

## Article

# Dual Heuristic Dynamic Programming Based Energy Management Control for Hybrid Electric Vehicles

Yaqian Wang<sup>1,2</sup> and Xiaohong Jiao<sup>1,2,\*</sup> 

<sup>1</sup> Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China; wangyaqian@stumail.ysu.edu.cn

<sup>2</sup> School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China

\* Correspondence: jiaoxh@ysu.edu.cn

**Abstract:** This paper investigates an adaptive dynamic programming (ADP)-based energy management control strategy for a series-parallel hybrid electric vehicle (HEV). This strategy can further minimize the equivalent fuel consumption while satisfying the battery level constraints and vehicle power demand. Dual heuristic dynamic programming (DHP) is one of the basic structures of ADP, combining reinforcement learning, dynamic programming (DP) optimization principle, and neural network approximation function, which has higher accuracy with a slightly more complex structure. In this regard, the DHP energy management strategy (EMS) is designed by the backpropagation neural network (BPNN) as an Action network and two Critic networks approximating the control policy and the gradient of value function concerning the state variable. By comparing with the existing results such as HDP-based and rule-based control strategies, the equivalent consumption minimum strategy (ECMS), and reinforcement learning (RL)-based strategy, simulation results verify the robustness of fuel economy and the adaptability of the power-split optimization of the proposed EMS to different driving conditions.

**Keywords:** dual heuristic dynamic programming (DHP); hybrid electric vehicle (HEV); backpropagation neural network (BPNN); energy management strategy (EMS)



**Citation:** Wang, Y.; Jiao, X. Dual Heuristic Dynamic Programming Based Energy Management Control for Hybrid Electric Vehicles. *Energies* **2022**, *15*, 3235. <https://doi.org/10.3390/en15093235>

Academic Editor: Daniela Chrenko

Received: 23 March 2022

Accepted: 26 April 2022

Published: 28 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Environmental pollution and a shortage of petroleum resources are urgent problems the world faces. It promotes reducing petroleum demand and exhaust emission to become the top priority for the automobile industry. In recent two decades, automotive and control scholars have made many achievements in energy-saving research on hybrid and electric vehicles [1,2]. It has been demonstrated that the powertrain type, component configuration, and energy management strategy (EMS) play a crucial role among numerous indicators affecting the performance of hybrid and electric vehicles [3]. Accordingly, various powertrain systems and topologies of hybrid electric vehicles (HEVs) are assessed [4,5]. The numerous solutions to the energy management control problem are investigated to improve the fuel economy of HEVs [1,3,6].

According to the main framework of energy management control, the EMS for HEV can be categorized into rule-based control strategy, optimization-based control strategy, and learning-based control strategy.

The deterministic rule-based (RB) EMSs are the first control method applied to HEVs. The rules are derived from heuristic and engineering knowledge to operate HEVs' powertrains without prior knowledge of predefined driving cycles [7]. Although the control structure is simple and easy to realize, it is not easy to fully play the role of HEV to reduce fuel consumption under different driving conditions [8,9]. The fuzzy logic rule-based control strategy is another type of RB EMS, which uses a fuzzy reasoning mechanism to replace the original deterministic logic rules. Similarly, the system action of the conventional

fuzzy logic rule-based control strategy needs to be determined through the experience and intuition of humans, so the control performance is also difficult to guarantee. Thereby, RB energy management control strategies with optimized threshold parameters [8] or fuzzy reasoning using an optimization algorithm [9] are developed to enhance adaptability to driving conditions and improve fuel economy.

The optimization-based EMS realizes the power demand distribution through the optimal solution of the dynamic optimal control problem with the physical constraint conditions transformed from the energy management control problem of HEVs. Therefore, when driving conditions are completely known, there is a global optimization strategy for HEV energy management control; further, the optimal fuel economy can be obtained theoretically for the whole route. Two typical methods deriving the global optimization strategy are dynamic programming (DP) [10,11] and Pontryagin minimum principle (PMP) [12,13]. The deterministic DP-based control strategy entirely depends on specific driving conditions. Furthermore, it is time-consuming and has a heavy calculation burden. As a result, the DP-based EMS is always used as a benchmark strategy for the designed EMS. Meanwhile, various measures are proposed to achieve the online implementation of DP-based energy management, such as the stochastic dynamic programming (SDP) optimization approach [14]. Similarly, it is also challenging to achieve the accurate PMP-based EMS in an actual vehicle due to the heavy computational burden and the uncertainty of the future vehicle driving cycle. In this regard, the equivalent minimum fuel consumption strategy (ECMS) [15–17] emerges, which belongs to the category of real-time optimization. When the equivalence factor (EF) balancing the electrical energy and the fuel energy in the ECMS is linked to the optimal costate of PMP, ECMS is regarded as a realization of PMP-based global optimization problem over the whole of the driving cycle. On the other hand, model predictive control (MPC) is also an apt real-time optimization strategy for handling the energy management control problem of HEVs [18]. There are two challenges in MPC energy management design that need attention: accuracy of the prediction model and real-time online optimization solution in the horizon time domain. For these purposes, a variety of vehicle speed prediction models [19–21] and optimal solution methods [22–24] are proposed and integrated into the MPC-based EMS frameworks.

The learning-based EMSs are derived from machine learning algorithms that have achieved remarkable progress in recent years. Learning-based control strategies can solve the control problems of complex systems that are difficult to be solve with traditional control methods. On this basis, learning-based control strategies such as artificial neural networks (ANNs) and reinforcement learning (RL) have been proposed for control design in the EMSs of HEVs. For example, an intelligent power controller for power-split of HEV is proposed in [25], which is composed of three neural networks (NNs). The NN-based controller, trained offline, needs to use a large amount of data and cannot be applied online under random driving conditions. RL can solve this problem. For example, two EMSs based on RL, namely Q-learning and Dyna, are proposed in [26]. An RL-enabled predictive control strategy with a velocity predictor is proposed in [27] for a parallel HEV. However, these RL algorithms also have problems with the extensive computation and long time consumption when solving complex energy management problems. In this regard, several deep Q-learning (DQL)-based EMSs are developed [28–30], where the NN is used to approximate the Q value, resulting in reduced fuel consumption and training duration. Although these DQL-based methods are continuous in state space, the control action space still needs discretization. Consequently, EMSs are proposed based on Deep Deterministic Policy Gradient (DDPG) with continuous action space tasks [31,32].

From the research on the EMS design, it can be found that the DP algorithm is the most effective method in terms of the global optimality of the solution. However, DP's strict requirements for future information and computational burden limit its practical real-time application. Fortunately, modern adaptive/approximative dynamic programming (ADP) emerges, which is DP, RL, and NN techniques all in one, resulting in ADP effectively solving the dimension disaster problem of DP. Meanwhile, the ADP algorithm has a similar idea to

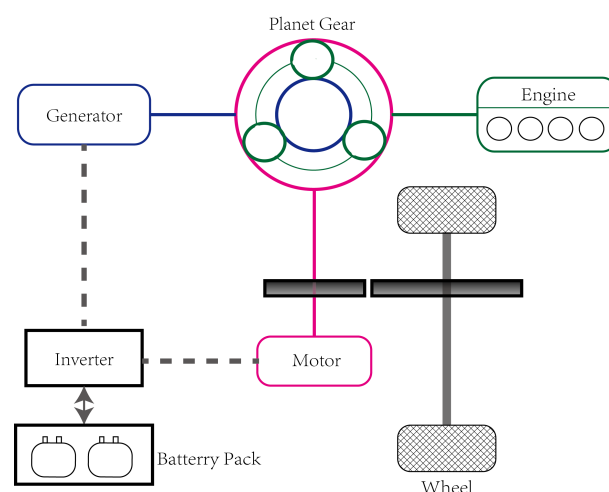
RL, but its structure is more straightforward than RL [33,34]. ADP has relevant applications in the energy management problems of new energy vehicles, such as for the electric vehicle (EV) commercial parking lots [35] and for EV charging stations [36]. Moreover, the heuristic dynamic programming (HDP), a typical control structure of the ADP, has been applied to online EMS of HEVs [37–39]. Dual heuristic dynamic programming (DHP), another typical control structure of the ADP, has substantial benefits in performance over HDP and develops more and more advantages as the number of state variables grows, which has been verified in early simulations studies [33].

Motivated by this, this paper proposes a DHP-based real-time EMS for a series-parallel HEV, to further minimize the equivalent fuel consumption while satisfying the battery level constraints and vehicle power demand. DHP approximates the derivative of the performance index function concerning the state variable in the dynamic programming equation. Compared with the HDP directly approximating the performance indicator itself, and the RL estimating the value function, the DHP can obtain a value function that is closer to the optimal value. The backpropagation neural network (BPNN) acts as the action and critic functions. Cost-to-go is utilized for training the action network. A dual heuristic adaptive critic design is used to update the weights of the critic network in real-time. Compared with the existing methods, this paper mainly has two contributions. Firstly, a DHP-based control strategy is proposed for HEVs, which uses the high precision of the DHP to update the network weights in real-time according to the current driving information to obtain the optimal control and reduce the energy consumption without prior knowledge. Next, the network structure of the action network is adjusted. The added hidden layer nodes make the grid fit the control variables more accurately and make the rate of network weights convergence faster.

The rest of the paper is organized as follows. Section 2 presents the model of HEV and the energy management problem. Section 3 designs a real-time DHP-based energy management control strategy. Section 4 verifies the effectiveness and advantage of the proposed EMS of HEVs by the simulation comparisons with other existing EMSs. The final section is conclusions.

## 2. HEV Model and Problem Description

A series-parallel HEV with a planetary gear set is taken as the research object to complete the design of the EMS for HEVs. The configuration of the HEV system is illustrated in Figure 1, which is adapted from ref. [40]. The powertrain architecture mainly consists of a planetary gear set, an internal combustion engine (ICE), a motor, a generator, and a battery pack. The generator connects to the sun gear, the engine connects to the carrier gear, and the motor connects to the ring gear. The basic specifications for this HEV are listed in Table 1, as in ref. [14].



**Figure 1.** Powertrain configuration of HEV.

**Table 1.** Physical parameters of HEV in simulation.

Parameter [Symbol]	Specification	Unit
Gross vehicle weight [ $M$ ]	1460	[kg]
Tire radius [ $R_{tire}$ ]	0.2982	[m]
Frontal area [ $A$ ]	3.8	[m <sup>2</sup> ]
air density [ $\rho$ ]	1.293	[kg/m <sup>3</sup> ]
drag coefficient [ $C_d$ ]	0.33	[-]
coefficient of rolling resistance [ $\mu_r$ ]	0.015	[-]
transmission efficiency of differential gear [ $\eta_f$ ]	0.97	[-]
Max power	51	[kW]
Motor max power	50	[kW]
Generator max power	30	[kW]
Final differential gear ratio	4.113	[-]
Sun gear teeth number [ $R_s$ ]	30	[-]
Ring gear teeth number [ $R_r$ ]	78	[-]
Max charge capacity [ $Q_{batt}$ ]	6.5	[Ah]

### 2.1. Powertrain Model

Serial-parallel HEVs distribute power using planetary gears. According to the mechanical connections between the gears, there are the following relations:

$$\begin{cases} (R_r + R_s)\omega_c = R_r\omega_r + R_s\omega_s \\ T_r = \frac{R_r}{R_s + R_r}T_c, \quad T_s = \frac{R_s}{R_s + R_r}T_c \end{cases} \quad (1)$$

where  $R$ ,  $\omega$  and  $T$  denote the radius, speed and torque of the gears, respectively.  $s$ ,  $c$ , and  $r$  in the subscript represent the sun gear, carrier gear, and ring gear, respectively.

The dynamic with respect to the rotational speeds of generator, engine and motor can be obtained from Newton's laws:

$$\begin{cases} J_g\dot{\omega}_g = T_s + T_g \\ J_e\dot{\omega}_e = T_e - T_c \\ J_m\dot{\omega}_m = T_m + T_r - \frac{T_{trac}}{g_f} \end{cases} \quad (2)$$

where  $J$  denotes the inertia.  $g$ ,  $e$ , and  $m$  in the subscript represent generator, engine, and motor, respectively.  $T_{trac}$  denotes the torque on the axle of the differential gear, and  $g_f$  denotes the final gear ratio.

Assuming that the connecting shafts are rigid, the following speed relationships hold:

$$\omega_c = \omega_e, \quad \omega_r = \omega_m, \quad \omega_s = \omega_g, \quad \omega_m = \frac{g_f}{R_{tire}}v \quad (3)$$

The dynamics of the vehicle velocity is modeled as:

$$M\dot{v} = \frac{\eta_f T_{trac} - T_{br}}{R_{tire}} - Mg(\mu_r \cos \theta + \sin \theta) - \frac{1}{2}\rho AC_d v^2 \quad (4)$$

where  $M$  and  $g$  denote the vehicle mass and the gravity acceleration, respectively.  $T_{br}$  is the friction brake torque.  $\eta_f$  is the transmission efficiency of differential gear;  $\mu_r$  is the coefficient of rolling resistance;  $\rho$  is air density;  $A$  is frontal area of vehicle;  $C_d$  is drag coefficient; and  $\theta$  is road angle.  $R_{tire}$  is the tire radius.  $v$  is the vehicle velocity.

Fuel consumption is one term of the cost function, which is measured by the fuel mass flow rate  $\dot{m}_f$  and is a function about engine torque and engine speed:

$$\dot{m}_f = BSFC(\omega_e, T_e) \cdot T_e \cdot \omega_e \cdot 10^{-3} \cdot \Delta t / 3600 \quad (5)$$

where  $\Delta t$  is sample time. BSFC is brake specific fuel consumption, which can usually be described by a mapping of engine torque and engine speed. For example, the BSFC diagram of a gasoline engine is shown in Figure 2 [14].

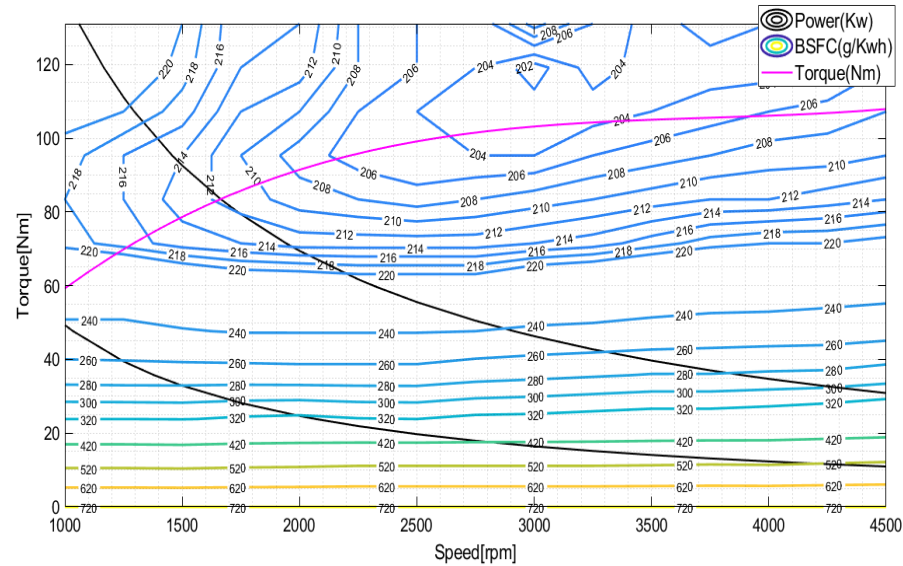


Figure 2. BSFC map of a gasoline engine.

## 2.2. Battery Model

In the driving process, the battery state of charge (SOC) of the non-plug-in HEV needs to maintain near the initial value, and the final value is equal to its initial value. Capturing energy from braking charges the battery, and the electricity converted from the excess engine power also charges the battery. Like fuel consumption, electricity consumption is calculated by the instantaneous rate of change in the battery's internal energy:

$$P_{elec} = U_{oc} \cdot I_{batt} = -U_{oc} \cdot Q_{batt} \cdot \dot{SOC} \quad (6)$$

where  $U_{oc}$ ,  $I_{batt}$  and  $Q_{batt}$  are battery open circuit voltage, current and maximum charge capacity, respectively. The instantaneous change rate of SOC can be expressed as:

$$\dot{SOC} = \frac{-U_{oc} - \sqrt{U_{oc}^2 - 4R_b P_{batt}}}{2Q_{batt} R_b} \quad (7)$$

where  $R_b$  is battery resistance,  $P_{batt}$  is battery power. Both  $U_{oc}$  and  $R_b$  can be fitted as functions of SOC. The following relationship can be found:

$$P_{batt} = \eta_m^{k_m} T_m \omega_m + \eta_g^{k_g} T_g \omega_g \quad (8)$$

where  $\eta_m$  and  $\eta_g$  denote the efficiency of motor and generator, respectively, and  $k_m$ ,  $k_g$  can be given by:

$$k_i = -1, \text{ discharging; or } 1, \text{ charging, } i = m, g \quad (9)$$

## 2.3. Energy Management Optimization Problem

Energy management optimization control for the HEV is to ensure the minimization of the energy consumption during the whole journey while normal drivability of the vehicle. The energy consumption of HEV mainly includes fuel consumption and electric energy consumption, so the optimization goal is to minimize the equivalent fuel consumption. Accordingly, the optimization problem is formulated as follows.

To find the optimal control policy  $u^*(t)$  such that:

$$J^*(t) = \min_u \left\{ J(t) = \sum_{i=t}^{t_{end}-1} \gamma^{i-t} \cdot Q(x(i), u(i), i) \right\}, \quad t \in [t_0, t_{end} - 1] \quad (10)$$

subject to the dynamic constraint condition

$$\begin{cases} SOC(t+1) = SOC(t) + \Delta t \cdot \frac{-U_{oc} - \sqrt{U_{oc}^2 - 4R_b(\eta_m^{k_m} T_m \omega_m + \eta_g^{k_g} T_g \omega_g)}}{2Q_{batt} R_b} \\ v(t+1) = \frac{\Delta t(\eta_f T_{trac} - T_{br})}{M \cdot R_{tire}} - g(\mu_r \cos \theta + \sin \theta) \cdot \Delta t - \frac{1}{2M} \rho A C_d v^2 \cdot \Delta t + v(t) \end{cases} \quad (11)$$

and the physical constraint conditions

$$\begin{cases} SOC_{min} \leq SOC \leq SOC_{max} \\ \omega_{e\_min} \leq \omega_e \leq \omega_{e\_max} \\ \omega_{m\_min} \leq \omega_m \leq \omega_{m\_max} \\ \omega_{g\_min} \leq \omega_g \leq \omega_{g\_max} \\ T_{e\_min}(\omega_e) \leq T_e \leq T_{e\_max}(\omega_e) \\ T_{m\_min}(\omega_m) \leq T_m \leq T_{m\_max}(\omega_m) \\ T_{g\_min}(\omega_g) \leq T_g \leq T_{g\_max}(\omega_g) \end{cases} \quad (12)$$

where  $0 < \gamma \leq 1$  is the discount factor,  $\lambda$  is the weight factor,  $t_{end}$  is the total driving time. The state variable  $x = [v \quad SOC]^T$  and the control input  $u = [T_m \quad \omega_g]^T$ . The instantaneous energy consumption  $Q$  is described as:

$$Q(t) = BSFC \cdot T_e \cdot \omega_e \cdot \Delta t \times 10^{-3}/3600 + \lambda \cdot P_{elec} \cdot \Delta t \cdot 3.6 \times 10^{-3} \quad (13)$$

with

$$\begin{aligned} P_{elec} &= \frac{U_{oc}^2 - U_{oc} \sqrt{U_{oc}^2 - 4R_b P_{batt}(\omega_g, T_m, v)}}{2R_b}, \\ P_{batt}(\omega_g, T_m, v) &= \eta_m^{k_m} T_m \cdot \frac{g_f}{R_{tire}} v + \eta_g^{k_g} \left[ P_{dem} - T_m \cdot \frac{g_f}{R_{tire}} v - T_e \left( \frac{R_s}{R_r + R_s} \omega_g + \frac{R_r}{R_r + R_s} \frac{g_f}{R_{tire}} v \right) \right]. \end{aligned} \quad (14)$$

As a consequence, the optimal control  $u^*(t)$  is determined by solving the minimum of  $J(t)$ .

$$\begin{aligned} u^*(t) &= \arg \lim_{u(t)} \{ J(t) = \sum_{i=t}^{t_{end}-1} \gamma^{i-t} \cdot Q(i) \} \\ &= \arg \lim_{u(t)} \{ J(t) = Q(t) + \gamma \cdot J^*(t+1) \} \\ &= \arg \lim_{u(t)} \{ J(t) = Q(t) + \gamma \cdot Q(t+1) + \gamma^2 \cdot J^*(t+2) \} \\ &= \dots \\ &= \{ u^*(t), \dots, u^*(t_{end} - 1) \} \end{aligned} \quad (15)$$

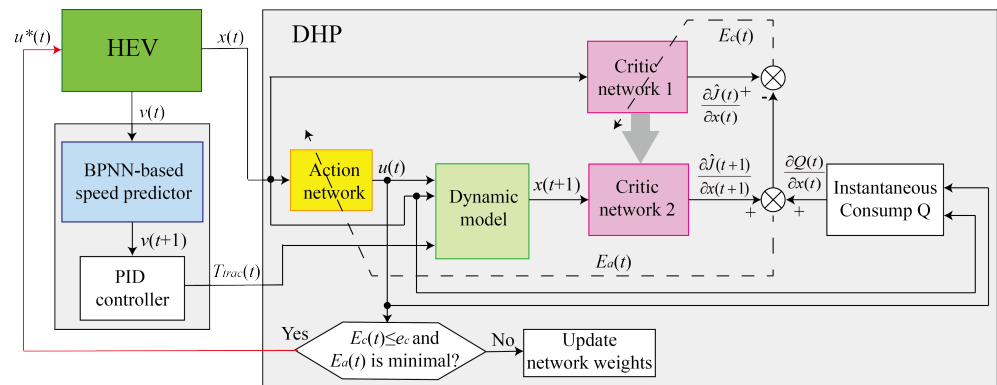
According to Bellman's equation, it can be known that if  $J^*(t+1)$  in the above equation is known at time  $t$ , the value of  $u^*(t)$  can be calculated by the DP method.  $J^*(t+1)$  represents the minimum energy consumption from the time  $t+1$  to the final driving time  $t_{end}$  of the vehicle. However, to obtain  $J^*(t+1)$ , it is necessary to know the driving information of the time  $t+1$  to the final driving time  $t_{end}$  in advance. DP is difficult to achieve energy management control under unknown driving conditions. In addition, DP utilizes the backward iterative algorithm to obtain the optimal solution, which requires a large amount of calculation and takes a long time.

ADP is developed based on DP, and its idea is consistent with the Actor-Critic framework in RL. The NN is used to replace the action model and critic functions. The critic network outputs the approximate value  $\hat{J}(t+1)$  of  $J^*(t+1)$  through inputting state variables, and the action network outputs the approximate value  $\hat{u}(t)$  of  $u^*(t)$ . In this process of obtaining  $J^*(t+1)$ , the ADP method updates the network weight by calculating the error value and optimizes the parameters of the nonlinear function continuously to render  $\hat{J}(t+1)$  gradually to approach  $J^*(t+1)$ . Based on this, the output  $\hat{u}(t)$  of the action net-

work can be seen as  $u^*(t)$ , which is just the optimal solution in the DP. This control strategy does not require the driving information in advance and allows energy management of the HEV to be online and in real-time. Therefore, this paper designs a real-time EMS of HEV using the DHP method, detailed in the next section.

### 3. Design of DHP-Based Real-Time EMS

The diagram structure of the designed DHP-based real-time energy management control system of the HEV is shown in Figure 3.



**Figure 3.** Structure of DHP-based real-time energy management control system of HEV.

The DHP-based real-time EMS is comprised of the speed prediction for obtaining the power demand of the vehicle and the DHP algorithm for the power distribution. The speed prediction model is established by the BPNN offline trained by history traffic data. A PID-type driver model obtains the power demand (traction torque of the vehicle wheel). The DHP algorithm structure includes an action network (AN), a dynamic model, and two critic networks, CN1 and CN2. AN describes the mapping between the state  $x(t)$  and the approximate  $\hat{u}(t)$ , CN1 describes the mapping between the state  $x(t)$  and the derivative of performance indicator  $J$  to state  $x$ ,  $\frac{\partial \hat{J}(t)}{\partial x(t)}$ , and CN2 similarly describes the mapping between the state  $x(t+1)$  and  $\frac{\partial \hat{J}(t+1)}{\partial x(t+1)}$ . The dynamic model represents the HEV model by function relation, which calculates the state value  $x(t+1)$  at the next moment by inputting the approximate value  $\hat{u}(t)$  of the control value.

In the DHP algorithm structure, the error between  $\frac{\partial \hat{J}(t)}{\partial x(t)}$  and  $\frac{\partial Q(t)}{\partial x(t)} + \frac{\partial \gamma \hat{J}(t+1)}{\partial x(t+1)}$  is used to train CNs parameters. This process is executed in a cycle when the HEV is running, and after training the network continuously,  $\frac{\partial \hat{J}(t)}{\partial x(t)} = \frac{\partial Q(t)}{\partial x(t)} + \frac{\partial \gamma \hat{J}(t+1)}{\partial x(t+1)}$  is obtained. At this time, the output  $\hat{u}(t)$  of AN can be regarded as the optimal control value  $u^*(t)$ , and the output  $\frac{\partial \hat{J}(t)}{\partial x(t)}$  of CN1 can be regarded as the optimal co-state  $\frac{\partial J^*(t)}{\partial x(t)}$ . Where  $\hat{J}(t)$  can be regarded as the optimal cost function  $J^*(t)$ . It is worth noting that CN1 and CN2 update weight parameters synchronously, and the two networks are the same. In this control process, the idea and process of this strategy of constantly adjusting network parameters of AN and CN is similar to the RL strategy.

BPNN possesses strong nonlinear fitting and self-learning abilities. Based on this characteristic, BPNN is selected to construct the prediction model, AN, and CN. The detailed design of the speed prediction model, AN, and CN will be introduced below.

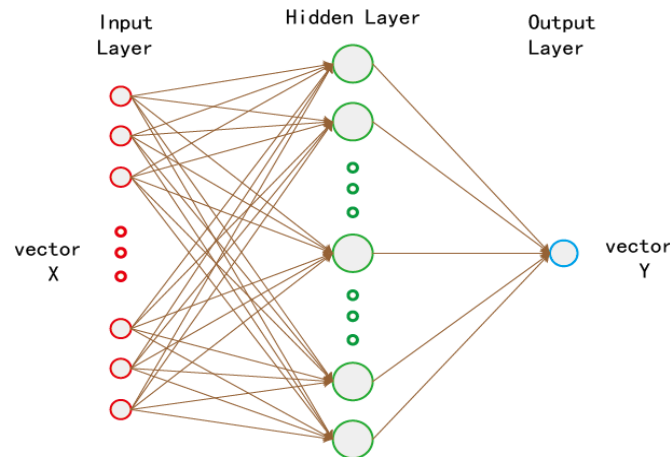
#### 3.1. Speed Prediction Model

The speed prediction model adopts the BPNN structure [41] shown in Figure 4. The objective function of the network can be expressed as



$$\begin{aligned}
 \mathbf{Y} &= F(\mathbf{X}) \\
 \mathbf{X} &= [v_{t-\Delta H}, \dots, v_{t-1}, v_t] \\
 \mathbf{Y} &= [v_{t+1}, v_{t+2}, \dots, v_{t+\Delta P}]
 \end{aligned} \tag{16}$$

where  $F$  represents the objective function of neural network training,  $\Delta P$  and  $\Delta H$  are the predict horizon and input horizon, respectively.



**Figure 4.** Speed prediction model structure.

Many historical vehicle speeds were used to train the BPNN offline, and then the trained network was used to predict the HEV speed online. When the BPNN model is used to predict the speed online, the current speed and the speed of the several previous instants, namely the historical speed sequence, are used as inputs into the speed prediction model to obtain the speed of the prediction sequence. The training set of the speed prediction model is from the actual commuting speed data collected on conventional routes in urban traffic, which is provided by JSAE-SICE benchmark problem 2, see [42].

In the application of speed prediction in this paper, only the speed at the next instant needs to predict, so the output sets as  $v_{t+1}$ .

### 3.2. Design of Critic Network

Figure 5a shows the BPNN structure of CN. Where the input layer has two nodes of two-state values of HEV, the hidden layer number of the network is set as one with five nodes, and the output layer has one node of the output value  $\frac{\partial \hat{J}(t)}{\partial x(t)}$ . The transfer functions of the hidden and output layers are the tansig and purelin functions, respectively. Thus, the nonlinear function of CN can be described as:

$$\begin{cases} \mathbf{c}_1(t) = \mathbf{x}(t) \times \mathbf{W}_{c1}(t) \\ \mathbf{c}_2(t) = \frac{1 - e^{-\mathbf{c}_1(t)}}{1 + e^{-\mathbf{c}_1(t)}} \\ \frac{\partial \hat{J}(t)}{\partial x(t)} = \mathbf{c}_2(t) \times \mathbf{W}_{c2}(t) \end{cases} \tag{17}$$

where  $x(t) = [v, SOC]$ ,  $\mathbf{c}_1$  and  $\mathbf{c}_2$  are the inputs and outputs of hidden layer nodes.  $\mathbf{W}_{c1}$  and  $\mathbf{W}_{c2}$  represent the weight matrices from the input layer to the hidden layer and from the hidden layer to the output layer, respectively, with the structure as follows:

$$\mathbf{W}_{c1} = \begin{bmatrix} W_{11}(t) & W_{12}(t) & W_{13}(t) & W_{14}(t) & W_{15}(t) \\ W_{21}(t) & W_{22}(t) & W_{23}(t) & W_{24}(t) & W_{25}(t) \end{bmatrix} \tag{18}$$

$$\mathbf{W}_{c2}^T = \begin{bmatrix} W_{11}(t) & W_{21}(t) & W_{31}(t) & W_{41}(t) & W_{51}(t) \\ W_{12}(t) & W_{22}(t) & W_{32}(t) & W_{42}(t) & W_{52}(t) \end{bmatrix} \tag{19}$$



The purpose of training CN is to keep the actual value  $\frac{\partial Q(t)}{\partial x(t)} + \frac{\partial \gamma \hat{f}(t+1)}{\partial x(t+1)}$  approaching the target value  $\frac{\partial \hat{f}(t)}{\partial x(t)}$ . Therefore, the error function of CN can be defined as the difference between the target value and the actual value, i.e.,

$$e_c(t) = \frac{\partial \hat{f}(t)}{\partial x(t)} - \frac{\partial Q(t)}{\partial x(t)} - \gamma \cdot \frac{\partial \hat{f}^*(t+1)}{\partial x(t+1)} \quad (20)$$

To make the value of the error function converge to 0, let:

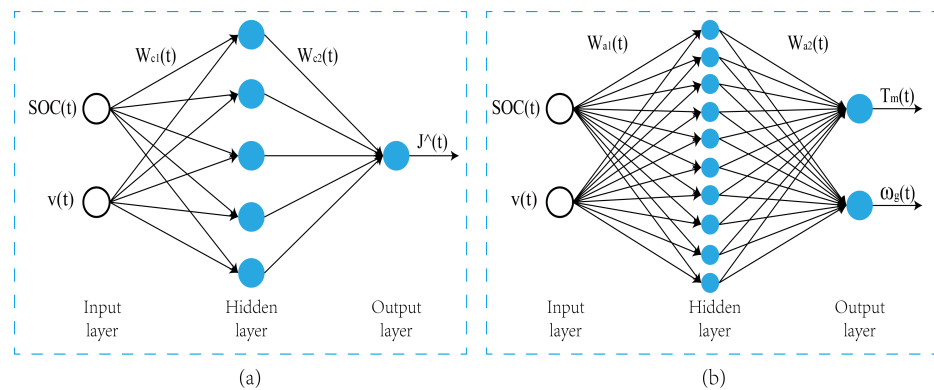
$$E_c(t) = \frac{1}{2} e_c^2(t) \leq \varepsilon_c \quad (21)$$

where  $\varepsilon_c$  is a default error value very close to 0,  $E_c$  is used to train network weights. In order to achieve this goal, we chose the gradient descent algorithm to train  $\mathbf{W}_{c1}$  and  $\mathbf{W}_{c2}$ . If let  $\hat{f}(t) = \frac{\partial \hat{f}(t)}{\partial x(t)}$ , the update of  $\mathbf{W}_{c1}$  and  $\mathbf{W}_{c2}$  are calculated as follows:

$$\begin{cases} \Delta \mathbf{W}_{c1}(t) = \eta_c \cdot \left[ -\frac{\partial E_c(t)}{\partial \mathbf{W}_{c1}(t)} \right] \\ \frac{\partial E_c(t)}{\partial \mathbf{W}_{c1}(t)} = \frac{\partial E_c(t)}{\partial \hat{f}(t)} \cdot \frac{\partial \hat{f}(t)}{\partial \mathbf{c}_2(t)} \cdot \frac{\partial \mathbf{c}_2(t)}{\partial \mathbf{c}_1(t)} \cdot \frac{\partial \mathbf{c}_1(t)}{\partial \mathbf{W}_{c1}(t)} \\ \mathbf{W}_{c1}(t+1) = \mathbf{W}_{c1}(t) + \Delta \mathbf{W}_{c1}(t) \end{cases} \quad (22)$$

$$\begin{cases} \Delta \mathbf{W}_{c2}(t) = \eta_c \cdot \left[ -\frac{\partial E_c(t)}{\partial \mathbf{W}_{c2}(t)} \right] \\ \frac{\partial E_c(t)}{\partial \mathbf{W}_{c2}(t)} = \frac{\partial E_c(t)}{\partial \hat{f}(t)} \cdot \frac{\partial \hat{f}(t)}{\partial \mathbf{W}_{c2}(t)} \\ \mathbf{W}_{c2}(t+1) = \mathbf{W}_{c2}(t) + \Delta \mathbf{W}_{c2}(t) \end{cases} \quad (23)$$

where  $0 < \eta_c \leq 1$  is the learning factor.



**Figure 5.** The BPNN structures of CN and AN. (a) CN. (b) AN.

### 3.3. Design of Actor Network

Figure 5b shows the structure of AN. In the same way, we use BPNN to design AN. The hidden layer number of AN is one in which the input layer has two nodes of the two-state of HEV, the hidden layer has ten nodes, and the output layer has two nodes of the control values  $\hat{u}(t)$ . Experiments show that the network has the best effect when the number of hidden layer nodes is 10. The transfer functions of the hidden layer and output layer are consistent with CN. Thus, the nonlinear function of AN can be described as:

$$\begin{cases} \mathbf{a}_1(t) = x(t) \times \mathbf{W}_{a1}(t) \\ \mathbf{a}_2(t) = \frac{1 - e^{-\mathbf{a}_1(t)}}{1 + e^{-\mathbf{a}_1(t)}} \\ \hat{u}(t) = \mathbf{a}_2(t) \times \mathbf{W}_{a2}(t) \end{cases} \quad (24)$$

where  $\hat{u}(t) = [T_m, \omega_g]$ ,  $\mathbf{a}_1$  and  $\mathbf{a}_2$  are the inputs and outputs of hidden layer nodes.  $\mathbf{W}_{a1}$  and  $\mathbf{W}_{a2}$  represent the weight matrices from input layer to hidden layer and from hidden layer to output layer, respectively.

According to the principle of optimality, the optimal control should satisfy the necessary condition of first-order differential, i.e.,

$$\frac{\partial J^*(t)}{\partial u(t)} = \frac{\partial Q(t)}{\partial u(t)} + \frac{\partial \gamma \cdot J^*(t+1)}{\partial u(t)} \quad (25)$$

so, the optimal control can be obtained:

$$u^*(t) = \arg \lim_{u(t)} \left\{ \left| \frac{\partial J^*(t)}{\partial u(t)} - \frac{\partial Q(t)}{\partial u(t)} - \frac{\partial \gamma \cdot J^*(t+1)}{\partial x(t+1)} \cdot \frac{\partial x(t+1)}{\partial u(t)} \right| \right\} \quad (26)$$

where  $\frac{\partial \gamma \cdot J^*(t+1)}{\partial x(t+1)}$  is the optimal co-state, which can be obtained directly from the output of CN2, and  $\frac{\partial x(t+1)}{\partial u(t)}$  can be obtained from the input–output relation of the dynamic model.

In this paper, we adopt the optimization method to find the optimal control value. In the process of network update iteratively,  $i$  is set as the number of iterations. For every  $\frac{\partial \hat{f}(i)}{\partial x(i)}$  and  $\frac{\partial \hat{f}(i+1)}{\partial x(i+1)}$  of CN at a certain time, search for  $u^*(i)$  until getting the optimal control at that time. Therefore, the update process for  $\mathbf{W}_{a1}$  and  $\mathbf{W}_{a2}$  are as follows:

$$E_a(t) = \frac{1}{2} [u^*(t) - \hat{u}(t)]^2 \leq \varepsilon_a \quad (27)$$

$$\begin{cases} \Delta \mathbf{W}_{a1}(t) = \eta_a \cdot \left[ -\frac{\partial E_a(t)}{\partial \mathbf{W}_{a1}(t)} \right] \\ \mathbf{W}_{a1}(t+1) = \mathbf{W}_{a1}(t) + \Delta \mathbf{W}_{a1}(t) \end{cases} \quad (28)$$

$$\begin{cases} \Delta \mathbf{W}_{a2}(t) = \eta_a \cdot \left[ -\frac{\partial E_a(t)}{\partial \mathbf{W}_{a2}(t)} \right] \\ \mathbf{W}_{a2}(t+1) = \mathbf{W}_{a2}(t) + \Delta \mathbf{W}_{a2}(t) \end{cases} \quad (29)$$

where  $0 < \eta_a \leq 1$  is the learning factor.  $\varepsilon_a$  is a predetermined error value very close to 0.

### 3.4. DHP-Based Real-Time EMS

The specific implementation process of the DHP-based EMS algorithm framework in Figure 3 is as follows:

- (1) Initialize HEV parameters and initialize AN and CNs parameters.
- (2) Input the next time speed  $v(t+1)$  predicted by the speed prediction model and the current time speed  $v(t)$  into the PID controller, and calculate the demand torque  $T_{trac}(t)$  at the current moment;
- (3) According to the current state  $x(t) = [v(t), SOC(t)]$  of HEV, CN1 calculates the co-state  $\frac{\partial \hat{f}(t)}{\partial x(t)}$  by (17), and AN calculates the control  $\hat{u}(t) = [T_m(t), \omega_g(t)]$  by (24);
- (4) According to the control policy obtained in the previous step, the dynamic model calculates the state  $x(t+1) = [v(t+1), SOC(t+1)]$  at the next moment through (11), and calculates the instantaneous energy consumption  $Q(t)$  by (13).
- (5) According to the state  $x(t+1)$  at the next moment, CN2 calculates  $\frac{\partial \hat{f}(t+1)}{\partial x(t+1)}$  by (17), then calculates  $E_c(t)$  and  $u^*(t)$  by (21) and (26), and calculates  $E_a(t)$  by (27).
- (6) Determine whether  $E_c(t)$  is less than or equal to  $\varepsilon_c$ , and whether  $E_a(t)$  is less than or equal to  $\varepsilon_a$ . When any of the conditions cannot be met, the parameters of AN and CN should be updated. Formulas (22) and (23) are used to update weights  $\mathbf{W}_{c1}$  and  $\mathbf{W}_{c2}$  of CN, and Formulas (28) and (29) are used to update weights  $\mathbf{W}_{a1}$  and  $\mathbf{W}_{a2}$  of AN. Then step (3) to step (6) are repeated until both conditions are met, when the next step is carried out.

(7) Provide the optimal control calculated in the current moment to the HEV, then go back to step (2) and continue to calculate the control policy for the next moment until the HEV completes driving.

The detailed algorithm flow of the DHP-based EMS is illustrated in Algorithm 1. Below is the description of the DHP algorithm.

---

**Algorithm 1:** Online learning algorithm of HEV with DHP.

---

**Parameters initialization**

State variable: SOC,  $v$ ; Discount factor:  $\gamma$ ;

Weights in CN:  $\mathbf{W}_{c1}$ ,  $\mathbf{W}_{c2}$ ; Weights in AN:  $\mathbf{W}_{a1}$ ,  $\mathbf{W}_{a2}$ ;

Learning factor of CN and AN:  $\eta_c$ ,  $\eta_a$ ; error value:  $\varepsilon_c$ ,  $\varepsilon_a$ ;

**for**  $i = 1 : t_{end}$

**Speed prediction and demand torque determination**

Getting the current speed  $v(i)$  from HEV;

Running the speed prediction model to obtain  $v(i + 1)$ ;

Using PID controller to get  $T_{trac}(i)$ ;

**Estimating  $\hat{u}(i)$  and  $\frac{\partial \hat{f}(i)}{\partial x(i)}$**

CN1:  $\frac{\partial \hat{f}(i)}{\partial x(i)} = f(\text{SOC}(i), v(i), \mathbf{W}_{c1}(i), \mathbf{W}_{c2}(i))$ ;

AN:  $\hat{u}(i) = f(\text{SOC}(i), v(i), \mathbf{W}_{a1}(i), \mathbf{W}_{a2}(i))$ ;

$\hat{u}(i) = [T_m(i), \omega_g(i)]$ ;

**Calculating  $x(i + 1)$  and  $Q(i)$**

$x(i + 1) = x(i) + \Delta x(i)$ ;

$Q(i) = (\text{BSFC}(i) \cdot T_e(i) \cdot \omega_e(i) / 3600 + \lambda \cdot P_{elec}(i) \cdot 3.6) \times 10^{-3}$ ;

**Calculating  $E_c(i)$  and  $E_a(i)$**

CN2:  $\frac{\partial \hat{f}(i+1)}{\partial x(i+1)} = f(\text{SOC}(i + 1), v(i + 1), \mathbf{W}_{c1}(i), \mathbf{W}_{c2}(i))$ ;

$E_c(i) = \frac{1}{2} [\frac{\partial \hat{f}(i)}{\partial x(i)} - \frac{\partial Q(i)}{\partial x(i)} - \gamma \cdot \frac{\partial \hat{f}^*(i+1)}{\partial x(i+1)}]^2$ ;

$u^*(i) = \arg \lim_{u(i)} \{ |\frac{\partial \hat{f}^*(i)}{\partial u(i)} - \frac{\partial Q(i)}{\partial u(i)} - \frac{\partial \gamma \cdot \hat{f}^*(i+1)}{\partial x(i+1)} \cdot \frac{\partial x(i+1)}{\partial u(i)}| \}$ ;

$E_a(i) = \frac{1}{2} [u^*(i) - \hat{u}(i)]^2$ ;

**Optimal control judgement**

**while**  $E_c(i) > \varepsilon_c$  and  $E_a(i) > \varepsilon_a$

**Weights update**

$\mathbf{W}_{c1}(i) = \mathbf{W}_{c1}(i) + \Delta \mathbf{W}_{c1}(i)$ ;  $\mathbf{W}_{c2}(i) = \mathbf{W}_{c2}(i) + \Delta \mathbf{W}_{c2}(i)$ ;

$\mathbf{W}_{a1}(i) = \mathbf{W}_{a1}(i) + \Delta \mathbf{W}_{a1}(i)$ ;  $\mathbf{W}_{a2}(i) = \mathbf{W}_{a2}(i) + \Delta \mathbf{W}_{a2}(i)$ ;

**Estimating  $\hat{u}(i)$  and  $\frac{\partial \hat{f}(i)}{\partial x(i)}$**

CN1:  $\frac{\partial \hat{f}(i)}{\partial x(i)} = f(\text{SOC}(i), v(i), \mathbf{W}_{c1}(i), \mathbf{W}_{c2}(i))$ ;

AN:  $\hat{u}(i) = f(\text{SOC}(i), v(i), \mathbf{W}_{a1}(i), \mathbf{W}_{a2}(i))$ ;

$\hat{u}(i) = [T_m(i), \omega_g(i)]$ ;

**Calculating  $x(i + 1)$  and  $Q(i)$**

$x(i + 1) = x(i) + \Delta x(i)$ ;

$Q(i) = (\text{BSFC}(i) \cdot T_e(i) \cdot \omega_e(i) / 3600 + \lambda \cdot P_{elec}(i) \cdot 3.6) \times 10^{-3}$ ;

**Calculating  $E_c(i)$  and  $E_a(i)$**

CN2:  $\frac{\partial \hat{f}(i+1)}{\partial x(i+1)} = f(\text{SOC}(i + 1), v(i + 1), \mathbf{W}_{c1}(i), \mathbf{W}_{c2}(i))$ ;

$E_c(i) = \frac{1}{2} [\frac{\partial \hat{f}(i)}{\partial x(i)} - \frac{\partial Q(i)}{\partial x(i)} - \gamma \cdot \frac{\partial \hat{f}^*(i+1)}{\partial x(i+1)}]^2$ ;

$u^*(i) = \arg \lim_{u(i)} \{ |\frac{\partial \hat{f}^*(i)}{\partial u(i)} - \frac{\partial Q(i)}{\partial u(i)} - \frac{\partial \gamma \cdot \hat{f}^*(i+1)}{\partial x(i+1)} \cdot \frac{\partial x(i+1)}{\partial u(i)}| \}$ ;

$E_a(i) = \frac{1}{2} [u^*(i) - \hat{u}(i)]^2$ ;

**end while**

$u^*(i) = \hat{u}(i) \quad u^*(i) \rightarrow \text{HEV}$

**end for**

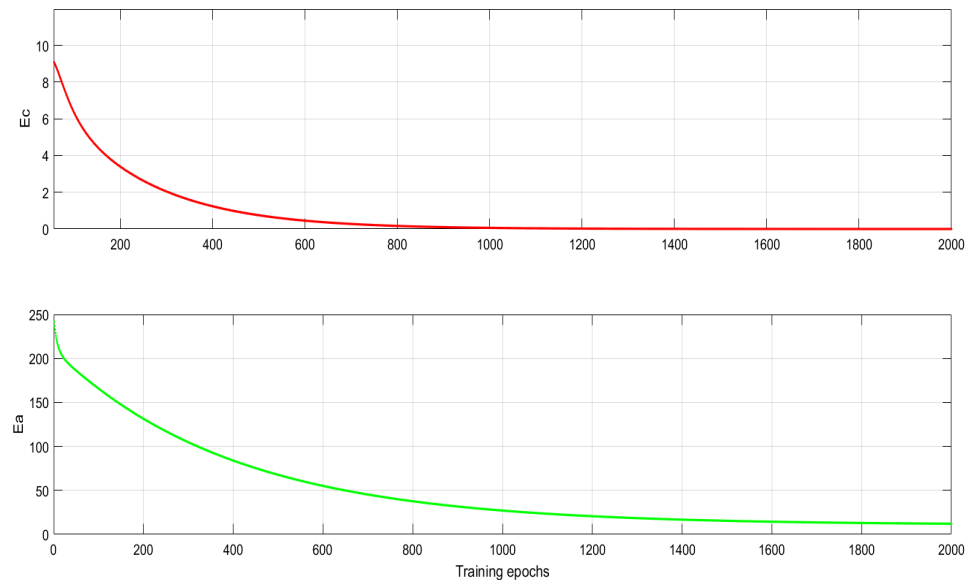
---

#### 4. Simulation Verification and Results Discussion

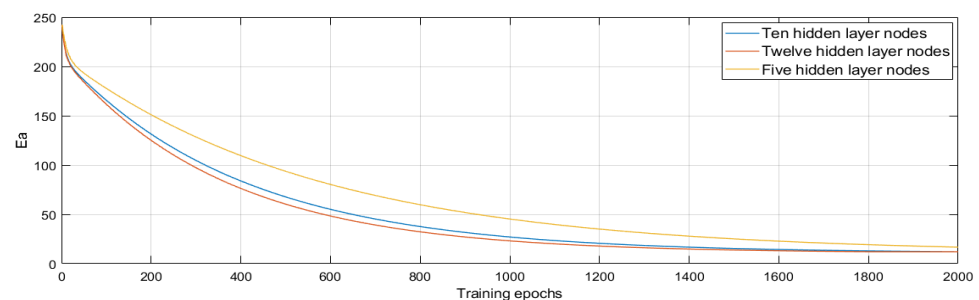
Firstly, the simulation demonstrates the effectiveness and advantages of the BPNN model in the DHP algorithm structure.

For the AN and CNs, the initial network weights are randomly set values, the learning rates are 0.01, the discount factor is 0.95, and the error value is 0.001. The number of network iterations is 2000. The initial value of SOC is 0.5.

Figure 6 shows the convergence curves of  $E_c$  and  $E_a$ . In the process of continuous networks iteration and constant weights update, the value of  $E_c$  approaches zero infinitely, and the value of  $E_a$  approaches a constant. These show the effectiveness of network training. Figure 7 shows the convergence curve of  $E_a$  in different hidden layer nodes of AN. It can be seen from the curve that as the number of nodes in the hidden layer increases, the convergence speed of  $E_a$  accelerates, and the number of iterations when  $E_a$  approaches a constant value also decreases. However, the increase in nodes in the hidden layer will lead to added training time, so the training time should also be considered when considering the convergence effect.



**Figure 6.** The convergence curves of  $E_c$  and  $E_a$ .



**Figure 7.** The convergence curve of  $E_a$  in different hidden layer nodes of AN.

Table 2 shows the number of iterations and training time required by the different number of hidden layer nodes, which means that when the number of hidden layer nodes of AN is 10, the training time and the convergence effect are appropriate. By adjusting network parameters, the training speed is faster. The results show that DHP has a higher algorithm accuracy than the HDP, and the increase in computational load does not significantly prolong the training time.

Table 3 shows the weights of the final training of AN and CN, that is, the weight symbols and their corresponding matrix sequences.

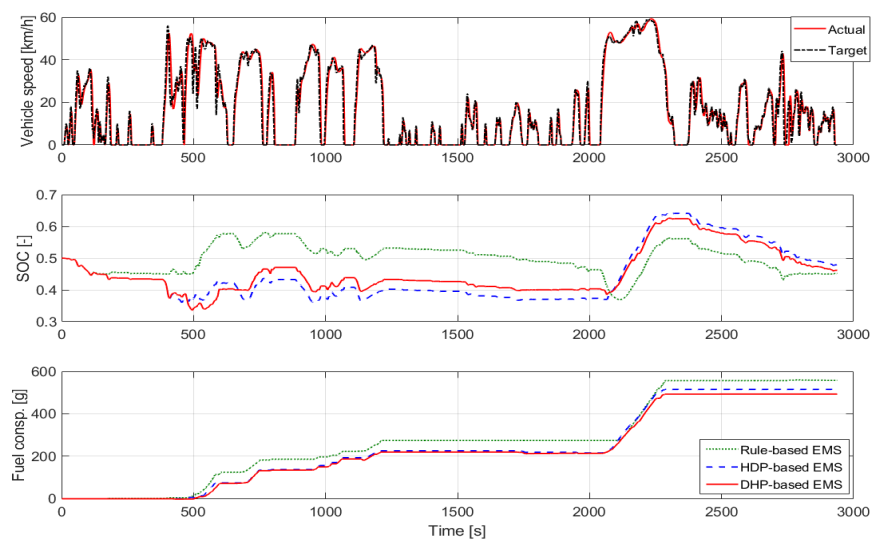
**Table 2.** The number of iterations and training time of different hidden layer nodes.

Hidden Layer Nodes	Iterations	Training Time (s)
5	2800	13,725
10	2000	12,680
12	1800	14,500

**Table 3.** The weight of the CN and AN.

$\mathbf{W}_{c1}(2 \times 5)$											3.819	4.204	4.274	−5.009	−4.597					
											8.426	8.100	8.331	−8.080	−4.597					
$\mathbf{W}_{c2}^T(5 \times 1)$											0.234	1.582	0.471	−1.693	0.520					
$\mathbf{W}_{a1}(2 \times 10)$											0.168	0.881	−0.245	0.039	−0.387	−0.158	0.640	0.133	0.164	0.114
											0.975	0.358	−0.050	0.555	0.115	0.257	0.205	0.368	0.628	0.544
$\mathbf{W}_{a2}^T(10 \times 2)$											−0.384	0.004	0.486	0.250	−0.084	0.541	0.448	0.085	−0.035	0.037
											0.287	0.141	0.645	0.820	0.886	0.776	0.147	0.718	0.476	0.332

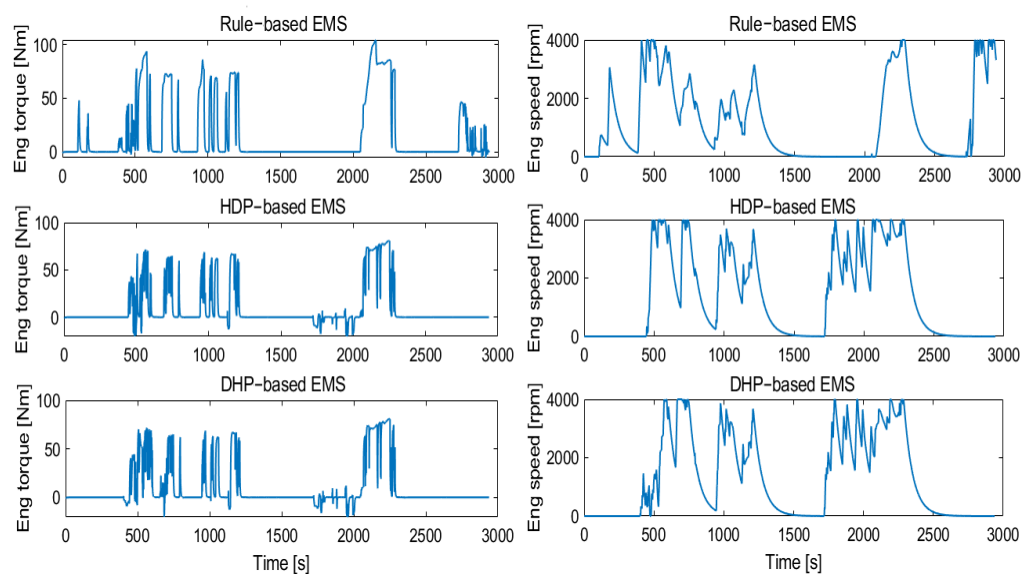
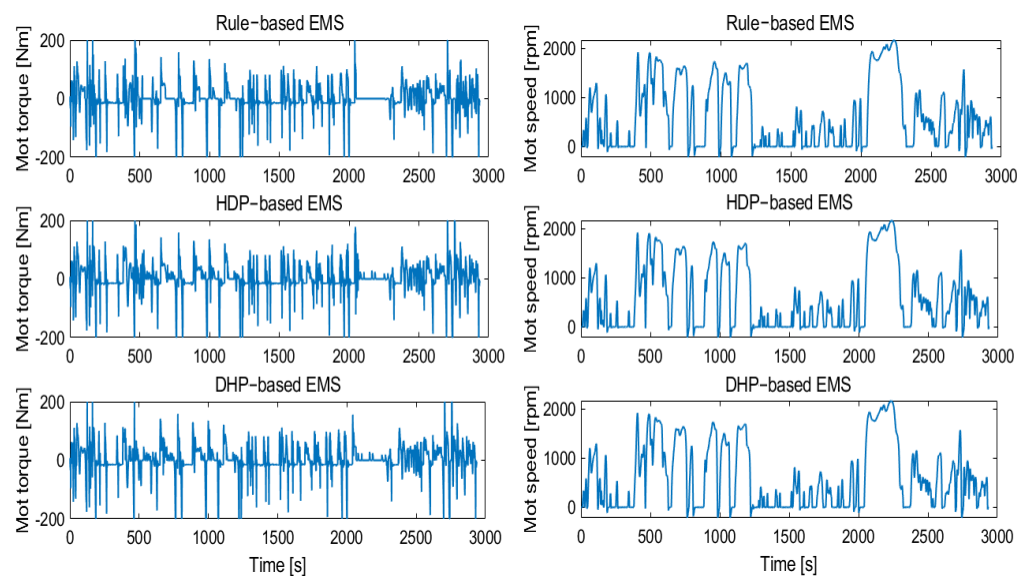
Next, for an actual driving cycle, simulation comparisons with the HDP-based EMS and the RB-based EMS will be given to verify the effectiveness of the DHP-based real-time EMS for the fuel economy under the constraints of driving power demand and SOC. The simulation results of the speed tracking curve, the SOC fluctuation curve, and the equivalent fuel consumption of the three EMSs are shown in Figure 8. Meanwhile, Figure 9–11 show the torque and speed curves of the engine, motor, and generator for these three strategies, respectively.

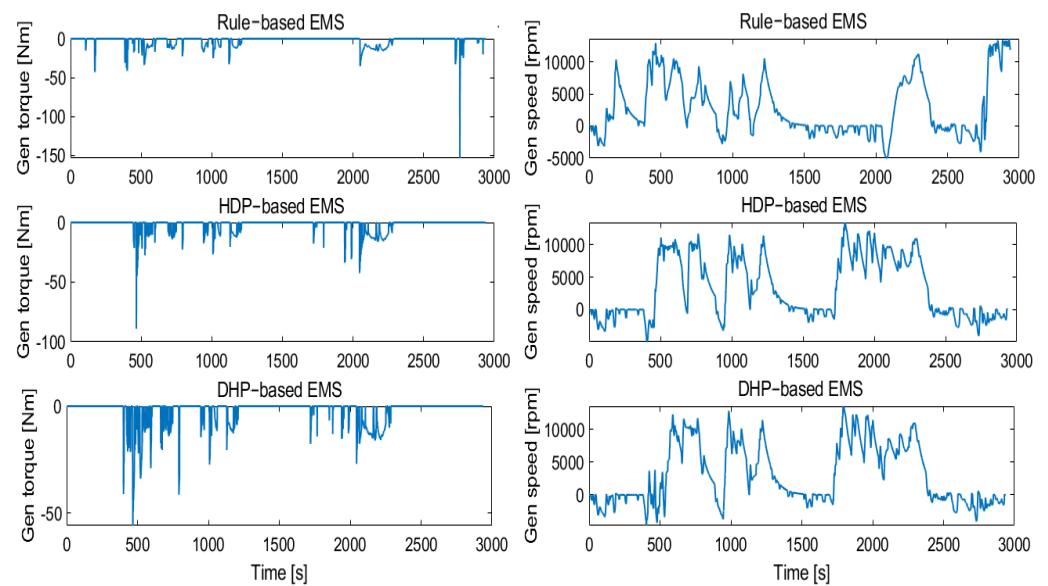
**Figure 8.** Simulation results of DHP-based EMS, HDP-based EMS and RB-based EMS.

For clarity of comparison, the comparison results of the three EMSs are listed in Table 4. From Table 4, the simulation results show that with the difference in SOC final value of the three strategies being small, the equivalent fuel consumption of the DHP-based EMS is the least, which significantly improves the fuel economy.

**Table 4.** Comparisons among DHP-based EMS, HDP-based EMS and RB-based EMS.

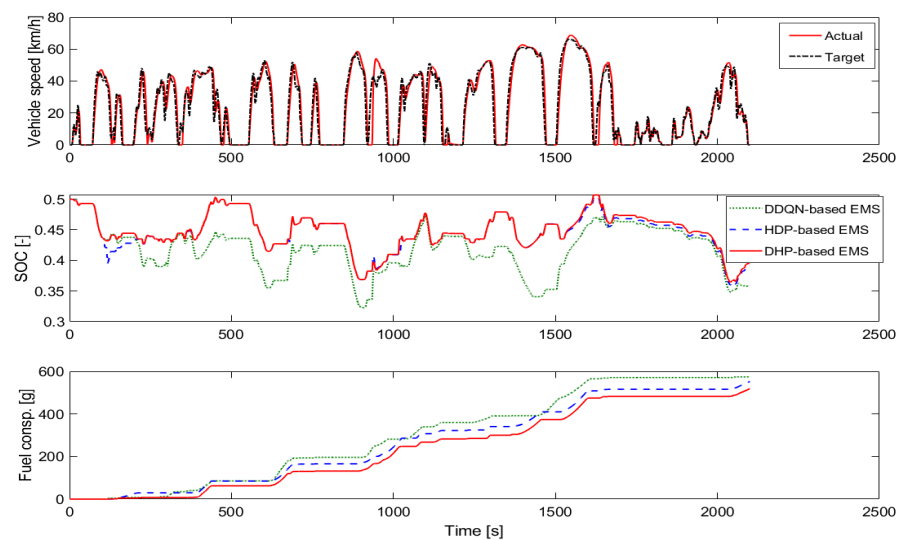
Algorithm	Final SOC	Equival. Fuel Consump. (g)	Reduction (%)
RB	0.452	557.8	—
HDP	0.479	514.9	7.69
DHP	0.462	492.9	11.63

**Figure 9.** Torque and speed of the engine in the three strategies.**Figure 10.** Torque and speed of the motor in the three strategies.



**Figure 11.** Torque and speed of the generator in the three strategies.

The adaptability of the learning-based EMS to different driving conditions is then verified. In order to verify the adaptability of the DHP strategy, the DHP-based EMS is compared with other learning-based EMS in one different driving condition. Here, an energy management strategy based on a double deep Q-Network proposed in [43] is compared, in which the deep neural network is combined with Q-learning in reinforcement learning, having specific adaptability to complex and changeable working conditions. The simulation comparison result of the DHP-based EMS, the HDP-based EMS, and the DDQN-based EMS is shown in Figure 12. The comparisons of the three energy management strategies are also listed in Table 5.



**Figure 12.** Simulation results of DHP-based EMS, HDP-based EMS and DDQN-based EMS.

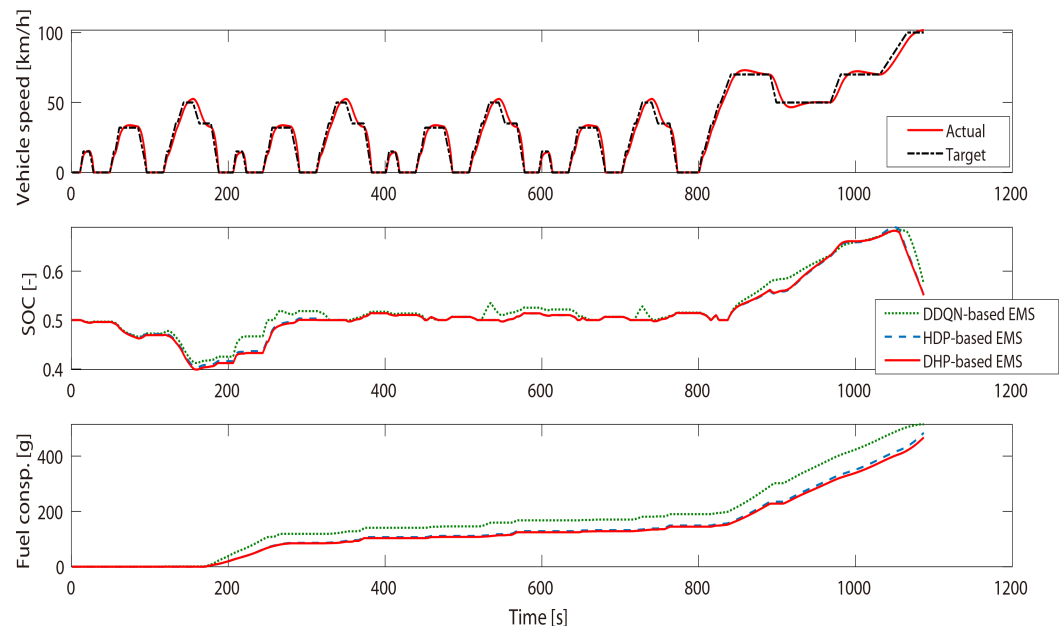


**Table 5.** Comparison of DHP-based EMS, HDP-based EMS and DDQN-based EMS.

Algorithm	Final SOC	Equival. Fuel Consump. (g)	Reduction (%)
DDQN	0.359	574.1	—
HDP	0.391	552.6	3.74
DHP	0.395	519.1	9.58

It can be seen from Figure 12 that the DHP-based EMS makes HEV achieve higher fuel economy under different driving conditions than the others. Specifically, the DHP-based EMS reduces the equivalent fuel consumption by 9.58% compared with the DDQN strategy, and by 6.06% compared with the HDP strategy. Moreover, in DHP-based EMS, the output of CN is co-state, which uses the performance index itself to approximate the optimal control, resulting in the network being able to estimate the solution closer to the optimal value using the co-state training network. When the trained DHP-based EMS is applied in different driving conditions, the network weights in the DHP algorithm will be slightly adjusted according to different driving conditions. The simulation results verify that the proposed DHP-based EMS can continuously learn based on the current driving information and has better adaptability to changeable driving conditions.

Finally, to further verify the adaptability of the proposed EMS to various driving conditions, the above three learning-based EMSs (DDQN, HDP, and DHP) are simulated under the New European Driving Cycle (NEDC). Figure 13 shows the simulation results. Table 6 lists the comparison results of the three strategies. They show that the EMS proposed in this paper has low equivalent fuel consumption when the final SOC values of the three strategies are almost the same. It further indicates that the EMS proposed in this paper has wide applicability.

**Figure 13.** Simulation results of the three learning-based EMSs in NEDC driving cycle.**Table 6.** Comparisons among the three learning-based EMSs in NEDC driving cycle.

Algorithm	Final SOC	Equival. Fuel Consump. (g)	Reduction (%)
DDQN	0.574	515.0	—
HDP	0.552	488.3	5.18
DHP	0.551	480.5	6.70

## 5. Conclusions

This paper designed a DHP-based real-time EMS for HEVs. The EMS aims to achieve maximum energy saving for HEV even without prior driving information. The actor–critic thought in DHP effectively solves the problem of "dimensional disaster" of the traditional optimization algorithm DP. Without knowing all driving information in advance, the DHP can learn and adjust network parameters in real-time to optimize strategies. The proposed EMS was verified by simulation under various driving conditions, including actual and NEDC driving cycles. By analyzing the simulation results, the output torque of the proposed DHP-based EMS can effectively make the HEV track the desired speed, and the speed tracking accuracy can reach more than 95%. Compared with the HDP-based EMS, the accuracy of the DHP algorithm is indeed higher. On the premise of ensuring real-time performance, the DHP-based EMS can further reduce the equivalent fuel consumption of HEV compared with the two existing learning-based EMSs. For the actual driving cycle, the DHP-based EMS reduces the equivalent fuel consumption by 9.58% compared with the DDQN strategy and by 6.06% compared with the HDP strategy. In the NEDC driving cycle, the DHP-based EMS reduces the equivalent fuel consumption by 6.70% compared with the DDQN strategy, and by 1.60% compared with the HDP strategy. The results verify the effectiveness, fuel economy, and adaptability under different driving conditions of DHP-based EMS.

In addition, the states in EMS proposed in this paper are HEV speed  $v$  and SOC. A relatively single selection of state variables is not conducive to fitting the relationship between input and output of the network, and more iterations are needed to make the parameters converge. In future studies, we will design EMS by increasing the state input in the network so that the network can learn more quickly and make corresponding decisions, aiming to further improve the energy consumption optimization performance of the proposed online EMS.

**Author Contributions:** Conceptualization, Y.W. and X.J.; methodology, Y.W.; validation, Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, X.J.; supervision, X.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China grant number 61973265.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Salmasi, F.R. Control strategies for hybrid electric vehicles: Evolution, classification, comparison, and future trends. *IEEE Trans. Veh. Technol.* **2007**, *56*, 2393–2404. [\[CrossRef\]](#)
2. Caiazzo, B.; Coppola, A.; Petrillo, A.; Santini, S. Distributed nonlinear model predictive control for connected autonomous electric vehicles platoon with distance-dependent air drag formulation. *Energies* **2021**, *14*, 5122. [\[CrossRef\]](#)
3. Tran, D.D.; Vafaeipour, M.; Baghdadi, M.E.; Barrero, R.; Van Mierlo, J.; Hegazy, O. Thorough state-of-the-art analysis of electric and hybrid vehicle powertrains: Topologies and integrated energy management strategies. *Renew. Sustain. Energy Rev.* **2020**, *119*, 109596. [\[CrossRef\]](#)
4. Lekshmi, S.; Lal Priya, P.S. Mathematical modeling of electric vehicles—A survey. *Control Eng. Pract.* **2019**, *92*, 104138.
5. Enang, W.; Bannister, C. Modelling and control of hybrid electric vehicles (A comprehensive review). *Renew. Sustain. Energy Rev.* **2017**, *74*, 1210–1239. [\[CrossRef\]](#)
6. Wirasingha, S.G.; Emadi, A. Classification and review of control strategies for plug-in hybrid electric vehicles. *IEEE Trans. Veh. Technol.* **2010**, *60*, 111–122. [\[CrossRef\]](#)
7. Anbaran, S.A.; Idris, N.R.N.; Jannati, M.; Aziz, M.J.; Alsofyani, I. Rule-based supervisory control of split-parallel hybrid electric vehicle. In Proceedings of the 2014 IEEE Conference on Energy Conversion (CENCON), Johor Bahru, Malaysia, 13–14 October 2014; pp. 7–12.

8. Peng, J.; He, H.; Xiong, R. Rule based energy management strategy for a series-parallel plug-in hybrid electric bus optimized by dynamic programming. *Appl. Energy* **2017**, *185*, 1633–1643. [\[CrossRef\]](#)
9. Yang, C.; Liu, K.; Jiao, X.; Wang, W.; Chen, R.; You, S. An adaptive firework algorithm optimization-based intelligent energy management strategy for plug-in hybrid electric vehicles. *Energy* **2022**, *239*, 122120. [\[CrossRef\]](#)
10. Zhu, C.; Lu, F.; Zhang, H.; Sun, J.; Mi, C. A real-time battery thermal management strategy for connected and automated hybrid electric vehicles (CAHEVs) based on iterative dynamic programming. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8077–8084. [\[CrossRef\]](#)
11. Liu, J.; Chen, Y.; Li, W.; Shang, F.; Zhan, J. Hybrid-trip-model-based energy management of a PHEV with computation-optimized dynamic programming. *IEEE Trans. Veh. Technol.* **2017**, *67*, 338–353. [\[CrossRef\]](#)
12. Zheng, C.; Cha, S.W. Real-time application of Pontryagins Minimum Principle to fuel cell hybrid buses based on driving characteristics of buses. *Int. J. Precis. Eng.-Manuf.-Green Technol.* **2017**, *4*, 199–209. [\[CrossRef\]](#)
13. Wang, Y.; Jiao, X. Multi-objective energy management for PHEV via Pontryagin's Minimum Principle and PSO online. *Sci. China Inf. Sci.* **2021**, *64*, 119204. [\[CrossRef\]](#)
14. Jiao, X.; Shen, T. SDP policy iteration-based energy management strategy using traffic information for commuter hybrid electric vehicles. *Energies* **2014**, *7*, 4648–4675. [\[CrossRef\]](#)
15. Onori, S.; Tribioli, L. Adaptive Pontryagins Minimum Principle supervisory controller design for the plug-in hybrid GM Chevrolet Volt. *Appl. Energy* **2015**, *147*, 224–234. [\[CrossRef\]](#)
16. Han, L.; Jiao, X.; Jing, Y. Recurrent-neural-network-based adaptive energy management control strategy of plug-in hybrid electric vehicles considering battery aging. *Energies* **2020**, *13*, 202. [\[CrossRef\]](#)
17. Xie, S.; Hu, X.; Qi, S.; Lang, K. An artificial neural network-enhanced energy management strategy for plug-in hybrid electric vehicles. *Energy* **2018**, *163*, 837–848. [\[CrossRef\]](#)
18. Huang, Y.; Wang, H.; Khajepour, A.; He, H.; Ji, J. Model predictive control power management strategies for HEVs: A review. *J. Power Sources* **2017**, *341*, 91–106. [\[CrossRef\]](#)
19. Shen, P.; Zhao, Z.; Zhan, X.; Li, J.; Guo, Q. Optimal energy management strategy for a plug-in hybrid electric commercial vehicle based on velocity prediction. *Energy* **2018**, *155*, 838–852. [\[CrossRef\]](#)
20. Guo, J.; He, H.; Peng, J.; Zhou, N, T. A novel MPC-based adaptive energy management strategy in plug-in hybrid electric vehicles. *Energy* **2019**, *175*, 378–392.
21. Chen, Z.; Hu, H.; Wu, Y.; Zhang, Y.; Li, G.; Liu, Y. Stochastic model predictive control for energy management of power-split plug-in hybrid electric vehicles based on reinforcement learning. *Energy* **2020**, *211*, 118931. [\[CrossRef\]](#)
22. Xie, S.; Hu, X.; Xin, Z.; Brighton, J. Pontryagins minimum principle based model predictive control of energy management for a plug-in hybrid electric bus. *Appl. Energy* **2019**, *236*, 893–905. [\[CrossRef\]](#)
23. Li, X.; Han, L.; Liu, H.; Wang, W.; Xiang, C. Real-time optimal energy management strategy for a dual-mode power-split hybrid electric vehicle based on an explicit model predictive control algorithm. *Energy* **2019**, *172*, 1161–1178. [\[CrossRef\]](#)
24. Li, T.; Liu, H.; Wang, H.; Yao, Y. Hierarchical predictive control-based economic energy management for fuel cell hybrid construction vehicles. *Energy* **2020**, *198*, 117327. [\[CrossRef\]](#)
25. Park, J.; Chen, Z.; Murphey, Y.L. Intelligent vehicle power management through neural learning. In Proceedings of the 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona, Spain, 18–23 July 2010; pp. 1–7.
26. Liu, T.; Zou, Y.; Liu, D.; Sun, F. Reinforcement learning Cbased energy management strategy for a hybrid electric tracked vehicle. *Energies* **2015**, *8*, 7243–7260. [\[CrossRef\]](#)
27. Liu, T.; Hu, X.; Li, E. S.; Cao, D. Reinforcement learning optimized look-ahead energy management of a parallel hybrid electric vehicle. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 1497–1507. [\[CrossRef\]](#)
28. Inuzuka, S.; Zhang, B.; Shen, T. Real-time HEV energy management strategy considering road congestion based on deep reinforcement learning. *Energies* **2021**, *14*, 5270. [\[CrossRef\]](#)
29. 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. *Appl. Energy* **2018**, *222*, 799–811. [\[CrossRef\]](#)
30. Li, Y.; He, H.; Peng, J.; Wang, H. Deep reinforcement learning-based energy management for a series hybrid electric vehicle enabled by history cumulative trip information. *IEEE Trans. Veh. Technol.* **2019**, *68*, 7416–7430. [\[CrossRef\]](#)
31. Li, Y.; He, H.; Peng, J.; Zhang, H. Power management for a plug-in hybrid electric vehicle based on reinforcement learning with continuous state and action spaces. *Energy Procedia* **2017**, *142*, 2270–2275. [\[CrossRef\]](#)
32. Wu, Y.; Tan, H.; Peng, J.; Zhang, H.; He, H. Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus. *Appl. Energy* **2019**, *247*, 454–466. [\[CrossRef\]](#)
33. Lewis, F.L.; Liu, D. *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*, 1st ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2012; pp. 452–473.
34. Buşoniu, L.; De Schutter, B.; Babuška, R. Chapter of Interactive Collaborative Information Systems. In *Approximate Dynamic Programming and Reinforcement Learning*; Babuška, R., Groen, F.C.A., Eds.; SCI281; Springer: Berlin/Heidelberg, Germany, 2010; pp. 3–44.
35. Sedighzadeh, M.; Mohammadpour, A.; Alavi, S. A daytime optimal stochastic energy management for EV commercial parking lots by using approximate dynamic programming and hybrid big bang big crunch algorithm. *Sustain. Cities Soc.* **2019**, *45*, 486–498. [\[CrossRef\]](#)

36. Wu, Y.; Ravey, A.; Chrenko, D.; Miraoui, A. Demand side energy management of EV charging stations by approximate dynamic programming. *Energy Convers. Manag.* **2019**, *196*, 878–890. [[CrossRef](#)]
37. Liu, J.; Chen, Y.; Zhan, J.; Shang, F. Heuristic dynamic programming based online energy management strategy for plug-in hybrid electric vehicles. *IEEE Trans. Veh. Technol.* **2019**, *68*, 4479–4493. [[CrossRef](#)]
38. Li, G.; Göerges, D. Fuel-efficient gear shift and power split strategy for parallel HEVs based on heuristic dynamic programming and neural networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9519–9528. [[CrossRef](#)]
39. Li, G.; Göerges, D. Ecological adaptive cruise control and energy management strategy for hybrid electric vehicles based on heuristic dynamic programming. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3526–3535. [[CrossRef](#)]
40. Wang, Y.; Jiao, X.; Sun, Z.; Li, P. Energy management strategy in consideration of battery health for PHEV via stochastic control and particle swarm optimization algorithm. *Energies* **2017**, *10*, 1894. [[CrossRef](#)]
41. Xie, S.; Hu, X.; Liu, T.; Qi, S.; Lang, K.; Li, H. Predictive vehicle-following power management for plug-in hybrid electric vehicles. *Energy* **2019**, *166*, 701–714. [[CrossRef](#)]
42. Yasui, Y. JSAE-SICE benchmark problem2: Fuel consumption optimization of commuter vehicle using hybrid powertrain. In Proceedings of the 10th World Congress on Intelligent Control and Automation, Beijing, China, 6–8 July 2012.
43. Zhang, J.; Jiao, X.; Yang, C. A Ddqn-based energy management strategy for hybrid electric vehicles under variable driving cycles. *Energy Technol.* **2021**, 2000770. [[CrossRef](#)]