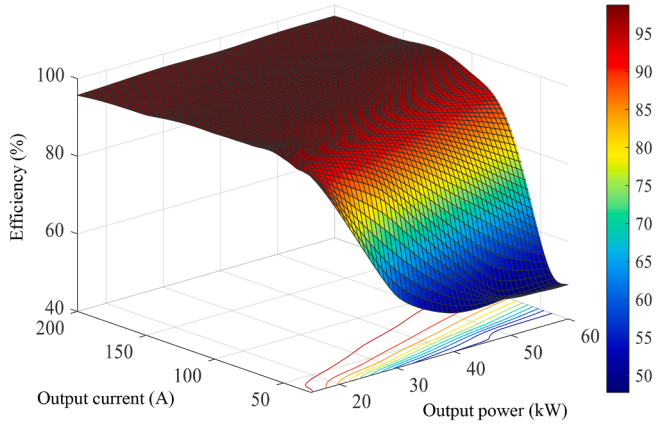


Fig. 8. Efficiency of the DC/DC converter.

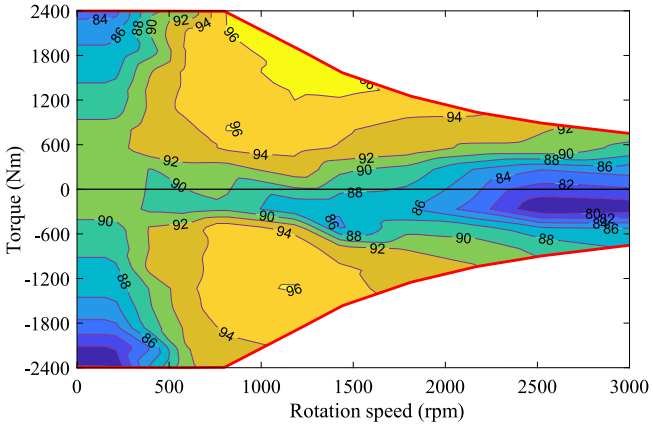


Fig. 9. Efficiency of the drive motor.

of the multi-horizons prediction method based on optimal ΔSoC approximation is shown in Fig. 10, which mainly includes three parts. The first part is to establish a relationship database between the vehicle states and the best ΔSoC. Secondly, a DNN-based predictor is built to predict the best ΔSoC. The third part is to determine the optimal prediction horizon for each moment.

Database establishment for prediction network

The database construction scheme is shown in Fig. 11. Firstly, the actual bus driving cycles collected on Zhengzhou 727 bus lines are taken as known, and DP is adopted to solve the optimal SoC trajectory under each driving cycle. The algorithm for solving the offline global optimal strategy using DP to obtain the optimal SOC is shown in Eq. (11) [21]. The state variable is the battery SoC and the output power of FCS. The control variable is the change in the output power of FCS. The optimization goal is to minimize energy consumption while constraining the change rate of the output power of the FCS. As the FCS is a slowing system, the FC power change rate constraints are added to the objective optimization function to reduce its life degradation. When the FCB executes the optimal control sequence of the DP solution, the SoC of the battery is the offline optimal SoC trajectory.

$$\begin{aligned} x &= [SOC,\ P_{\mathrm{fc}}] \\ u &= [\Delta P_{\mathrm{fc}}] \\ J &= \min(\sum \chi(\frac{dE_{\mathrm{fc}}}{dt} + \varphi\frac{dE_{\mathrm{bat}}}{dt}) + \zeta\Delta P_{\mathrm{fc}}) \end{aligned} \tag{11}$$

After obtaining the optimal SoC, the rate of change of SoC at each moment can be calculated, denoted as ΔSoC, expressed as Eq. (12). Correspondingly, the vehicle's status at each moment, including distance travelled, current SoC, speed, acceleration, and power, can be extracted, shown in Eq. (13). Hence, the optimal ΔSoC at the current moment can be matched with the vehicle's status. Thus, a database for optimal ΔSoC prediction is established, as shown in Eq. (14).

$$Y(t) = \begin{cases} \Delta SoC_k^* = SoC_{k+1}^* - SoC_k^* t = k \\ \Delta SoC_{k+1}^* = SoC_{k+2}^* - SoC_{k+1}^* t = k+1 \\ \cdots \\ \Delta SoC_{k+n}^* = SoC_{k+n+1}^* - SoC_{k+n}^* t = k+n \end{cases} \tag{12}$$

$$X(t) = \begin{cases} s_k, SoC_k, v_k, a_k, P_k t = k \\ s_{k+1}, SoC_{k+1}, v_{k+1}, a_{k+1}, P_{k+1} t = k+1 \\ \cdots \\ s_{k+n}, SoC_{k+n}, v_{k+n}, a_{k+n}, P_{k+n} t = k+n \end{cases} \tag{13}$$

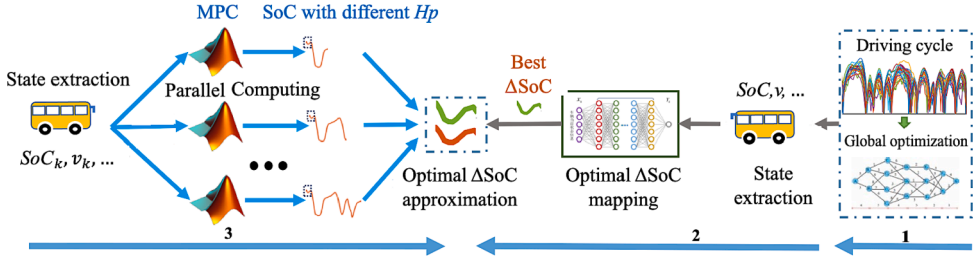


Fig. 10. Multi-horizons prediction method based on optimal ΔSoC approximation.

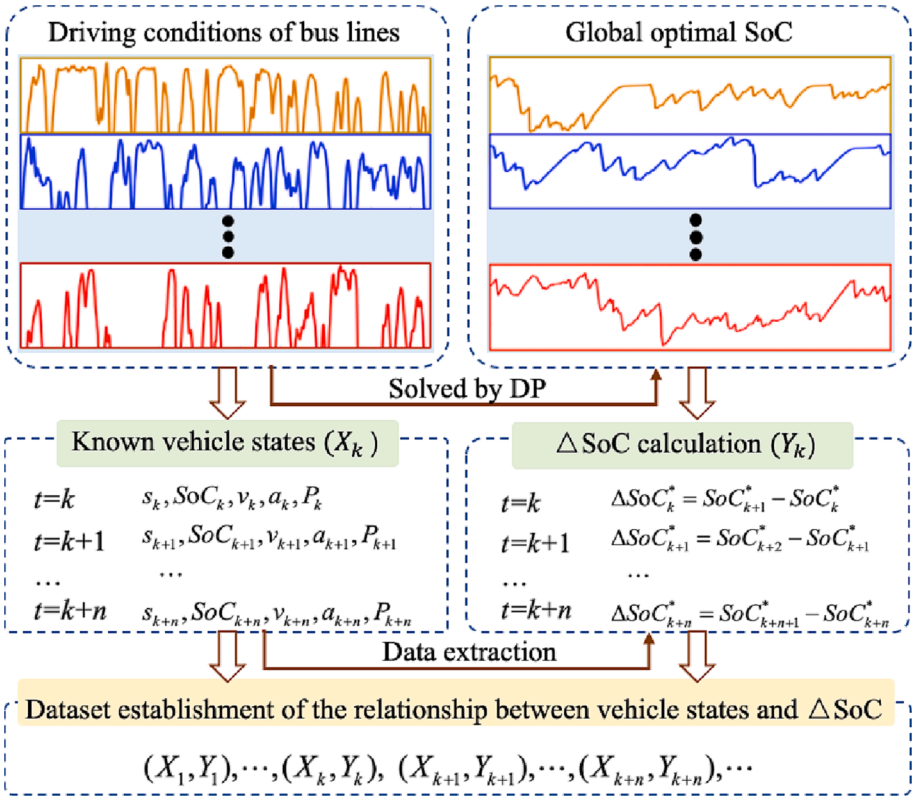


Fig. 11. Database construction scheme.

$$(X_1, Y_1), \cdots, (X_k, Y_k), (X_{k+1}, Y_{k+1}), \cdots, (X_{k+n}, Y_{k+n}) \tag{14}$$

DNN-based optimal ΔSoC predictor establishment

To find the mapping relationship between the current vehicle driving states and the corresponding optimal ΔSoC, an optimal ΔSoC prediction method based on DNN is proposed. First, a prediction network DNN_SoC for optimal ΔSoC prediction is built, shown in Fig. 12. Then, the hidden layers, nodes, and activation function of DNN_SoC are adjusted according to the training results. The expression of the prediction network DNN_SoC is shown in Eq. (15) [42]. The input of DNN_SoC is the current vehicle driving states, including the driving distance, SoC, speed, acceleration and demand power, shown in Eq. (16). The output of DNN_SoC is ΔSoC, that is, the difference between the SoC at the current moment and the next moment, shown in Eq. (17).

$$Y(t) = f_{\mathrm{DNN\_SoC}}(X(t)) \tag{15}$$

$$X_k(t) = s_k, SoC_k, v_k, a_k, P_k t = k \tag{16}$$

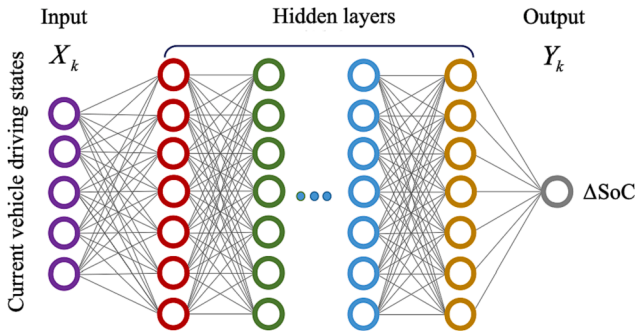


Fig. 12. Optimal ΔSoC prediction network DNN_SoC.

$$Y_k = \Delta SoC_k^* \tag{17}$$

The data set is randomly divided, 75 % of the data is used as the training set, and the remaining 25 % is used as the test set. The established ΔSoC prediction network DNN_SoC is trained until meeting the accuracy requirements, and then the trained DNN_SoC network is verified with the test set.

The optimal ΔSoC is predicted via the DNN_SoC prediction network. Then it is compared to the ΔSoC solved in different prediction horizons to determine the selection of the optimal prediction horizon. The flowchart is shown in Fig. 13.

Multi-prediction horizons optimization based on deep reinforcement learning

The multi-prediction horizons optimization based on DRL mainly includes two parts. One part is the agent, which learns the states passed by the environment and gives the optimal control action through the deep Q-network (DQN). The second part is the environment. For the research question in this paper, the environment is composed of MPC-based PEMS and application object FCB. After the environment receives the action command sent from the agent, the PEMS in the environment sequentially performs speed prediction and rolling optimization. Finally, the research object FCB executes the control command and transfers to a new state. The current states, action, reward and new states are formed into an information group, and the transition process of FCB state-action-reward-state is recorded and stored in the replay buffer.

Before the online application, the DRL algorithm needs to be trained offline. Taking the bus driving cycles collected on the Zhengzhou 727 bus line as the known conditions, the optimal action-value function is found through continuous exploration and trial under the DRL framework shown in Fig. 14. To ensure the independence of action selection, the replay buffer is adopted. Then a set of data is randomly selected from

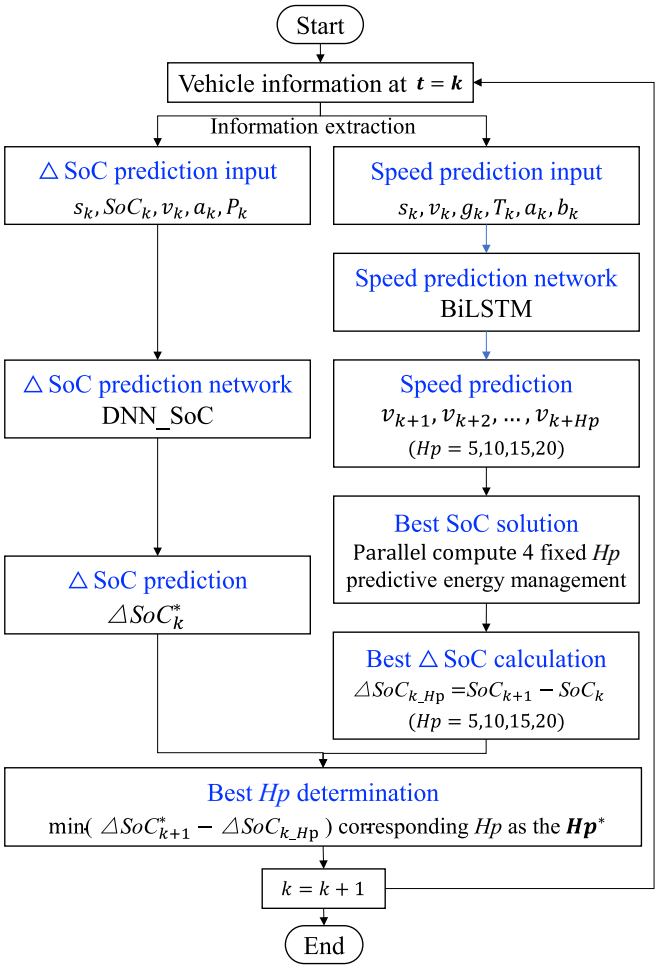


Fig. 13. Flowchart for the selection of the optimal prediction horizon.

the experience pool and passed to the agent to fit the DQN. At the same time, to ensure the convergence and efficiency of training, the state variables are normalized.

Determination of deep reinforcement learning elements

The objective function of the DQN is expressed in Eq. (18) [43]. The gradient descent function is used to update the weight parameters of the network in training DQN, and the greedy algorithm is used to optimize the training of the DRL algorithm. Finally, the offline training of the DQN is realized, so the optimal action control can be output to achieve a better control effect in practical applications.

$$Q(t) = R(sta_t, act_t) + \beta \max Q(sta_t, act_t; \theta) \tag{18}$$

For the FCB EMS, the optimization goal is to reduce the energy consumption and constrain the change rate of the output power of the FCS, which is the same as that of DP for solving the optimal SoC. So, the optimization objective function, also the reward function in the DRL algorithm, is expressed as Eq. (19).

$$R = -\left(\sum \chi\left(\frac{dE_{fc}}{dt} + \varphi \frac{dE_{bat}}{dt}\right) + \zeta \Delta P_{fc}\right) \tag{19}$$

The DRL algorithm does not suffer from the curse of dimensionality caused by the increase of the states. Considering the characteristics influence of FCB energy sources and the external environment on the optimization objective, the battery SoC, the vehicle position, the current speed and the predicted speed are input as state variables, as shown in Eq. (20). The control action is to choose the optimal prediction horizon. To improve the computational efficiency, the selection range of the prediction horizon is consistent with Section 3.1, which are 5 s, 10 s, 15 s and 20 s, as shown in Eq. (21).

$$sta = \{SoC, s, v, a, P\} \in S \tag{20}$$

$$act = \{hp_i\} = \{hp_1, hp_2, hp_3, hp_4\} \in Act \tag{21}$$

Training of deep reinforcement learning agent

Table 2 shows the procedures for offline training of DRL-based multi-prediction horizons optimization. The greedy coefficient is ε, the number of cycles is N, the length of the driving cycle is T, the sampling length is n, and the DNN is used in the decision network to fit the action-value function Q_{value}.

Predictive energy management strategy with variable multi-prediction horizons

Both data-driven methods mentioned above can obtain the optimal prediction horizon under different driving conditions. The DB-PEMS for FCB within the MPC framework can be performed, and the signal transmission of MPC is shown in Fig. 15.

The overall algorithm flow of variable multi-prediction horizons PEMS is as follows:

1) At the moment $t = k$, obtain the optimal prediction horizon Hp^* based on the best ΔSoC approximation or DRL agent.

2) Predict future speed $\{v_{k+1}, v_{k+2}, \cdots v_{k+Hp^*}\}$ based on BiLSTM predictor in optimal prediction horizon Hp^*.

3) Use the DP algorithm to solve the optimal control sequence $\{c_{k+1}, c_{k+2}, \cdots c_{k+Hp^*}\}$ in the Hp^* horizon, and apply the first step control sequence $\{c_{k+1}\}$ to the control object.

4) After the control object executes the control command, the vehicle state is updated, and the new state is fed back to the controller.

5) When the vehicle travels to the next moment $k = k + 1$, repeat 1)

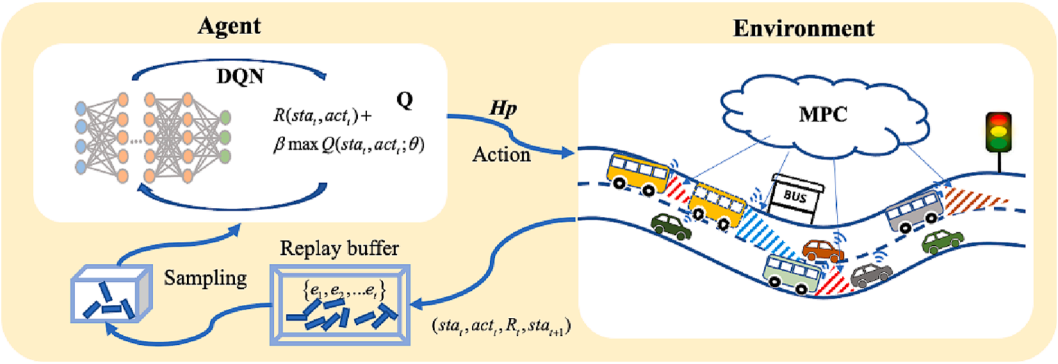


Fig. 14. DRL framework for multi-prediction horizon.

Table 2
Procedures for offline training of DRL-based multi-prediction horizons optimization.

DRL-based multi-prediction horizons optimization:	
	Input: state space S, action space Act, discount rate γ, learning rate ς.
1	Initialise replay buffer D, size L.
2	Initialize the parameter θ of the action-value network Q.
3	Initialize the parameter θ^* of the target action-value network Q^*.
4	**repeat**
5	Initialize the start state s.
6	**repeat**
7	Select action $act_t = \pi^{\varepsilon}(Hp)$, or $act_t = \arg\max_{act \in A} Q_{value}(sta_t, act_t; \theta)$ according to the target value network.
8	Predict speed in selected action Hp, calculate the reward r_t and the next moment state sta_{t+1} within MPC.
9	Store $(sta_t, act_t, r_t, sta_{t+1})$ in D.
10	Mini batch.
11	Order $Q_i = R_i + \beta \max Q_{value}(sta_t, act_t; \theta))$, if $i = n, y_i = r_i$.
12	Update $[03B8](W, b)$ by gradient descent $(Q_i - Q_{value}(sta_i, act_i; \theta))^2$.
13	**until** s terminated.
14	**until** $\forall s, a, Q(s, a)$ converged.
	Output: $Q_{value}(s, a)$.

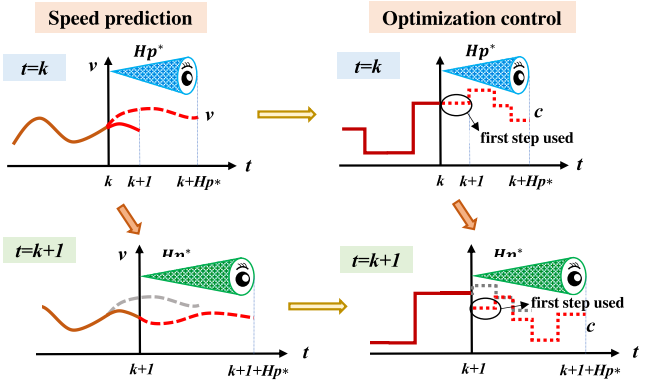


Fig. 15. Signal transmission of MPC.

to 4) until the end of the trip, to realize the multi-prediction horizons PEMS.

Speed prediction in the variable prediction horizon is carried out by BiLSTM-based speed predictor. The inputs and outputs of the BiLSTM-based speed predictor are shown in Eq. (22).

$$\begin{aligned} Output(t) &= f_{\text{BiLSTM\_NN}}(Input(t)) \\ Input(t) &= \begin{bmatrix} [v_{t-hh+1}, v_{t-hh+2}, \ldots, v_{t-1}, v_t] \\ [a_{t-hh+1}, a_{t-hh+2}, \ldots, a_{t-1}, a_t] \\ [s_{t-hh+1}, s_{t-hh+2}, \ldots, s_{t-1}, s_t] \\ [p_a] \\ [p_d] \end{bmatrix} \\ Output(t) &= [v_{t+1}, v_{t+2}, \ldots, v_{t+hp}] \end{aligned} \tag{22}$$

The optimization objective in the DB-PEMS for FCB with variable multi-prediction horizons is set to minimize equivalent hydrogen consumption and constraints on the output power change rate of the FCS. This is consistent with the optimization objective of solving the optimal SOC trajectory offline. The state variable is set to the battery SOC and the output power of the FCS, the control variable is set to the change of the output power of the FCS, and the disturbance variable is set to the predicted velocity. At the same time, the FCS, battery and drive motors need to meet corresponding physical constraints. Therefore, the control problem of DB-PEMS can be expressed as Eq. (23).

$$\begin{aligned} J &= \min\left(\sum \chi\left(\frac{dE_{\text{fc}}}{dt} + \varphi \frac{dE_{elec}}{dt}\right) + \delta \Delta P_{\text{fc}}\right) \\ x &= [SOC,\ P_{\text{fc}}] \\ u &= [\Delta P_{\text{fc}}] \\ w &= [v] \\ s.t. &\begin{cases} Fuel \begin{cases} P_{fc}(k+1) = P_{fc}(k) + \Delta P_{fc}(k) \\ 0 \leqslant P_{fc.\min} \leqslant P_{fc} \leqslant P_{fc.\max} \leqslant P_{fc.m} \end{cases} \\ Battery \begin{cases} SOC_{\min} \leqslant SOC \leqslant SOC_{\max} \\ P_{bat.\min} \leqslant P_{bat} \leqslant P_{bat.\max} \\ I_{bat.\min} \leqslant I_{bat} \leqslant I_{bat.\max} \end{cases} \\ Motor \begin{cases} P_{m.\min} \leqslant P_m \leqslant P_{m.\max} \\ T_{m.\min} \leqslant T_m \leqslant T_{m.\max} \end{cases} \end{cases} \end{aligned} \tag{23}$$

Results and discussions

Data collection of online fuel cell buses

A data collection device is installed on the FCB to collect real-world operation data on the No. 727 bus line in Zhengzhou, China. The main parameters of YUTONG FCB are shown in Table 3. The installed data acquisition equipment and the bus route are shown in Fig. 16. The NPOS220 receiver of BDStar Navigation obtains vehicle posture information and position information, and Kvaser USBcan is used to obtain vehicle bus status information. During the data collection process on the actual FCB, the designed controller was not utilized for data acquisition. Because the original controller is deemed to better reflect the data status. Furthermore, the proposed DB-PEMS belongs to end-to-end algorithm. Due to hardware limitations on the test vehicle and road, there are potential safety risks if the designed controller is directly applied in real-world environments.

The speed information database contains approximately 1.4 million sets of real-world collected data, randomly divided into training and testing datasets in a ratio of 75 % and 25 %. And the two datasets do not intersect. The test data is randomly extracted from the test dataset to ensure the effectiveness and generalization of the prediction network

Table 3
Main parameters of vehicle and power system for YUTONG FEB [38].

Name	Parameters	Value
Vehicle	Length × Width × Height	12000 × 2550 × 3400 mm
	Curb weight	13500 kg
	Full load quality	18000 kg
	Maximum windward area	8.16 m^2
	Air drag coefficient	0.55
	Rolling resistance coefficient	0.0085
	Wheel radius	0.466 m
	Main gear ratio	6.2
	Maximum speed	69 km/h
	Maximum acceleration	0 ~ 50 km/h-20 s
FC system	Rated power	60 kW
	Rated current	0 ~ 350 A
	Output voltage	145 ~ 290 V
DC/DC converter	Rated power	60 kW
	Output voltage	400–720 V
	Efficiency	90 ~ 95%
Power battery	Capacity	108.14 kWh
	voltage range	540 ~ 738 V
Drive motor	Rated power	100 kW
	Rated torque	1200 Nm
	Peak power	200 kW
	Peak torque	2400 Nm
	Rated/Maximum Speed	800/3000 rpm
	Efficiency	Max 97%, 85%

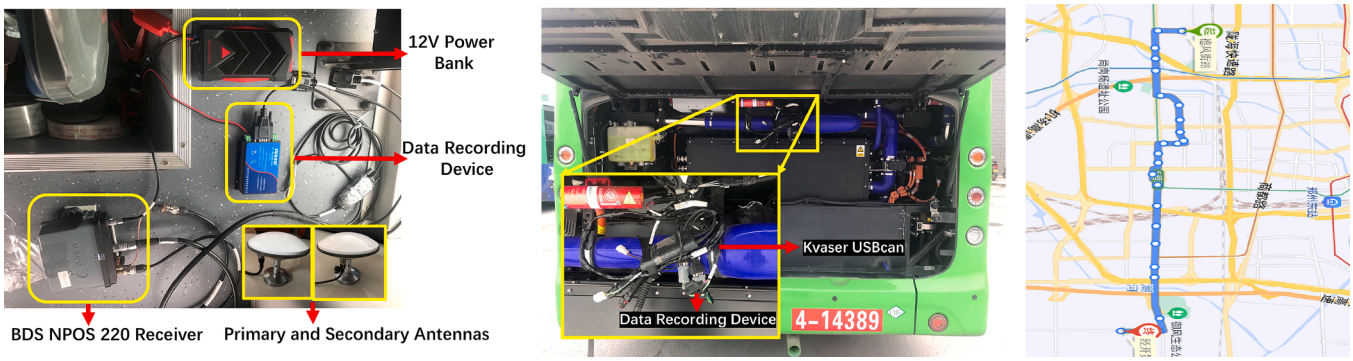


Fig. 16. Data acquisition equipment installed in FCB and bus line map.

trained. Fig. 17 shows part of the test set's speed, acceleration, and demand power.

Analysis and discussion for the selection of multi-prediction horizon

Multi-prediction horizons based on optimal ΔSoC approximation

To assess the efficacy of selecting the optimal prediction horizon with the optimal ΔSoC approximation approach, the precision of ΔSoC prediction and the proximity between the SoC based on ΔSoC approximation and the optimal SoC solved by DP are analyzed and discussed. Additionally, the distribution and proportion of the multi-prediction horizons are analyzed to demonstrate the corresponding variations in the prediction horizon under diverse driving conditions.

Fig. 18 shows the actual and predicted ΔSoC, which exhibits a high degree of similarity in the overall change trend and numerical value. The maximum error of both the actual and predicted ΔSoC is less than 0.04 %, and the minimum error is greater than −0.06 %. The simulation results indicate that the prediction network DNN_SoC can learn the relationship between the current states and the future energy changes of the battery, that is, the predicted ΔSoC can well represent the actual ΔSoC. The accurate prediction of ΔSoC is beneficial for selecting the optimal prediction horizons and effectively guiding the battery's SoC trajectory.

The prediction error between the actual ΔSoC and the predicted ΔSoC is illustrated in Fig. 19. The discrepancy between the two values fluctuates between −0.014 % and +0.014 %. The small mean absolute error (MAE) of 1.687e-05 indicates that the ΔSoC prediction network can accurately forecast the future ΔSoC. This also suggests that utilizing the ΔSoC approximation method for optimal horizon selection is a feasible and practical approach.

Fig. 20 presents a comparison between the SoC obtained via DP (SoC_DP) and the predicted SoC obtained by accumulating the predicted ΔSoC based on the initial value of SoC = 0.8 (SoC_ΔSoC). The graph illustrates that the changing trend of SoC_ΔSoC is similar to that of SoC_DP. Both have an initial value of 0.8, with the final value of SoC_ΔSoC being 0.8031 and SoC_DP being 0.8008. The gap between SoC_ΔSoC and SoC_DP is slightly larger in the initial stage of 0 ~ 1300 s and at the end of the trip compared to the middle driving process. Both reach the maximum SoC of 0.826 around the 2480 s. This indicates that the ΔSoC approximation method can accurately predict the changing trend of the battery's SOC, which provide an energy trajectory reference for optimization processes and serve as a reliable measure for obtaining the optimal prediction horizon.

Fig. 21 depicts the demand power of the FCB power system and the corresponding prediction horizon distribution used to predict the velocity in DB-PEMS based on the optimal ΔSoC approximation approach.

As depicted in Fig. 21, the demand power fluctuates between −200 kW and 200 kW. The prediction horizon of 5 s accounts for the largest proportion of speed prediction, followed by 10 s and 15 s, while the prediction horizon of the 20 s has the smallest proportion. The latter mainly occurs when the frequency of power changes drastically or the amplitude changes significantly, such as 450 ~ 480 s, 990 ~ 1030 s, 2700 ~ 2880 s, 3050 ~ 3150 s, 4000 ~ 4080 s, etc. The prediction horizon of 15 s mainly occurs when the power variation range is relatively moderate, such as 0 ~ 270 s, 1590 ~ 1700 s, 3680 ~ 3750 s, etc. The prediction horizon of 10 s mainly occurs when the power frequency change is relatively low and the amplitude range is slight, such as 150 ~ 480 s, 1300 ~ 1900 s, etc. The prediction horizon of 5 s mainly occurs

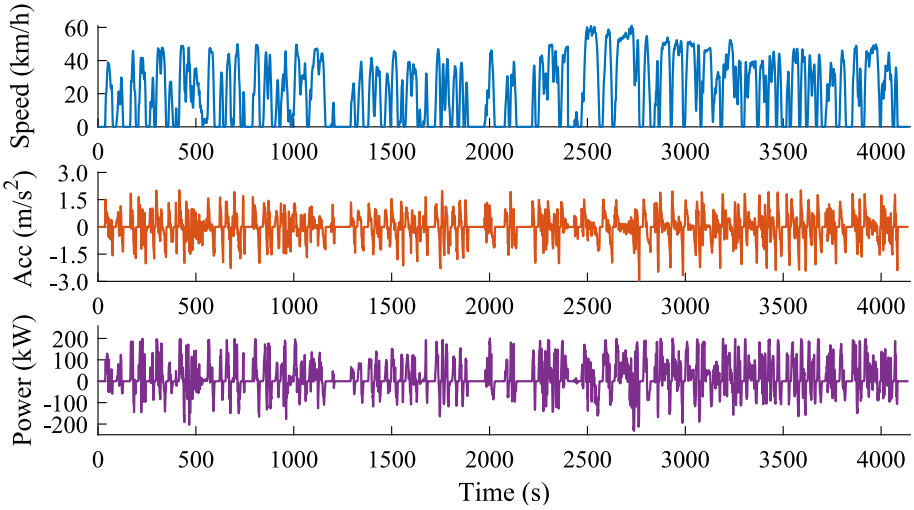


Fig. 17. Speed, acceleration and demand power in the test set.

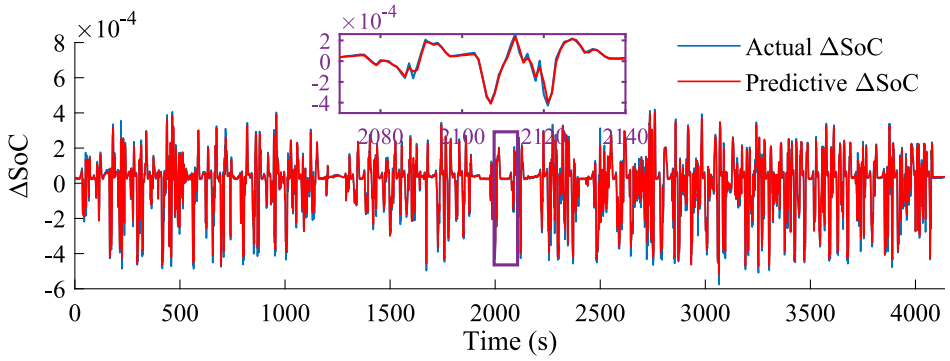


Fig. 18. Actual ΔSoC and the predicted ΔSoC.

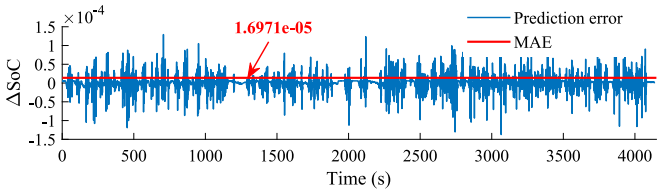


Fig. 19. Prediction error between the actual ΔSoC and the predicted ΔSoC.

during the vehicle's idling or low-speed stage. When ΔSoC is the same in different prediction horizons, the shorter prediction horizon is preferred for the current step.

Fig. 22 displays a comparison between the SoC_ΔSoC and the SoC obtained with the fixed prediction horizon (SoC_fixed horizon, 5 s, 10 s, 15 s, 20 s). As depicted in Fig. 22, the SoC_ΔSoC fluctuates among the SOCs with the four fixed prediction horizons. When the predicted horizon of the velocity prediction with variable horizons is 5 s, the SoC_ΔSoC is closer to the SOC_fixed 5 s. Similarly, when the predicted horizon of the velocity prediction with variable horizons is 10 s, the SoC_ΔSoC is closer to the SOC_fixed 10 s. From Fig. 22, it can be inferred that the SoC_ΔSoC is generally closer to the SoC curves with the fixed horizons of 5 s and 10 s, which is also supported by the analysis in Fig. 21, where the variable prediction horizon of 5 s and 10 s has the highest proportion.

As revealed by the above analysis and discussion, variations in driving conditions and power demand can impact the selection of the optimal prediction horizon. Therefore, employing a variable prediction horizon for speed forecasting in real-time, dynamic driving environments is more appropriate.

Multi-prediction horizons optimization based on deep reinforcement learning

To evaluate the feasibility and effectiveness of using DRL to obtain the optimal prediction horizons, the relationship between DRL rewards and iteration times, the correlation coefficient between the DRL input parameters and prediction horizons, and the acceleration and the corresponding prediction horizon distribution used for velocity prediction are analyzed and discussed.

Fig. 23 illustrates the relationship between DRL rewards and iteration times. From Fig. 23, it can be observed that DRL converges at around 11,500 iterations, with a relatively fast convergence rate and good reward stability. In the initial stages, DRL has not yet learned the pattern of reward changes, and the DRL agent is in a phase of random exploration due to insufficient iterations. As the number of iterations increases, the agent learns towards convergence. When the average reward value stabilizes at around 5, the agent has learned the basic principles between optimal prediction horizons and energy consumption. Moreover, as the number of iterations increases, the positive impact on the reward diminishes.

Fig. 24 displays a comparison between the SoC_ΔSoC and the SoC solved by DB-PEMS based on DRL (SoC_DRL). As depicted in Fig. 24, the overall trends of the two SOC curves are almost identical due to the same

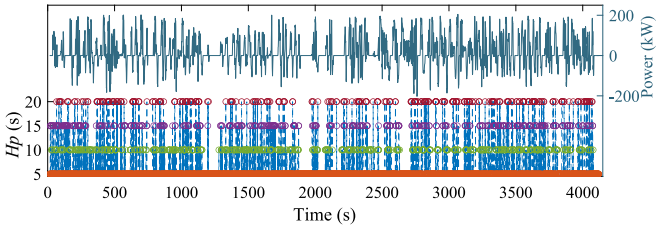


Fig. 21. Demand power and corresponding prediction horizon distribution.

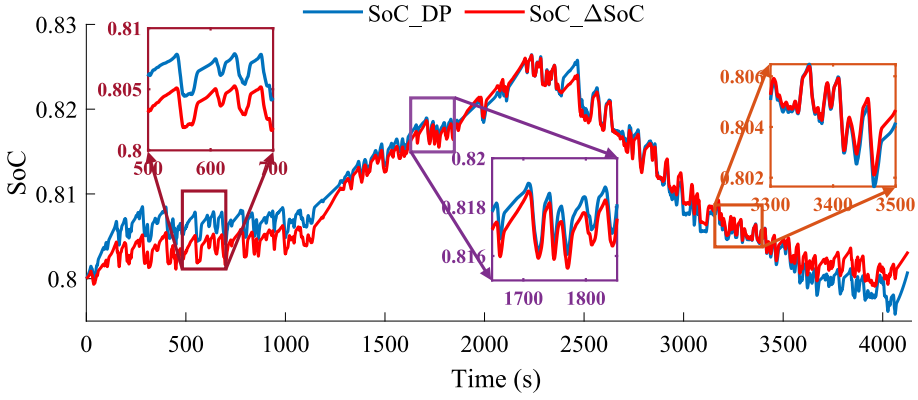


Fig. 20. Comparison between the SoC_DP and the SoC_ΔSoC.

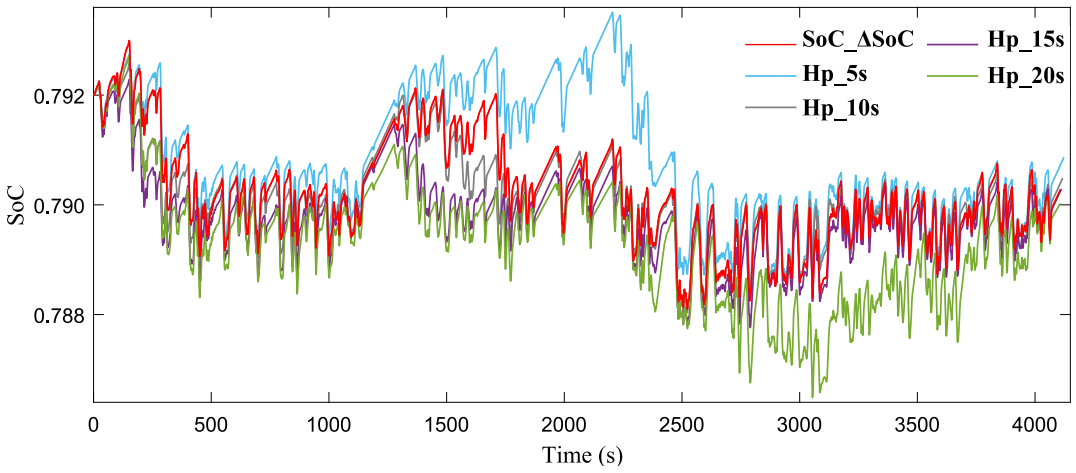


Fig. 22. Comparison between the SoC_ΔSoC and SoC_fixed horizon.

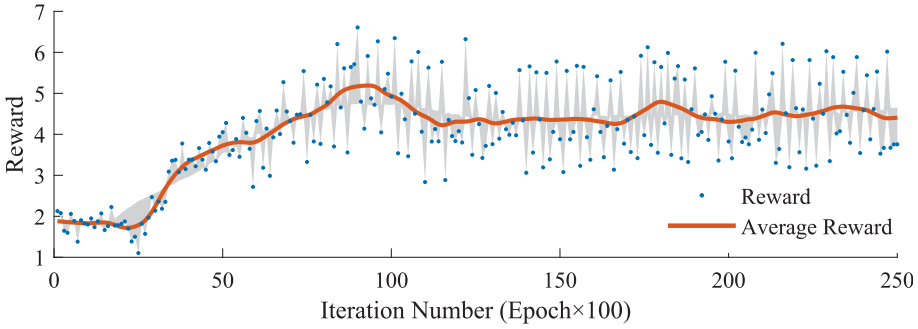


Fig. 23. Relationship between the iteration number and reward of DRL.

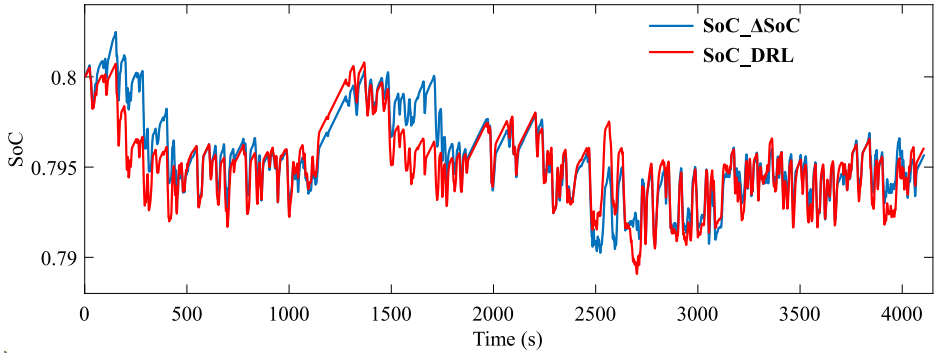


Fig. 24. Comparison between the SoC_ΔSoC and SoC_DRL.

driving conditions, the use of variable horizon for velocity prediction, and the use of ΔSOC as one of the control variables. In some cases, the similarity between the two is very high, such as 500–1100 s, 1900–2450 s, and 2700–4000 s. During these periods, the ΔSOC of the two strategies is highly similar, indicating that the prediction horizons are also similar or identical. In the 0–400 s and 1500–1900 s stages, the trends of the two SOC curves are similar, but the SoC_ΔSoC is higher than the SoC_DRL, indicating that the difference in solving methods between the two strategies still leads to significant differences in the SOC obtained.

In conducting learning for optimal prediction horizon, the input parameters for DRL include velocity, acceleration, driving characteristics (D_chara), and demand power (P_dem). A correlation analysis is performed on these parameters to examine the interrelationships among different parameters and their effects on the selected prediction horizon (H_pre). The results are presented in Table 4. The results in Table 4 reveal that acceleration and demand power have the most significant impact on the prediction horizon, with correlation coefficients of 0.489 and 0.421, respectively. In contrast, driving characteristics and velocity have a relatively minor effect on the prediction horizon.

Drawing on the findings from Table 4, we further analyzed the

Table 4
Correlation coefficient between input parameter of DRL and prediction horizon.

	Speed	Acceleration	D_chara	P_dem	H_pre
Speed	1.000	0.060	−0.237	0.238	0.013
Acceleration	0.060	1.000	0.012	0.928	0.489
D_chara	−0.237	0.012	1.000	−0.001	0.114
P_dem	0.238	0.928	−0.001	1.000	0.421
H_pre	0.013	0.489	0.114	0.421	1.000

relationship between acceleration (the most influential parameter) and the distribution of prediction horizons, as shown in Fig. 25. The acceleration mainly varies between −2 and 2 m/s^2. The prediction horizon of 5 s has the highest frequency, followed by 20 s and then 15 s. The prediction horizon of 10 s accounts for a minor proportion. The 5 s prediction horizon appears most frequently throughout the driving cycle, while the 20 s prediction horizon mainly occurs during the constant or increasing acceleration phase. The 15 s prediction horizon mainly occurs during frequent acceleration fluctuations, such as 0–220 s, 2200–2350 s, and 3800–3910 s. The 10 s prediction horizon is relatively sparse. The other finding is that the same prediction horizon is more likely to cluster.

Additionally, the enlarged view in Fig. 25 reveals that the prediction horizon also tends to become larger as acceleration increases. Furthermore, the 20 s prediction horizon is primarily observed when the acceleration is about to increase from zero and lasts until the acceleration decreases from its peak for a certain period, with a few instances of other prediction horizons occurring in between.

Based on the analysis above, it is evident that the optimal prediction horizon obtained through DRL varies depending on the characteristics of different operating conditions. This implies that adopting different prediction horizons for energy management in real world driving environments may be more reasonable and conducive to fully tapping into energy-saving potential under practical operating conditions.

Comparison and discussion of results of DB-PEMS with multi-horizons

A comparative analysis is conducted to assess the feasibility and effectiveness of two proposed DB-PEMS approaches with algorithm fusion. One is based on ΔSoC approximation, and the other is based on DRL. Furthermore, the variable multi-prediction horizons DB-PEMS is compared to traditional PEMS with fixed prediction horizons to validate the superiority of the proposed strategies.

Fig. 26 presents a comparison of the distribution of prediction horizons for DB-PEMSs with multiple horizons based on the optimal ΔSoC approximation and DRL. Fig. 26 demonstrates that the two strategies share a common feature, namely, the proportion of the prediction horizon of 5 s is the largest and very similar. Specifically, the proportion of the optimal ΔSoC approximation-based DB-PEMS is 68 %, while the proportion of DRL-based DB-PEMS is 70 %. However, the proportions of the other three prediction horizons, which collectively account for about 30 %, differ significantly between the two strategies. Notably, for the optimal ΔSoC approximation-based DB-PEMS, the prediction horizon of 10 s constitutes the largest proportion among the three prediction horizons, at 14 %. Conversely, for the DRL-based DB-PEMS, the prediction horizon of 10 s represents a minor proportion, at only 2 %. In contrast, the prediction horizon of 20 s accounts for the largest proportion at 24

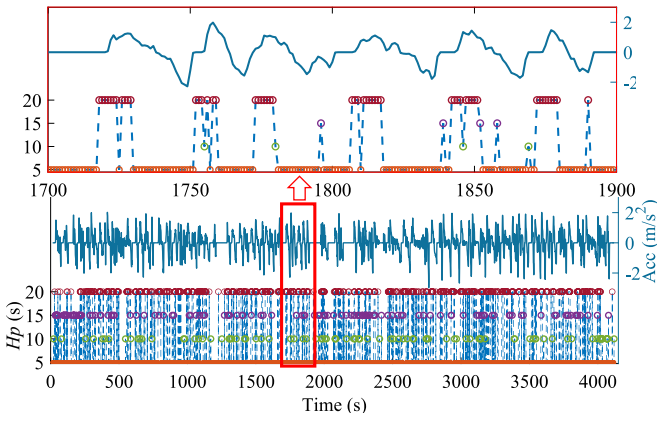


Fig. 25. Relationship between acceleration and corresponding prediction horizon distribution.

%. The difference in prediction horizon selection precisely explains why the SoC curves solved by the two strategies are inconsistent in certain regions.

Based on the proportions of the prediction horizons under the two DB-PEMSs, it is evident that the prediction proportions differ for the same driving condition. Therefore, the optimal prediction horizon selection cannot rely solely on the driving condition features and should consider overall energy consumption.

Table 5 displays the statistical results of the computational time for DB-PEMSs with two types of variable prediction horizons and PEMSs with four fixed prediction horizons of 5 s, 10 s, 15 s, and 20 s. As shown in Table 5, for PEMSs with fixed prediction horizons, the time spent is solely allocated to speed prediction and control optimization. Notably, the time spent on speed prediction is significantly less than on control optimization. This is because the speed predictor is pre-trained offline, and only a negligible amount of time is required for speed prediction in the predictive model. However, as the prediction horizon increases, the time spent on both speed prediction and control optimization increases.

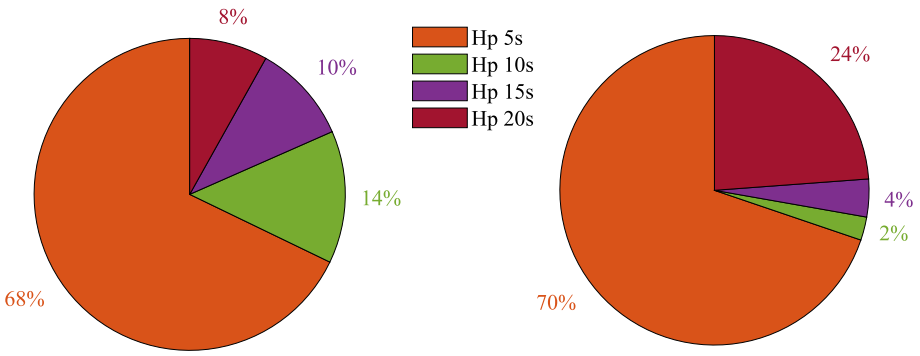
Compared to fixed horizon PEMSs, additional time for selecting the optimal prediction horizon is required for DB-PEMSs, in addition to the time spent on speed prediction and optimization. However, as shown in Table 5, not much time is consumed for horizon selection. Specifically, the horizon selection time for the optimal ΔSoC approximation-based DB-PEMS is 1.225 s, while that for the DRL-based DB-PEMS is 4.751 s. As the optimal horizon selection based on ΔSoC approximation relies on experiential learning, it can be completed relatively quickly. Conversely, the DRL-based approach involves free exploratory learning, necessitating a comparatively longer completion time.

Regarding overall time expenditure, the optimal ΔSoC approximation-based DB-PEMS requires the most time for optimization control among these PEMSs, taking 655.044 s. This is due to the need for parallel computing in prediction and optimization within the optimal ΔSoC approximation-based DB-PEMS. Before determining the optimal prediction horizon, optimization results for the four prediction horizons need to be obtained, resulting in a time cost comparable to that of the PEMS with the longest fixed prediction horizon of the 20 s, leading to the longest total time spent. In contrast, the optimization process of the DRL-based DB-PEMS takes significantly less time, at 325.397 s, with a total time spent of 332.515 s. In terms of timeliness, all of these strategies meet the real-time requirements. However, the DRL-based DB-PEMS incurs less computational burden, while the optimal ΔSoC approximation-based DB-PEMS has the highest computational burden.

Table 6 presents a comparison of energy consumption among PEMSs utilizing two types of variable prediction horizons and four fixed prediction horizons of 5 s, 10 s, 15 s, and 20 s. The benchmark for evaluating the energy-saving effect of the other PEMSs is the global offline optimal solution obtained by DP. The optimal ΔSoC approximation-based DB-PEMS achieves a final SoC of 0.7956, closely approximating the initial value of 0.8. The hydrogen consumption is 5587.96 g/100 km, and the equivalent consumption is 5727.39 g/100 km, representing 92.47 of the benchmark. Compared to the PEMSs utilizing fixed horizons of 5 s, 10 s, 15 s, and 20 s, energy consumption is reduced by 4.31 %, 1.12 %, 1.18 %, and 4.50 %, respectively.

For the DRL-based DB-PEMS, the final SoC is 0.7960, with hydrogen consumption and equivalent hydrogen consumption of 5402.46 g/100 km and 5529.02 g/100 km, respectively, achieving 95.79 % of the benchmark. Compared to PEMSs utilizing fixed prediction horizons of 5 s, 10 s, 15 s, and 20 s, the DRL-based DB-PEMS achieves energy savings of 7.62 %, 4.55 %, 4.60 %, and 7.80 %, respectively. Compared to the optimal ΔSoC approximation-based DB-PEMS, the DRL-based DB-PEMS achieves a 3.59 % reduction in energy consumption.

Fig. 27 shows the FC stack efficiency distributions of PEMSs with two multi-prediction horizons and four fixed prediction horizons (5 s, 10 s, 15 s, and 20 s). The DRL-based DB-PEMS has the largest number of working points in the high-efficiency range greater than 55 %, which is 2792, while the optimal ΔSoC approximation-based DB-PEMS and the



(a) Proportions of optimal ΔSoC approximation-based DB-PEMS

(b) Proportions of DRL-based DB-PEMS

Fig. 26. Comparison of the prediction horizon proportions of the two DB-PEMSs.

Table 5
Computational time of the PEMSs.

Methodology	*Hp* selection	Speed prediction	Optimization	Total time
PEMS with 5 s	——	2.004 s	202.033 s	204.037 s
PEMS with 10 s	——	2.143 s	346.736 s	348.879 s
PEMSs with 15 s	——	2.408 s	479.960 s	482.368 s
PEMSs with 20 s	——	2.601 s	616.005 s	618.606 s
ΔSoC-based DB-PEMS	1.225 s	2.642 s	655.044 s	658.871 s
DRL-based DB-PEMS	4.751 s	2.366 s	325.397 s	332.515 s

Table 6
Comparison of the energy consumption.

Methodology	Final SoC	Hydrogen consumption (g/100 km)	Equivalent hydrogen consumption (g/100 km)
DP	0.7997	5286.08	5296.10
PEMS with 5 s	0.7971	5895.67	5985.05
PEMS with 10 s	0.7962	5672.16	5792.28
PEMSs with 15 s	0.7956	5657.60	5795.60
PEMSs with 20 s	0.7950	5841.177	5997.05
ΔSoC-based DB-PEMS	0.7956	5587.96	5727.39
DRL-based DB-PEMS	0.7960	5402.46	5529.02

PEMSs with a fixed prediction horizon of 10 s have similar operating points of 2623 and 2659, respectively. For the two types of multi-horizons DB-PEMSs, although the number of operating points is less than that of the fixed horizons 10 s and 15 s PEMS in the efficiency range greater than 60 %, they have obvious advantages in the efficiency range of 55 %–60 %. The optimal ΔSoC approximation-based DB-PEMS has 1474 working points, while the DRL-based DB-PEMS has 1574. Moreover, the operating points of the PEMSs with fixed horizons of 10 s and 15 s are significantly reduced in the efficiency range of 55 %–60 %. The 15 s fixed horizon PEMS has the least number of working points, which is 1088. From the perspective of the number of operating points in the high-efficiency range greater than 55 %, the DRL-based DB-PEMS has a clear advantage over other methods.

Although both the optimal ΔSoC approximation-based DB-PEMS and the DRL-based DB-PEMS are algorithmic fusion-based data-driven methods, the former belongs to experiential learning based on the optimal ΔSoC mapping, while the latter is a type of free exploration based on DRL. Nevertheless, both methods have improved energy efficiency compared to fixed prediction horizon PEMS. Furthermore, compared to the optimal ΔSoC approximation-based DB-PEMS, the energy-saving effect of the free-exploration DRL-based DB-PEMS is better, and the complexity of the model has also been reduced.

Conclusions

This paper proposes two types of variable multi-prediction horizon methods for DB-PEMSs with algorithmics fusion: the optimal ΔSoC approximation-based DB-PEMS and the DRL-based DB-PEMS. The former focuses on mining historical data to extract useful information. Although it cannot achieve the optimal global EMS like offline DP, it can approximate the optimal ΔSoC change, which has more potential with larger datasets. The latter is a combination of DRL and MPC applications that adaptively regulates the prediction horizon in PEMS through DRL. Although the DRL-based method has a black-box effect and cannot visualize the optimization process, the results demonstrate that it achieves a better energy consumption optimization effect than the strategy

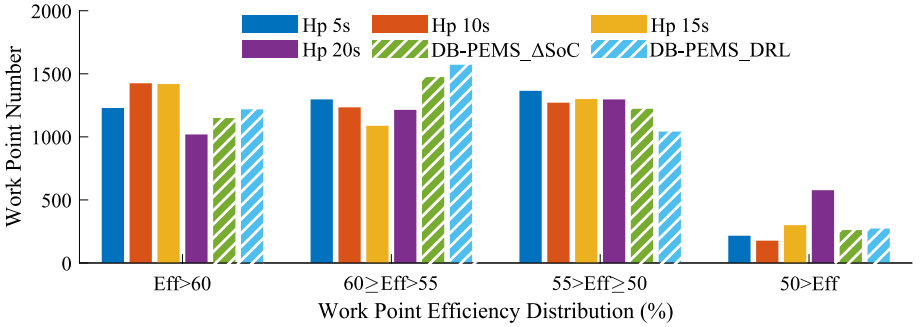


Fig. 27. Fuel cell stack efficiency distributions of the PEMSs.

based on the regularity of historical data. The main conclusions of this study are as follows.

(1) Both multi-horizons DB-PEMSs can realize the self-adaptive adjustment of the velocity prediction horizon under different driving conditions, and the DB-PEMSs with multi-horizons are more conducive to energy saving. Compared to the PEMSs with fixed prediction horizons of 5 s, 10 s, 15 s, and 20 s, the energy consumption of the optimal ΔSoC approximation-based DB-PEMS is reduced by 4.31 %, 1.12 %, 1.18 %, and 4.50 %, respectively, and the DRL-based DB-PEMS achieves energy savings of 7.62 %, 4.55 %, 4.60 %, and 7.80 %, respectively.

(2) The energy consumption, computational cost, and fuel cell stack operating efficiency distribution of DB-PEMSs with multi-horizons and PEMSs with fixed prediction horizons are compared. The DRL-based DB-PEMS has advantages in energy saving and computational cost compared to other strategies. The DRL-based DB-PEMS incurs less computational burden with a total time of 332.515 s. Compared to the best result of PEMS with a fixed prediction horizon of 10 s, the energy consumption is saved by 4.55 %, and compared to the optimal ΔSoC approximation-based DB-PEMS, it is saved by 3.59 %.

CRediT authorship contribution statement

Menglin Li: Conceptualization, Methodology, Validation, Writing – original draft. **Haoran Liu:** Investigation, Visualization, Data curation, Validation. **Mei Yan:** Methodology, Resources, Supervision, Project administration. **Jingda Wu:** Investigation, Visualization, Validation. **Lisheng Jin:** Visualization, Formal analysis, Funding acquisition. **Hongwen He:** Supervision, Software, Resources.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grand No. 52202484), the Hebei Natural Science Foundation (Grand Nos. F2021203118, E2020203174), the Beijing Natural Science Foundation (Grand No. J210007) and the Science and Technology Project of Hebei Education Department (Grand No. QN2022093).

References

[1] Cullen DA, Neyerlin KC, Ahluwalia RK, Mukundan R, More KL, Borup RL, et al. New roads and challenges for fuel cells in heavy-duty transportation[J]. Nat Energy 2021;6(5):462–74.
[2] Alcázar-García D, Romeral Martínez JL. Model-based design validation and optimization of drive systems in electric, hybrid, plug-in hybrid and fuel cell vehicles[J]. Energy 2022;254:123719.
[3] Muthukumar M, Rengarajan N, Velliyangiri B, Omprakas MA, Rohit CB, Kartheek Raja U. The development of fuel cell electric vehicles–A review[J]. Mater Today: Proc 2021;45:1181–7.
[4] Jiang H, Xu L, Li J, Hu Z, Ouyang M. Energy management and component sizing for a fuel cell/battery/supercapacitor hybrid powertrain based on two-dimensional optimization algorithms[J]. Energy 2019;177:386–96.
[5] Hu Z, Li J, Xu L, Song Z, Fang C, Ouyang M, et al. Multi-objective energy management optimization and parameter sizing for proton exchange membrane hybrid fuel cell vehicles[J]. Energy Convers Manage 2016;129:108–21.
[6] Hames Y, Kaya K, Baltacioglu E, Turksoy A. Analysis of the control strategies for fuel saving in the hydrogen fuel cell vehicles[J]. Int J Hydrogen Energy 2018;43 (23):10810–21.
[7] Song Ke, Li F, Hu X, He L, Niu W, Lu S, et al. Multi-mode energy management strategy for fuel cell electric vehicles based on driving pattern identification using learning vector quantization neural network algorithm[J]. J Power Sources 2018; 389:230–9.
[8] Geng C, Jin X, Zhang X. Simulation research on a novel control strategy for fuel cell extended-range vehicles[J]. Int J Hydrogen Energy 2019;44(1):408–20.
[9] Snoussi J, Ben Elghali S, Benbouzid M, Mimouni M. Auto-adaptive filtering-based energy management strategy for fuel cell hybrid electric vehicles[J]. Energies 2018;11(8):2118.
[10] Wang Y, Sun Z, Chen Z. Energy management strategy for battery/supercapacitor/ fuel cell hybrid source vehicles based on finite state machine[J]. Appl Energy 2019; 254:113707.
[11] Xie C, Xu X, Bujlo P, Shen Di, Zhao H, Quan S. Fuel cell and lithium iron phosphate battery hybrid powertrain with an ultracapacitor bank using direct parallel structure[J]. J Power Sources 2015;279:487–94.
[12] Tekin M, Hissel D, Pera M-C, Kauffmann JM. Energy-management strategy for embedded fuel-cell systems using fuzzy logic[J]. IEEE Trans Ind Electron 2007;54 (1):595–603.
[13] Kandi Dayeni M, Soleymani M. Intelligent energy management of a fuel cell vehicle based on traffic condition recognition[J]. Clean Techn Environ Policy 2016;18(6): 1945–60.
[14] Dantzig G B. Programming in a linear structure[J]. Comptroller united states air force washington dc, 1948.
[15] Bellman R. Dynamic programming treatment of the travelling salesman problem [J]. J ACM (JACM) 1962;9(1):61–3.
[16] Boyd S, Boyd SP, Vandenberghe L. Convex optimization[M]. Cambridge: Cambridge University Press; 2004.
[17] Pontryagin LS. The mathematical theory of optimal processes and differential games[J]. Trudy Matematicheskogo Instituta imeni VA Steklova 1985;169:119–58.
[18] Deng K, Peng H, Dirkes S, Gottschalk J, Ünlübayir C, Thul A, et al. An adaptive PMP-based model predictive energy management strategy for fuel cell hybrid railway vehicles[J]. eTransportation 2021;7:100094.
[19] Musardo C, Rizzoni G, Guezennec Y, Staccia B. A-ECMS: an adaptive algorithm for hybrid electric vehicle energy management[J]. Eur J Control 2005;11(4-5): 509–24.
[20] Mayne DQ, Rawlings JB, Rao CV, Scokaert POM. Constrained model predictive control: Stability and optimality[J]. Automatica 2000;36(6):789–814.
[21] Sun C, Hu X, Moura SJ, et al. Velocity predictors for predictive energy management in hybrid electric vehicles[J]. IEEE Trans Control Syst Technol 2014;23(3): 1197–204.
[22] Liu K, Asher Z, Gong X, et al. Vehicle velocity prediction and energy management strategy part 1: deterministic and stochastic vehicle velocity prediction using machine learning[R]. SAE Technical Paper 2019;01:1051.
[23] Wang H, Huang Y, Khajepour A, He H, Cao D. A novel energy management for hybrid off-road vehicles without future driving cycles as a priori[J]. Energy 2017; 133:929–40.
[24] Vatanparvar K, Faezi S, Burago I, Levorato M, Al Faruque MA. Extended range electric vehicle with driving behavior estimation in energy management[J]. IEEE Trans Smart Grid 2019;10(3):2959–68.
[25] Zhou Y, Ravey A, Péra M-C. Multi-objective energy management for fuel cell electric vehicles using online-learning enhanced Markov speed predictor[J]. Energ Conver Manage 2020;213:112821.
[26] Yang C, You S, Wang W, Li L, Xiang C. A stochastic predictive energy management strategy for plug-in hybrid electric vehicles based on fast rolling optimization[J]. IEEE Trans Ind Electron 2020;67(11):9659–70.
[27] Guo L, Zhang X, Zou Y, Guo N, Li J, Du G. Cost-optimal energy management strategy for plug-in hybrid electric vehicles with variable horizon speed prediction and adaptive state-of-charge reference[J]. Energy 2021;232:120993.
[28] Tang X, Jia T, Hu X, Huang Y, Deng Z, Pu H. Naturalistic data-driven predictive energy management for plug-in hybrid electric vehicles[J]. IEEE Trans Transp Electrif 2021;7(2):497–508.
[29] Zhao S, Mi CC. A two-stage real-time optimized ev battery cooling control based on hierarchical and iterative dynamic programming and mpcJ]. IEEE Trans Intell Transp Syst 2021;23(8):11677–87.
[30] Sun C, Guanetti J, Borrelli F, Moura SJ. Optimal eco-driving control of connected and autonomous vehicles through signalized intersections[J]. IEEE Internet Things J 2020;7(5):3759–73.
[31] Basso R, Kulcsár B, Egardt Bo, Lindroth P, Sanchez-Diaz I. Energy consumption estimation integrated into the electric vehicle routing problem[J]. Transp Res Part D: Transp Environ 2019;69:141–67.
[32] Lv Y, Duan Y, Kang W, Li Z, Wang F-Y. Traffic flow prediction with big data: a deep learning approach[J]. IEEE Trans Intell Transp Syst 2014;16(2):865–73.
[33] He H, Wang Y, Han R, Han Mo, Bai Y, Liu Q. An improved MPC-based energy management strategy for hybrid vehicles using V2V and V2I communications[J]. Energy 2021;225:120273.
[34] Zhang Y, Chu L, Ou Y, Guo C, Liu Y, Tang X. A cyber-physical system-based velocity-profile prediction method and case study of application in plug-in hybrid electric vehicle[J]. IEEE Trans Cybern 2021;51(1):40–51.
[35] Liu Y, Li J, Chen Z, Qin D, Zhang Yi. Research on a multi-objective hierarchical prediction energy management strategy for range extended fuel cell vehicles[J]. J Power Sources 2019;429:55–66.
[36] Liu J, Wang Z, Hou Y, Qu C, Hong J, Lin Ni. Data-driven energy management and velocity prediction for four-wheel-independent-driving electric vehicles[J]. eTransportation 2021;9:100119.
[37] Jinquan G, Hongwen He, Jiankun P, Nana Z. A novel MPC-based adaptive energy management strategy in plug-in hybrid electric vehicles[J]. Energy 2019;175: 378–92.
[38] Yan M, Xu H, Jin L, He H, Li M, Liu H. Co-optimization for fuel cell buses integrated with power system and air conditioning via multi-dimensional prediction of driving conditions[J]. Energy Convers Manage 2022;271:116339.

[39] Yan M, Li G, Li M, He H, Xu H, Liu H. Hierarchical predictive energy management of fuel cell buses with launch control integrating traffic information[J]. Energy Convers Manage 2022;256:115397.
[40] Zhang Xi, Mi C. Vehicle Energy Management: Modeling, Control, and Optimization [M]. Mechanical Industry Press; 2013.
[41] Razi M, Murgovski N, McKelvey T, Wik T. Design and comparative analyses of optimal feedback controllers for hybrid electric vehicles. IEEE Trans Veh Technol 2021;70(4):2979–93.
[42] Yan M, Li M, He H, Peng J. Deep learning for vehicle speed prediction[J]. Energy Procedia 2018;152:618–23.
[43] Wu J, He H, Peng J, Li Y, Li Z. Continuous reinforcement learning of energy management with deep Q network for a power split hybrid electric bus[J]. Appl Energy 2018;222:799–811.