Cross-type transfer for deep reinforcement learning based hybrid electric vehicle energy management

Renzong Lian, Huachun Tan, *Member, IEEE*, Jiankun Peng, Qin Li, Yuankai Wu

***Abstract*—Developing energy management strategies (EMSs) for different types of hybrid electric vehicles (HEVs) is a time-consuming and laborious task for automotive engineers. Experienced engineers can reduce the developing cycle by exploiting the commonalities between different types of HEV EMSs. Aiming at improving the efficiency of HEV EMSs development automatically, this paper proposes a transfer learning based method to achieve the cross-type knowledge transfer between deep reinforcement learning (DRL) based EMSs. Specifically, knowledge transfer among four significantly different types of HEVs is studied. We first use massive driving cycles to train a DRL-based EMS for Prius. Then the parameters of its deep neural networks, wherein the common knowledge of energy management is captured, are transferred into EMSs of a power-split bus, a series vehicle and a series-parallel bus. Finally, the parameters of 3 different HEV EMSs are fine-tuned in a small dataset. Simulation results indicate that, by incorporating transfer learning (TL) into DRL-based EMS for HEVs, an average 70% gap from the baseline in respect of convergence efficiency has been achieved. Our study also shows that TL can transfer knowledge between two HEVs that have significantly different structures. Overall, TL is conducive to boost the development process for HEV EMS.**

***Index Terms*—Transfer learning, Hybrid electric vehicle, Energy management strategy, Deep reinforcement learning.**

I. Introduction

Due to the dwindling global oil reserves and the increasingly stringent emissions regulations around the world, there is an urgent need to produce more fuel-efficient vehicles [1]. Hybrid electric vehicles (HEVs) are regarded as a promising solution to reduce emissions and save energy under present technological conditions [2]. HEV, which is considered as an in-between product for the transition from conventional fuel vehicles to zero-emission vehicles, is able to achieve outstanding mileage and tailpipe emission by combining the advantages of both fuel and electric vehicles [3].

Hence, the number and types of HEVs show a sharp upward trend in recent years, which leads to a fast-growing prosperous market [4], [5]. According to the types of powertrain, HEVs can be classified into three types of configurations: parallel HEVs, series HEVs and power-split HEVs [3]. In parallel hybrids, internal combustion engine (ICE) and traction motor can transmit power to wheel respectively or jointly, thus boosting the fuel efficiency during operation, especially at high speeds. In series hybrids, ICE serves as a generator to supply power for traction motor and power battery, and traction motor is used to drive the driveline directly, which is advantageous at lower speeds. Unlike the former two configurations, power-split HEVs consolidate the benefits of series and parallel HEVs, hence are promising to achieve lower fuel consumption than series and parallel hybrids. Given the flexibility and polytrope in powertrain structures of HEVs, the demands for rapidly developing efficient energy management strategies (EMSs) for different types of HEVs are rising. EMS targets to minimize fuel consumption as much as possible while maintaining battery charge-sustaining during operation [6]. Numerous EMSs have been proposed to tackle this problem.

Rule-based methods have dominance in industry due to their simplicity and real-time capability [7]. However, its further applications are hindered by the limited optimality and the requirement of human expertise [8]. To reduce the dependencies on the intuition and experience of professional engineers, optimization-based methods are introduced into EMSs to obtain better fuel economy by minimizing total energy consumption or instantaneous energy consumption [9]. In contrast to rule-based ones, these methods are heavily dependent on the prior knowledge of future trip information or the prediction accuracy of velocity, and they are usually computationally expensive [10].

More recently, reinforcement learning (RL) has emerged as a promising solution for HEV EMS. In RL-based approaches, energy management is modeled as a Markov decision process. The optimal solution for EMS can be learned through the interaction between an agent and an environment. Related works have demonstrated that deep reinforcement learning (DRL) based EMSs have a strong learning ability and adaptability under complex driving cycles, and consume less computational resources [11]. The robustness and optimality of DRL-based EMSs have been further verified in [12], [13]. Wu et al. [14] demonstrated that DRL-based EMSs can learn an optimal power-split solution based on multi-source information fusion. The traffic information from intelligent transportation systems can be easily incorporated into DRL-based EMSs to improve the fuel economy. In addition to the theoretical and simulated research advancements, the real-world application of RL-based EMSs is verified by a hardware-in-loop study [15].

The existing EMS researches are concentrated on case studies for one particular type of HEV. For example, Li et al. [16] only studied the optimization of fuel consumption for a parallel HEV, and Chen et al. [17] limited their optimization-based method on a series HEV. Xiang et al. [18] developed a cascaded control strategy for a power-split HEV based on

This work is supported by National Natural Science Foundation of China (Grant No.51705020 & No.61620106002).

R. Lian, Q. Li, Y. Wu are with School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China(e-mail: lianrz612@gmail.com; lqfy123@bit.edu.cn; kaimaogege@gmail.com).

H. Tan and J. Peng are with School of Transportation, Southeast University, Nanjing 211189, China(e-mail: tanhc@seu.edu.cn; pengjk87@gmail.com).

short-term velocity prediction. Those EMSs are often only applicable to a particular type of HEV, e.g. a strategy for power-split HEVs is not applicable to a series HEV. Hence automobile engineers need to redevelop a new strategy according to the characteristics of new HEV. Owing to the complexity of HEV EMSs, it is time-consuming and laborious to develop an effective EMS from scratch for each configuration of HEVs [1], [4]. For instance, in a fuzzy logic system, fuzzy IF-THEN rules are generally established using human expertise, which requires relatively long development period [19]. In addition, there is no standard approach to the control rules formation, and no way to determine whether a prior knowledge that given set of rules is appropriate for the targeted HEV [1]. In order to accelerate the development of HEV EMSs, the transferability and reusability of EMS are significant for automakers. Despite the diversity and abundance of HEV types, it is unquestionable that there will always exist certain commonalities between different types of HEVs owing to the similarities of powertrain and control aim. A solution capable of sharing common knowledge between EMSs of different types could dramatically reduce the development time and efforts required for energy management.

The idea of knowledge transfer across different but related tasks stems from the research in psychology and cognitive science, which indicates that human is able to learn a new target task faster and better by effectively reuse the past experience of similar tasks [20]. Neural network resembles the processing mechanism of human brain. Its front-layers can be used as a feature extractor, and the extracted features are universal [21]. In the field of visual recognition, Oquab et al. [22] had demonstrated that the front-layers of deep neural network trained on the ImageNet dataset can be efficiently transferred to other visual recognition tasks to improve their training efficiency. Given these characteristics of deep neural network, deep transfer learning (DTL) has been widely applied to various real-world fields. The objective of transfer learning (TL) is to extend a set of previous tasks to new unseen tasks, which utilizes the correlations between different tasks [23]. According to the techniques used in DTL, it can be classified into four categories: instances-based method, mapping-based method, network-based method, and adversarial-based method [21].

As mentioned above, a growing number of researches have been verified that DRL-based EMSs have addressed some drawbacks of traditional EMSs to some extent. Although DRL algorithms have made many breakthroughs in many fields, they are limited in the fact that the agent takes a long time to learn the optimal solution by trial-and-error interactions with the environment [24]. Even when encountering a new but similar task, the training process should be restarted from scratch [20]. Furthermore, for different tasks or environments, the choice of optimal hyper-parameters settings varies according to the actual task, which is also a time-consuming work [25]. Hence, in order to improve the training efficiency between similar tasks, several attempts have combined TL with DRL. Rajendran et al. [26] proposed a deep architecture for adaptive policy transfer (ADAAPT), which is able to avoid negative transfer and improve the efficiency of RL agent. Chen et al. [27] proposed a multi-robotic TL framework, which reduces the computational expense in collecting data on robot hardware. Parisotto et al. [28] proposed an ACTOR-MIMIC framework to acquire a generic agent in multiple environments, and transfer the pre-trained model to unseen target tasks to improve learning efficiency. By combining with knowledge graphs, Ammanabrolu et al. [29] demonstrated that knowledge transfer between DRL models facilitates the learning efficiency of control policy.

Overall, deep transfer reinforcement learning (DTRL) is a potential solution for facilitating the development of DRL-based control agents but currently has not been explored in HEV EMSs. In this paper, we study the transferability of DRL-based EMSs for different types of HEVs, and propose a novel DTRL framework for HEV energy management. The DTRL framework is based on a continuous DRL model, named deep deterministic policy gradient (DDPG), which achieves better generalization and avoids discretization error [30]. We demonstrate that the learning efficiency of a particular type of HEV can be improved by knowledge transfer from another HEV with significantly different powertrain structures.

This research encompasses three perspectives that contribute to relevant researches: 1) A novel framework of DRL-based EMS combined with TL is proposed based on a state-of-art DRL algorithm, DDPG. 2) In contrast to previous studies that only target to handle the EMS of a single HEV configuration, the proposed framework is utilized to implement EMSs of different types of HEVs. It is notable that this method can significantly shorten the EMS development cycle for different types of HEVs. 3) We also study TL between EMSs with different control variables. Interestingly the results show that the learning efficiency of an EMS that controls engine and motor can be improved by knowledge reuse from an EMS that only controls engine.

The remainder of this paper is organized as follows. Section II illustrates the basics of TL and RL. Section III illustrates the powertrain models of different types of HEVs and the statistical analysis of driving cycles. Section IV describes the utilization of DTL in HEVs and the evaluation metrics of TL. In section V, the simulation results of the DTRL framework are illustrated, and the transferability of EMSs across different types of HEVs is discussed. The last section includes the conclusion and discussion of this paper.

II. Preliminaries

A. Transfer learning

TL is a unique machine learning method that uses common knowledge to solve problems in different but related fields [31]. Unlike traditional machine learning, TL relaxes the basic assumption that training data must meet the condition of independent and identical distribution with test data, thereby avoiding the reconstruction of statistical models from scratch when the distribution changes. For this reason, it leads to a great influence on many domains, such as computer vision, natural language processing and autonomous driving [23].

DTL mainly studies how to effectively transfer knowledge between different tasks by deep neural networks. According to the definition in [21], a TL task can be defined as

$< D_s, T_s, D_t, T_t, f_\tau(\cdot) >$, where D_s and T_s denote the dataset and the task in source domain, and D_t and T_t denote those in target domain. $f_\tau(\cdot)$ is a non-linear function representing the deep neural network of target task, which can be improved by transferring latent knowledge from source domain. Source domain refers to the task that has been pre-trained by a large amount of training data, while target domain refers to the task that requires the transferred knowledge from source domain and a relatively small amount of training data for rapid training.

B. Reinforcement learning

In general, a RL problem that satisfies the Markov property can be modeled in terms of Markov decision process (MDP) which can be represented as (S, A, P, R, γ). Where S denotes a finite set of states, A denotes a finite set of actions, P denotes a state transition probability matrix, R denotes a reward function, and γ denotes a discount factor.

A RL agent continually learns from interactions with an environment to achieve a goal by maximizing future rewards. More specifically, based on the current states S_t, the agent chooses a set of actions A_t according to the policy $\pi : S \rightarrow P(A)$ that maps states to a probability distribution over the actions, and the environment responses to those chosen actions and delivers new states S_{t+1} and scalar reward R_t to the agent [32]. The reward from the feedback of environment is defined as the sum of discounted future rewards $R_t = \Sigma_{i=t}^{T} \gamma^{i-t} r(s_i, a_i)$ with a discount factor γ in the range of $0 \sim 1$. In the field of RL, the optimal action-value function $Q^\pi(s_t, a_t) = MaxE[R_t | s_t, a_t]$ represents the maximum expected return starting from state S_t, taking action A_t and thereafter following policy π [33]. It is recursively related by the Bellman equation and iteratively updated:

$$Q^\pi(s_t, a_t) = E[r(s_t, a_t) + \gamma \underset{a}{MaxE}[Q^\pi(s_{t+1}, a_{t+1})]] \quad (1)$$

Similarly, in this research, the energy management problem is modeled as the interactions between the EMS agent and the vehicle and traffic information in discrete time step. The optimal EMS solution is derived by a DRL algorithm.

III. HEV MODELLING AND DRIVING CYCLES

A. Configuration of HEVs

In this paper, we study the cross-type TL among four particular types of HEV EMSs. Similarly, all these HEVs are equipped with an engine, a generator, one or two traction motors, and a battery pack, and aim to realize the optimal fuel economy through these components. In this research, Prius acts as the source domain, and the other three types of plug-in HEVs act as the target domain. The reason why we choose Prius as the source task is that Prius is one of the first commercial and the most classical HEVs. EMS for Prius has been extensively studied, therefore the HEV community has gathered much expert knowledge on Prius. Another reason is that the operating range of Prius is more flexible than the other three studied HEVs in this paper. Therefore, the EMS of Pruis can be optimized with a wider range of driving cycles, which makes it more informative. To summarise, extensive operating range and diverse training data enable Prius to act as the source domain more appropriately. The main parameters of these HEVs are listed in table I. It is obvious that the parameters and types of these studied HEVs are significantly different from each other.

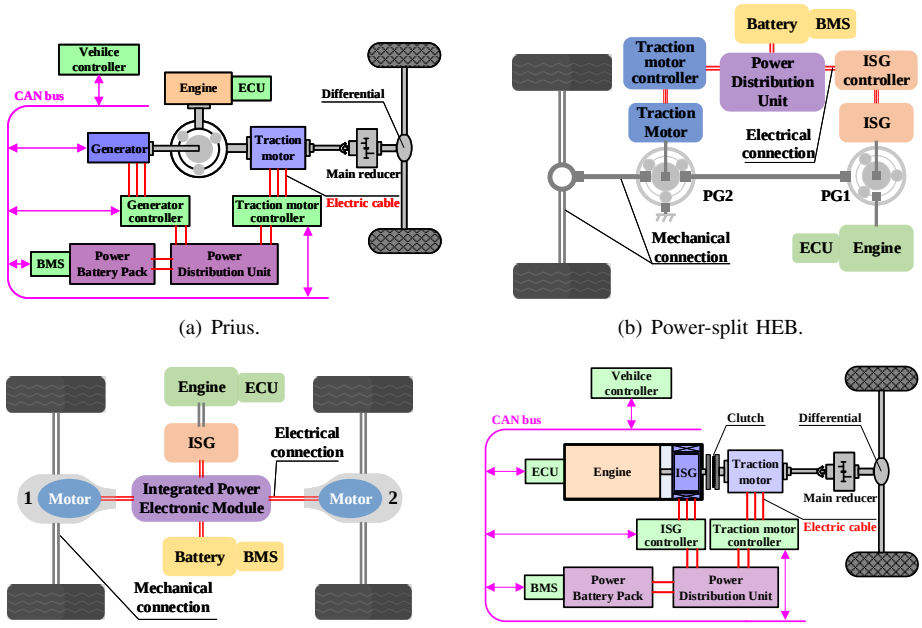
As shown in Fig. 1(a), the core power-split component of Prius [34] is a planetary gear (PG) which splits power among the engine, motor and generator. In this structure, its engine and generator are connected with the planet carrier and sun gear respectively, and its motor is connected with the ring gear that is linked with the output shaft simultaneously. In addition, Prius is equipped with a small capacity Nickel metal hydride (Ni-MH) battery which is used to drive the traction motor and generator. Prius combines the advantages of series and parallel HEVs, and consists of three driving modes: pure electric mode, hybrid mode and charging mode.

The investigated plug-in power-split hybrid electric bus (HEB) [12] is a 12m long bus, whose powertrain mainly consists of a diesel engine, a traction motor and an integrated starter generator (ISG). The power-split mechanism is composed of two sets of planetary gears. In this structure, the diesel engine and ISG are respectively connected with planetary gear 1 (PG1) wherein the ring gear is fixed with the planet carrier of planetary gear 2 (PG2). The sun gear of PG2 is linked with the traction motor, the ring gear of PG2 should be connected to ground, and the coupling power is transmitted to the output shaft through the planet carrier of PG2. Through the cooperation between these two planetary gears, the power-split mechanism of the plug-in power-split HEB is similar to the one of Prius, as shown in Fig. 1(b).

The plug-in series HEV [6] is a passenger car with wheelbase of $2.65m$, its powertrain is shown in Fig. 1(c). In contrast to the former two configurations, the vehicle is impelled by two traction motors whose power sources are composed of the engine-generator set and the battery pack. In this structure, the engine works as a generator to power the traction motors or to recharge the battery.

Compared with the former three types of HEVs, the power-split mechanism and parameters of the plug-in series-parallel HEB [7] are more complex. As shown in Fig. 1(d), the HEB consists of a diesel engine, an ISG, a traction motor, and a battery pack. In this structure, the engine and ISG are integrated together by a torque coupler, and a clutch is set between the ISG and the traction motor. The clutch is allowed to work according to the predetermined rule. In contrast to the EMSs of the former three HEVs that take engine as the control object, the EMS of the plug-in series-parallel HEV needs to control engine and traction motor simultaneously.

In this research, a backward HEV model is built for the training and evaluation of EMS [35]. Energy management system mainly deals with the power allocation among multiple power sources, where the vehicle power demand under the given driving cycle is calculated by the longitudinal force balance equation in equation (2). It mainly consists of four parts: rolling resistance F_{roll}, aerodynamic drag F_{aero}, gradient



(a) Prius.

(b) Power-split HEB.

(c) Series HEV (1: Integrated rear electric powertrain; 2: Integrated front electric powertrain).

(d) Series-parallel HEB (source [7]).

Fig. 1. Schematic graph of four types of HEV powertrain architecture.

TABLE I
MAIN PARAMETERS OF HEVS

Symbol	Parameters	Prius	Power-split HEB	Series HEV	Series-parallel HEB
Vehicle	Curb weight(kg)	1449	16000	3500	13000
	Air resistance coefficient	0.26	0.65	0.65	0.65
	Frontal area(m^2)	2.23	7.48	3.9	5.5
Traction motor	Maximum power(kW)	50	178	/	166
	Maximum torque(Nm)	400	830	Front:170/Rear:320	1800
Generator	Maximum power(kW)	37.8	118	/	166
	Maximum torque(Nm)	75	450	277	500
Engine	Maximum power(kW)	56	140	62	147
	Maximum torque(Nm)	120	730	227	700
Battery	Capacity(kWh)	1.54	26	8.7	29
	Voltage(V)	237	592	347.8	402
Transmission	Final drive ratio	3.93	6.43	5.857	5.33
	Characteristic parameter	2.6	PG1:2.63/PG2:1.98	N/A	N/A

resistance F_{grade}, and inertial force $F_{inertia}$ [36].

$$\begin{cases} F = F_{roll} + F_{aero} + F_{grade} + F_{inertia} \\ F_{roll} = m \cdot g \cdot f \\ F_{aero} = \frac{1}{2}\rho \cdot A_f \cdot C_d \cdot v^2 \\ F_{grade} = m \cdot g \cdot i \\ F_{inertia} = m \cdot a \end{cases} \tag{2}$$

where g is the gravity acceleration, f the rolling resistance coefficient (a function of velocity, tire pressure, external temperature, etc.) which is determined by test, ρ the air density, A_f the frontal area, C_d the coefficient of air resistance, v the longitudinal vehicle velocity without regard to wind speed, i the road slope (road slope is not considered in this paper), m the vehicle mass, a the acceleration.

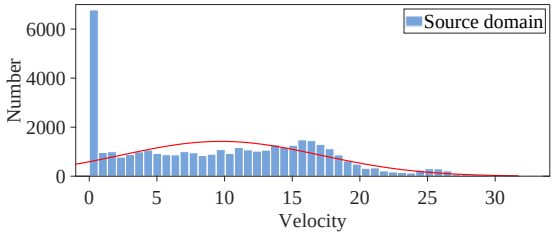
The power units of powertrain system, including engine, generator and motor, are modeled by their corresponding efficiency maps from bench experiments. The battery pack is modeled by an equivalent circuit model in equation (3), wherein the impact of temperature change and battery aging are not considered in this research.

$$\begin{cases} P_{batt}(t) = V_{oc}(t) - R_0 \cdot I^2(t) \\ I(t) = \frac{V_{oc}(t) - \sqrt{V_{oc}^2(t) - 4 \cdot R_0 \cdot P_{batt}(t)}}{2R_0} \\ SoC(t) = \frac{Q_0 - \int_0^t I(t)\,dt}{Q} \end{cases} \tag{3}$$

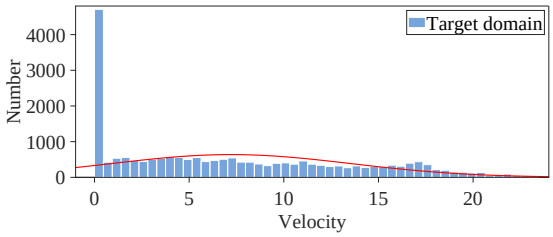
where SoC is the state of charge (SoC), V_{oc} the open-circuit voltage, R_0 the internal resistance, P_{batt} the output power in the charge-discharge cycles, Q_0 the initial battery capacity, Q the nominal battery capacity.

B. Statistical analysis of driving cycles

In this research, driving cycles consist of two parts: the common used standard driving cycles and the historical records of driving cycles collected from passenger and commercial vehicles. Thereinto, the global positioning system (GPS) and the inertial navigation system (OXTS inertial+) are used to collect the real driving cycles from plug-in HEBs. Due to the differences between different types of HEVs, we train their DRL-based EMSs with different driving cycles and divide collected driving cycles into source dataset and target dataset accordingly, in which the amount of data in the source dataset is much larger than that in the target dataset. As stated in section III-A, Prius acts as the source domain, and the other three types of HEVs are regarded as the target domain. Compared with the latter three HEVs, the operation range of Prius is wider than the other three. Hence, as shown in Fig. 2, the source dataset has larger maximum velocity and acceleration, which are $33.4m/s$ and $2.5m/s^2$ respectively. On the contrary, limited by the vehicle configuration and powertrain, the maximum velocity and acceleration of the target domain are $23.5m/s$ and $1.8m/s^2$, which are smaller than those of Prius. The total mileage of the driving cycles in the source domain is $420km$, and that in the target domain is $155km$. The mean and standard deviation of the source dataset are 9.73 and 7.34 respectively, and those of the target dataset are 7.21 and 6.31 respectively. Due to the similarities in vehicle dynamics and traffic environment, the data range of the source and target datasets have a substantial overlap, which may have a positive effect on TL [22].



(a) Velocity distribution of the source dataset



(b) Velocity distribution of the target dataset

Fig. 2. Velocity distributions of the source and target datasets.

IV. Deep Transfer Reinforcement Learning for Energy Management

A. The framework of DRL actor-critic method

Actor-critic methods have been gradually scaled up from virtual simulation to real-world application on a range of challenging control and prediction problems [37], [38]. The ingenious combination of policy search methods and value function has enabled actor-critic algorithm to learn faster [24]. DDPG algorithm, which outputs deterministic policies, is one of the most representative actor-critic methods. It is an off-policy and model-free algorithm, which learns the optimal policy by sampling transitions from a memory pool throughout the training process. Similar to the work of [6], [13], [14], DDPG is used to learn the optimal policy of HEV EMSs. The state and action variables of DDPG-based EMSs are set as follows:

$$\begin{cases} state = \{SoC, v, acc\} \\ action = \{T_{eng}, W_{eng}, T_{mot}\} \end{cases} \tag{4}$$

where v is the velocity of HEV, acc the acceleration of HEV, T_{eng} the engine torque, W_{eng} the engine speed, and T_{mot} the motor torque.

The multi-objective reward function of the DDPG-based EMS can be divided into two parts, the instantaneous energy consumption and the cost of battery charge-sustaining [39]. The control goal is to minimize the energy consumption while maintaining the battery SoC at an appropriate range throughout the trip. Thus, the multi-objective reward function is defined as:

$$reward = -\{\alpha[fuel(t) + elec(t)] + \beta[SoC_{ref} - SoC(t)]^n\} \tag{5}$$

where α represents the weight of fuel and electricity consumption, β represents the weight of battery charge-sustaining, and SoC_{ref} represents the SoC reference value while maintaining battery charge-sustaining.

DDPG uses multilayer perceptron to learn in large state and action spaces, where the optimal action-value function $Q^*(s_t, a_t)$ is represented by a deep neural network $Q(s_t, a_t|\theta^Q)$, namely critic network. The target policy π is represented by a parameterized actor function $u(s|\theta^u)$ which maps states to a set of deterministic actions. The procedure of DDPG algorithm is shown in table II. Furthermore, in order to guarantee the stability and convergence of large non-linear function approximators, a separate target network is introduced into the framework of algorithm to calculate y_t. The critic network is learned by Bellman equation to minimize loss function $L(\theta^Q)$ iteratively, and the actor network is updated by using the sampled policy gradient $\nabla_{\theta^u} J$ [30]:

$$\begin{cases} Q(s_t, a_t|\theta^Q) \approx Q^*(s_t, a_t) = E[r(s_t, a_t) + \gamma Q(s_{t+1}, \mu(s_{t+1}))] \\ y_t = r(s_t, a_t) + \gamma Q^{'}(s_{t+1}, \mu^{'}(s_{t+1}|\theta^{\mu^{'}})|\theta^{Q^{'}}) \\ L(\theta^Q) = E[(Q(s_t, a_t|\theta^Q) - y_t)^2] \\ \nabla_\theta L(\theta^Q) = E[r + \gamma Q^{'}(s_{t+1}, a_{t+1}|\theta^{Q^{'}}) \\ \qquad -Q(s, a|\theta^Q)\nabla_\theta Q(s, a|\theta^Q)Q(s, a|\theta^Q)] \\ \nabla_{\theta^u} J \approx E[\nabla_{\theta^u} Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t|\theta^\mu)}] \\ \qquad = E[\nabla_a Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t)} \nabla_{\theta^u} \mu(s|\theta^u)|_{s=s_t}] \end{cases} \tag{6}$$

Since the architectures of policy and value function networks have a significantly impact on the performance of DDPG algorithm, the actor-critic architecture is designed according to the EMS task as shown in Fig. 3 [40]. The topology of neural networks is pyramid-like, and the number of neurons in hidden layers decreases layer by layer. The performance of

this neural network structure has been validated in a large comparative study [41]. In this structure, ReLU is employed as the activation functions of hidden layers, and the output layer of actor network is processed by a Sigmoid function to map the value of action variables to a range between 0 and 1. Other than Sigmoid, since the series-parallel plug-in HEV requires the traction motor to rotate forward and backward, the activation function of its output layer is set as Tanh function which maps the action values to a range between -1 and 1.

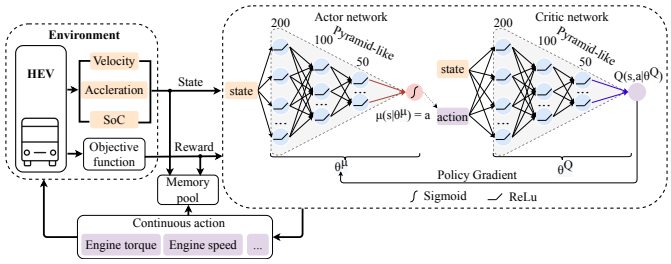


Fig. 3. Agent-environment interaction of HEV energy management.

B. Network-based transfer learning

Based on the transferability of neural networks, network-based DTL is incorporated with DDPG algorithm to realize EMS transfer between the source and target domains. The basic principle is to reuse the partial actor-critic network that has been pre-trained in the source domain, and utilize it to initialize the specific parts of actor-critic network in the target domain, as shown in Fig. 4. The key idea of this framework is that deep learning methods can transform data into internal representations, in which generic features extracted by neural networks can be reused in similar domains [42]. In the context of deep learning, it targets to optimize the spaces of weight ω_i and bias b_i in equation (7) and (8) by backpropagation algorithm, and the parameters of neural networks are the internal representations of data.

Generally, the implementation of this framework can be divided into three steps: the first step is to learn representations of a large number of driving cycles in the source domain until the DDPG algorithm converges to a stable and desired state. Afterwards, a part of internal representations of the source domain, namely the parameters of neural networks $w = \{Actor : [w_1, w_2, w_3, w_4], Critic : [w_{1a}, w_{1s}, w_2, w_3, w_4]\}$ and $b = \{Actor\&Critic : [b_1, b_2, b_3, b_4]\}$, are utilized to initialize the corresponding neural network parameters of the target domain while constantly fixing the network structure. Finally, initialize the other neural network parameters with random initialization and make a subtle tweak to the internal representations of the target domain using a small number of driving cycles.

$$Actor: \begin{cases} Y_1 = ReLu(w_1 \cdot S + b_1) \\ Y_2 = ReLu(w_2 \cdot Y_1 + b_2) \\ Y_3 = ReLu(w_3 \cdot Y_2 + b_3) \\ action = Sigmoid(w_4 \cdot Y_3 + b_4) \end{cases} \tag{7}$$

$$Critic: \begin{cases} Y_1 = ReLu(w_{1s} \cdot S + w_{1a} \cdot A + b_1) \\ Y_2 = ReLu(w_2 \cdot Y_1 + b_2) \\ Y_3 = ReLu(w_3 \cdot Y_2 + b_3) \\ Q\ value = (w_4 \cdot Y_3 + b_4) \end{cases} \tag{8}$$

Fine-tuning the internal representations of partial target network is a key technique that conduces to boosting the training process. In the final step, whether or not to fine-tune the transferred sub-network largely depends on the size of the target dataset and the number of neural network parameters [43]. Given the length of driving cycles described in section III-B, fine-tuning is incorporated into our framework to make a quick adaptation to a new HEV EMS.

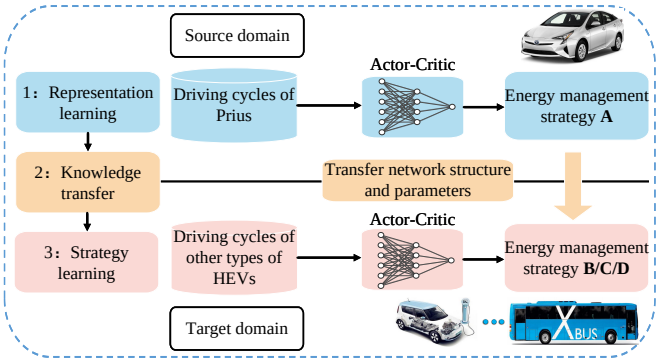


Fig. 4. Sketch map of network-based deep transfer learning.

C. Metrics of transfer learning in EMS

To evaluate the performance of TL on the target tasks, some evaluation metrics are defined particularly according to the EMS transfer task. Based on the definition in [44], there are five metrics to measure the benefits of task transfer: Jumpstart, Asymptotic Performance, Total Reward, Transfer Ratio, and Time to Threshold. In order to verify whether TL is beneficial for training DRL-based EMSs, the performance of a transferred EMS is compared to the baseline of a DDPG model that is initialized with random initialization. For our study on energy management, fuel economy and robustness are selected as the major indexes. Hence, combined with the evaluation metrics of EMS, the evaluation metrics for EMS are redefined as following:

1) Jumpstart: the initial performance of an agent before learning, i.e., the mean performance of fuel economy before the agent starts to learn, as shown in Fig. 5.
2) Robustness: the stability of EMS during training. To evaluate the robustness during the training process, the number of outliers of mean reward $\overline{R}$ is taken as the evaluation metric. Outliers are a set of measured values whose deviation from the mean is twice more than the standard deviation. The number of outliers represents the frequency of fluctuation after the convergence of algorithm. The concrete calculation methods are given

TABLE II
PROCEDURES OF DDPG ALGORITHM.

DDPG algorithm:

1: Initialization: critic network and actor network with weights θ^{Q} and θ^{μ}, target network $Q^{'}$ and $\mu^{'}$ with weights $\theta^{Q'} \leftarrow \theta^{Q}$, $\theta^{\mu'} \leftarrow \theta^{\mu}$, memory pool R, a random process N for action exploration
2: for episode = 1:M do
3: get initial states: v_1, acc_1, SoC_1
4: for t = 1, T do
5: select action $a_t = \mu(s_t|\theta^{\mu}) + N_t$ according to the current policy and exploration noise
6: execute action a_t, observe reward r_t and new states s_{t+1}
7: store transition (s_t, a_t, r_t, s_{t+1}) in R
8: sample a minibatch of transitions (s_t, a_t, r_t, s_{t+1}) from R with priority experience replay
9: set $y_i = r + \gamma Q^{'}(s_{i+1}, \mu^{'}(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
10: update critic by minimizing the loss: $L = \frac{1}{N}\sum_i[(Q(s_i, a_i|\theta^{Q}) - y_i)^2]$
11: update the actor policy using the sampled policy gradient: $\nabla_{\theta^{\mu}} J \approx \frac{1}{N}\sum_i[\nabla_a Q(s, a|\theta^{Q})|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s|\theta^{\mu})|_{s=s_i}]$
12: update the target networks: $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$, $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
13: end for
14: end for

below:

$$\begin{cases} \mu = \sum_{i=k}^{k+N} (\overline{R}_i)/N \\ \sigma = \sqrt{\frac{1}{N}\sum_{i=k}^{k+N} (\overline{R}_i - \mu)^2} \\ outliers \geq (\mu \pm 2\sigma) \end{cases} \tag{9}$$

where the mean μ and the standard deviation σ of mean reward $\overline{R}$ are obtained from the converged episodes of the baseline in the target task, and $(\mu \pm 2\sigma)$ serve as the the upper and lower bounds to determine the outliers for the transferred task.

3) Fuel economy: the overall fuel and electricity consumption of HEVs are calculated in a specific driving cycle, i.e., standard China city driving cycle. For comparison purpose, the energy consumption of dynamic programming (DP) serves as the benchmark.
4) Convergence efficiency: similar to Time to Threshold, convergence efficiency can be interpreted as the training episodes required for convergence, as shown in Fig. 5. In this research, the convergence of strategy given in equation (10) is judged by the distance between two adjacent iteration points and the SoC constraints.

$$\begin{cases} |\overline{R}_{i+1} - \overline{R}_i| \leq \varepsilon \\ SoC_{lo} \leq SoC \leq SoC_{up} \end{cases} \tag{10}$$

where $\overline{R}$ represents the mean reward, SoC_{lo} and SoC_{up} represent the prescribed lower bound and upper bound of SoC respectively.

5) Generalization performance: the performance that is generalized to new unseen driving cycles. In Fig. 6, two pieces of urban driving cycles are used to test the generalization performance of the transferred EMS, where driving cycle 1 is combined with the changes in the number of passengers to test the adaptation to changing environments.

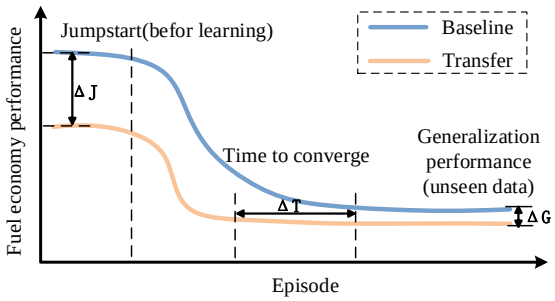
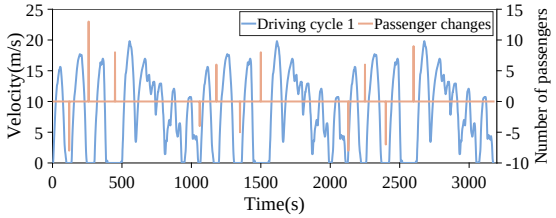
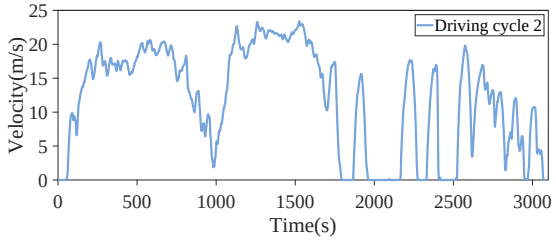


Fig. 5. Evaluation metrics. This figure shows four evaluation metrics: fuel economy, jumpstart, convergence efficiency and generalization performance on new unseen driving cycles. ΔJ, ΔT and ΔG represent the benefits of transfer respectively.



(a) Velocity of driving cycle 1 and number of passengers.



(b) Velocity of driving cycle 2.

Fig. 6. Test data for evaluating generalization performance.

6) Similarity degree: it is used to explain the intrinsic mechanism of EMS transferability between different types of HEVs, where the analysis of intrinsic mechanism mainly focus on the similarity degrees of the neural network parameters or output distributions. The similarity degree is measured by Euclidean distance as follow:

$$dist(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{11}$$

where X and Y represent the neural network outputs or parameters of different HEV EMSs.

V. Simulation Results

In this section, the transferability across four different types of HEVs is analyzed comprehensively. Then, three sets of experiments are conducted to test the feasibility of EMS transfer, and the simulation results are illustrated accordingly.

A. Experiment setup and analysis of transferability

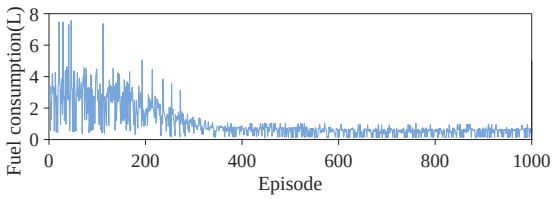
In order to validate the transferability of DRL-based EMSs across different types of HEVs, we set up three sets of EMS transfer experiments, including transfer from Prius to the power-split HEB, from Prius to the series HEV and from Prius to the series-parallel HEB. As our EMS agents are built upon hierarchical neural network layers, and different layers can capture different types of knowledge, the amount of transferable knowledge in different layers is important for our research. In our experiments, four sets of experiments, including transferring the first layer, transferring the first two layers, transferring the first three layers, and transferring all layers, are evaluated. Generally, transferring more layers means that more knowledge from the source domain are utilized by the target domain. Subsequently, the performance of transferred EMS is compared with the baseline that has no transferred knowledge. In addition, six evaluation metrics defined above are used for further comparison in these simulations.

In these experiments, the source task is a sufficiently trained EMS of Prius under the source dataset, as shown in Fig. 7. Using the network-based TL, a partial pre-trained EMS of Prius is transferred to the other three types of HEVs.

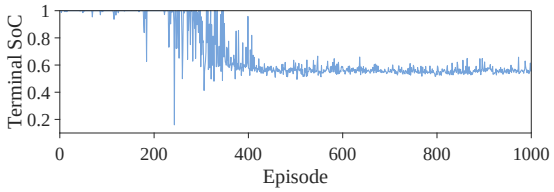
To realize EMS transfer, we mainly focus on the differences across the four types of HEV EMSs, including the vehicle configuration, reward function, dimension of action variables, and driving cycle, as shown in table III. The differences in vehicle configuration lie in the vehicle parameters and types of powertrain. Based on the specific intention, each HEV has specific vehicle parameters, thereby requiring corresponding components and powertrain models. The differences of reward function are reflected in the predetermined SoC reference value for maintaining battery charge-sustaining and the tradeoff between the energy consumption and the cost of battery charge-sustaining. Overall, knowledge transfer is implemented across four different environments.

B. Transfer from Prius to the power-split HEB

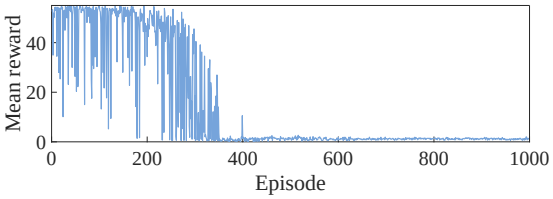
The baseline of the power-split HEB is trained from scratch under the target dataset. In Fig. 8, it can be seen that the



(a) Fuel consumption of Prius.



(b) Terminal SoC of Prius.



(c) Mean reward of Prius.

Fig. 7. Experimental data of Prius.

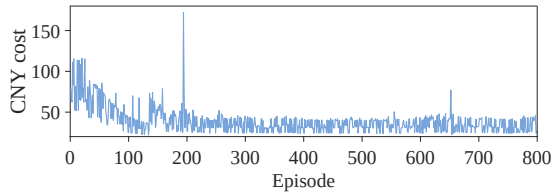
baseline starts to converge at 220th episode, and maintains a better stability thereafter. Based on the mean reward of the baseline, the average μ and standard deviation σ for computing the number of outliers are 4.79 and 2.65, and the upper and lower bounds of outliers are 10.09 and -0.51, respectively. Fig. 9 shows the differences of EMSs between Prius and the power-split HEB, from which we can see that Prius tends to maintain the SoC trajectory above 0.5, but the power-split HEB tries to keep battery SoC sustaining over 0.2.

In contrast to the baseline, we make full use of the pre-trained EMS in the source domain to speed up the policy learning in the target domain. From Fig. 10, it shows that TRL reduces the exploration time during the training process. Although it is free from exploration, the fuel economy in table IV indicates that DTRL-based EMS maintains the same level as the baseline by means of fine-tuning technique directly. It can also be seen that the DTRL-based EMS shows significant superiority in convergence efficiency compared with the baseline, especially the methods transferring the first two layers and the first three layers, which significantly improve the convergence efficiency by 64% and 80%, respectively. In addition, Fig. 11 shows that the methods transferring the first two layers or the first three layers have a smaller number of outliers than the other methods, which means that they own better robustness during the training process. On the contrary, the method transferring all layers performs worse in terms of robustness, which makes it not propitious to real application in HEVs.

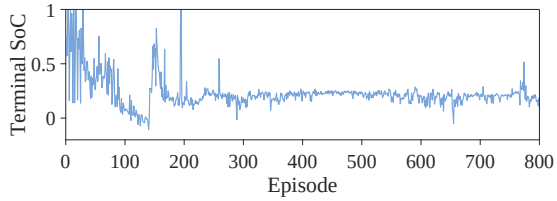
In the aspect of generalization performance, similar to the baseline, DTRL-based EMS also shows strong adaptability to driving cycle 1 through fine-tuning the target network, as show in table IV. As a plug-in HEB, its number of passengers is

TABLE III
DIFFERENCES AMONG FOUR TYPES OF HEVS

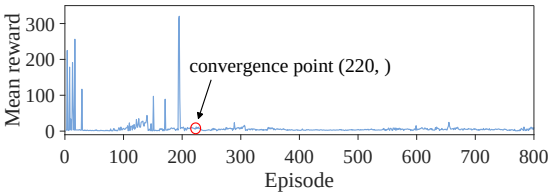
Domain	HEV	Type of powertrain	Reward function	Action variable	Driving cycle
Source domain	Prius	power split by a single PG	$SoC_0 = 0.5$	T_{eng}, W_{eng}	v_{max} : $33.4m/s$ acc_{max} : $2.5m/s^2$ total mileage: $420Km$
Target domain	Power-split HEB	power split by two PGs	$SoC_0 = 0.2$	T_{eng}, W_{eng}	v_{max} : $23.5m/s$ acc_{max} : $1.8m/s^2$ total mileage: $155Km$
	Series HEV	series mode	$SoC_0 = 0.1$	T_{eng}, W_{eng}	
	Series-parallel HEB	power split by a clutch (series and parallel modes)	$SoC_0 = 0.1$	T_{eng}, W_{eng}, T_{mot}	



(a) Fuel consumption of the power-split HEB.

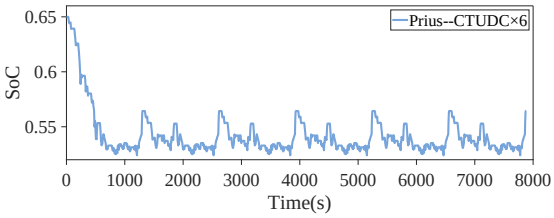


(b) Terminal SoC of the power-split HEB.

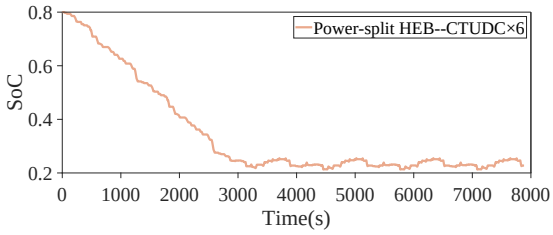


(c) Mean reward of the power-split HEB.

Fig. 8. Baseline of the plug-in HEB.



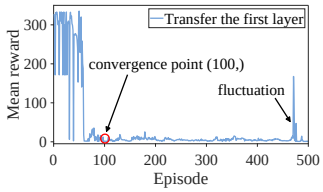
(a) SoC trajectory of Prius.



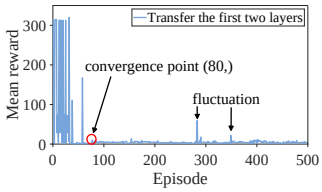
(b) SoC trajectory of the power-split HEB.

Fig. 9. SoC trajectories of the source and target domains.

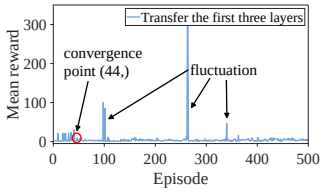
changeable, which means its mass is a stochastic variable. Hence, in order to simulate the real traffic environment, the changes in the number of passengers are added into driving cycle 1 to test the generalization performance to changing environments. In these experiments, the methods that transferring the first two and the first three layers are taken as examples to compare with the baseline. The simulation results in table IV show that the transferred EMS can adapt well to the changes in vehicle mass, which indicates that DTRL-based EMS is highly adaptive to stochastic environments.



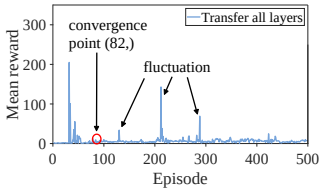
(a) Transfer the first layer.



(b) Transfer the first two layers.



(c) Transfer the first three layers.



(d) Transfer all layers.

Fig. 10. Transfer different layers.

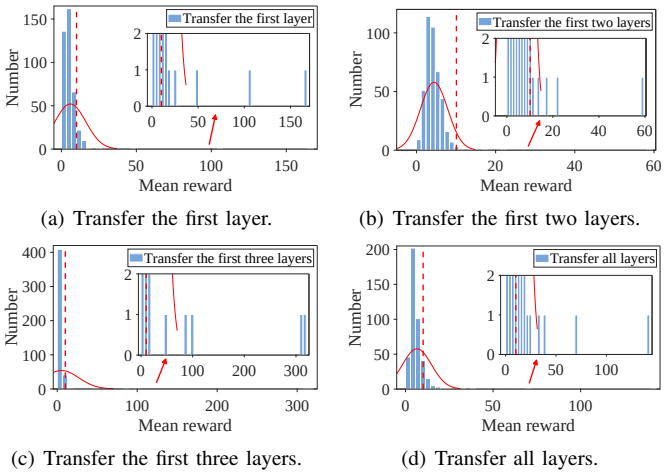
C. Transfer from Prius to the series HEV

To further explore the transferability of EMS in the other types of HEVs, we make a research on the TL from Prius to the series HEV. Compared with the task in section V-B, the differences between vehicle configurations in the source and target domains are larger. Similarly, the baseline of the series HEV is trained from scratch under the same target dataset, and the algorithm converges to a stable state from 200 episodes approximately, as shown in Fig. 12. Based on the mean reward of the baseline, the average μ and standard deviation σ are 0.50 and 0.12, and the upper and lower bounds of outliers calculated by equation (9) are 0.74 and 0.26, respectively. Unlike the EMS of Prius in Fig. 13, the series HEV operates in the charge-sustaining mode when SoC is lower than 0.1.

TABLE IV
PERFORMANCE EVALUATION OF TRANSFER

	Baseline	The first layer	The first two layers	The first three layers	All layers
Jumpstart (Chinese Yuan, CNY)	80.36	79.18	61.85	57.86	40.09
Number of outliers	7	37	5	7	29
Fuel economy (%)	80.7	81.2	80.5	80.0	80.2
Convergence efficiency(episodes)	220	100	80	44	82
Generalization performance(CNY)/	38.42 ± 0.62	38.63 ± 0.80	38.88 ± 0.88	38.54 ± 1.09	38.83 ± 1.00
$P_{fuel}(\%)$	81.8	81.3	80.8	81.4	80.9
Adaptation to changing environments(CNY)/	38.22 ± 0.37	N/A	38.60 ± 0.64	37.71 ± 0.71	N/A
$P_{fuel}(\%)$	82.3	N/A	81.4	83.2	N/A

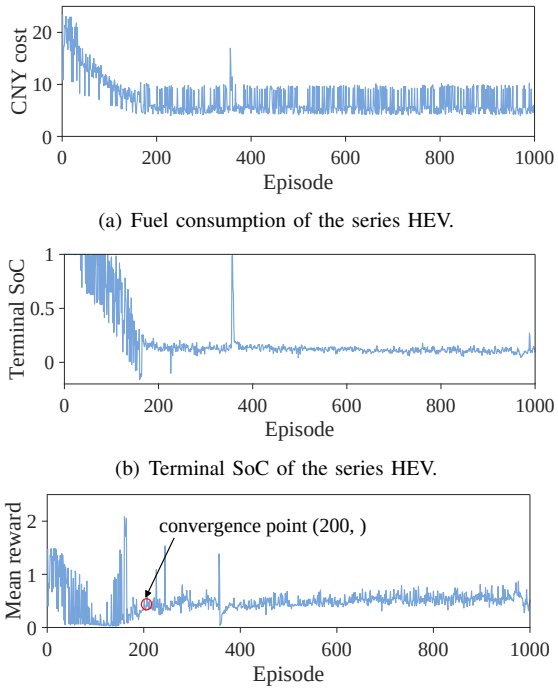
Fuel economy or P_{fuel}: energy consumption ratio of DP to the baseline or the transferred EMS.



(a) Transfer the first layer. (b) Transfer the first two layers.

(c) Transfer the first three layers. (d) Transfer all layers.

Fig. 11. Number of outliers.

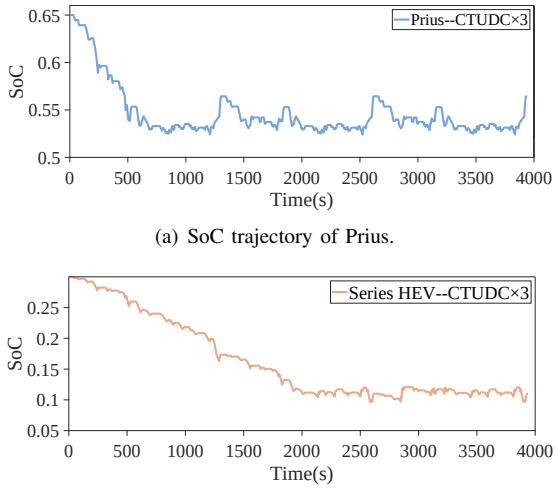


(a) Fuel consumption of the series HEV.

(b) Terminal SoC of the series HEV.

(c) Mean reward of the series HEV.

Fig. 12. Baseline of the series HEV.



(a) SoC trajectory of Prius.

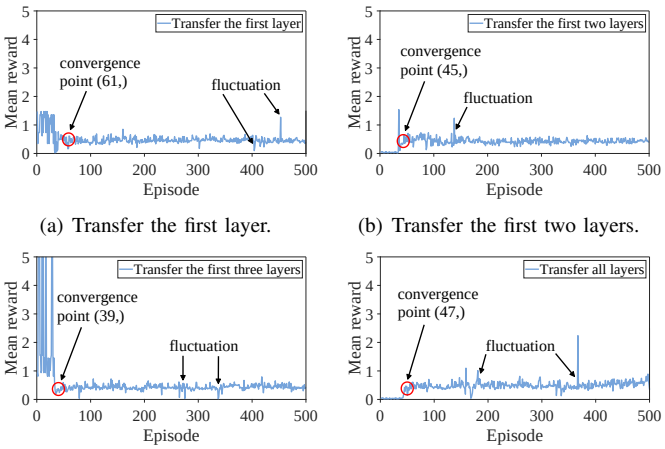
(b) SoC trajectory of the series HEV.

Fig. 13. SoC trajectories of the source and target domains.

The neural networks are fine-tuned by the same target dataset of the power-split HEB. The simulation results in Fig. 14 indicate that DTRL is a feasible solution to the EMS transfer between utterly different types of HEVs. No matter which layer is selected to transfer, all the neural networks using transferred knowledge improve the convergence efficiency, which are increased by 69.5%, 77.5%, 79.5% and 76.5% compared to the baseline. Similar to the statement in section V-B, the detailed results in Fig. 15 and table V further demonstrate that the methods transferring the first two layers or the first three layers show significant superiority in the convergence efficiency and robustness. It means that the developing cycle of a new EMS can be shortened by transferring knowledge from a different but well-developed HEV EMS.

Considering the generalization performance of fuel economy, DTRL-based EMS keeps the same level as the baseline under driving cycle 2. In real world, the initial battery SoC values of a HEV can be various, which greatly differs from the simulation settings. Hence, the generalization performance of the DTRL-based EMS with respect to different initial SoC values is also evaluated. In Fig. 16, it can be found that the transferred EMS is well-adapted to the cases with different

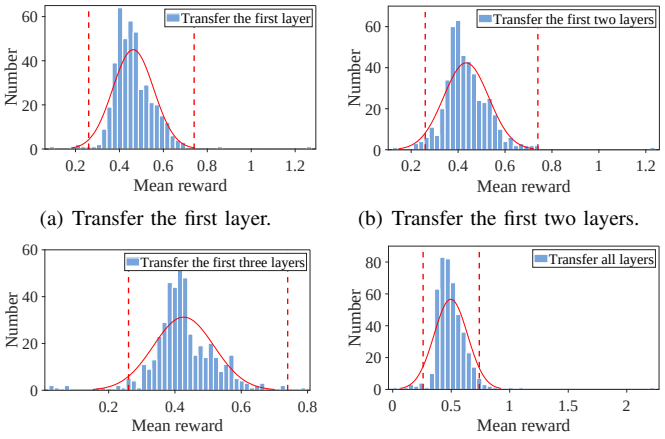
initial SoC values.



(a) Transfer the first layer. (b) Transfer the first two layers.

(c) Transfer the first three layers. (d) Transfer all layers.

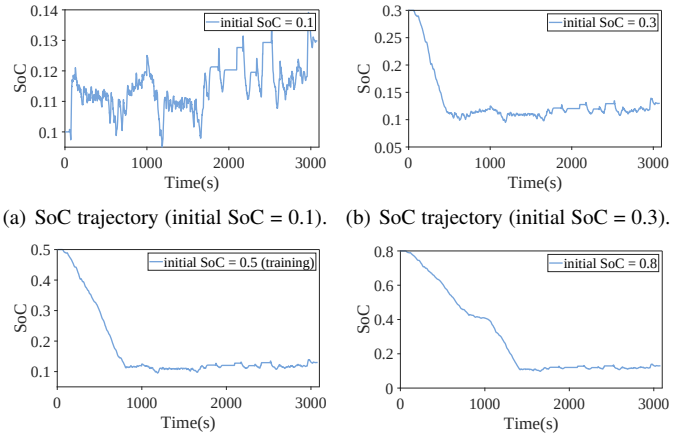
Fig. 14. Transfer different layers.



(a) Transfer the first layer. (b) Transfer the first two layers.

(c) Transfer the first three layers. (d) Transfer all layers.

Fig. 15. Number of outliers.



(a) SoC trajectory (initial SoC = 0.1). (b) SoC trajectory (initial SoC = 0.3).

(c) SoC trajectory (initial SoC = 0.5). (d) SoC trajectory (initial SoC = 0.8).

Fig. 16. Adaptability of the method transferring the first three layers to different initial SoC values.

D. Transfer from Prius to the series-parallel HEB

Unlike the above experiments that evaluate TL between tasks with the same action variables, we further explore the possible solution for TL between EMSs with different action variables.

Based on the above results, we transfer the internal representations of the first three layers from Prius to the series-parallel HEB. In this case, a knowledge transfer should be made from a two-dimensional action space (W_{eng}, T_{eng}) to a three-dimensional action space $(W_{eng}, T_{eng}, T_{mot})$. Due to the difference between the dimensions of action space, the input layers of critic networks in the source and target domains are different from each other, and therefore partial parameters of the input layer from Prius cannot be directly transferred to the EMS of the series-parallel HEB, as shown in Fig. 17.

Fig. 18 shows that the transferred EMS is approximately 40% faster than the baseline to achieve the same level of fuel economy. It can be found that the knowledge stored in the front neural network layers of Prius's EMS are beneficial to the EMS of the series-parallel HEB. Interestingly, this results indicate that transferring EMS knowledge between HEVs with different types of powertrains and control variables are beneficial. The essence of HEV EMS is to efficiently utilize fuel and electric energy. There exist some relations and common characteristics between different HEV EMSs on the control aim and powertrain. The neural networks in the source domain are able to effectively learn this common knowledge and store them in their neural network parameters as abstract knowledge. Transferring this abstract knowledge is helpful for a type-specific HEV EMS.

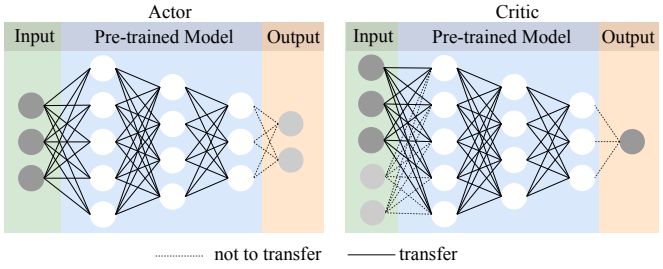


Fig. 17. Target actor-critic network.

E. Interpretability of EMS transfer

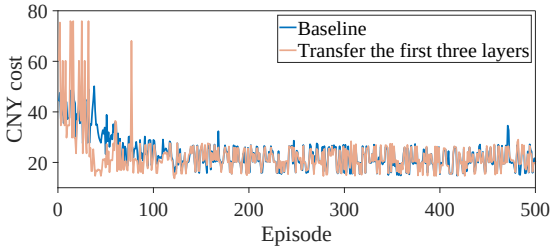
In the field of image processing, there has been a thorough understanding on the interpretability of neural network. A pioneering analysis is provided in [43]. Many deep neural networks trained on natural images exhibit a common phenomenon: the first-layers appear general in that they are applicable to many datasets and tasks. Representations will gradually transition from general to specific with the depth increase of neural network layers.

Similarly, in the field of DRL-based EMS, the driving cycles are also represented by a specific deep neural network. Based on this common knowledge, we assume that the general knowledge of EMSs is mostly stored in the first layer, whereas

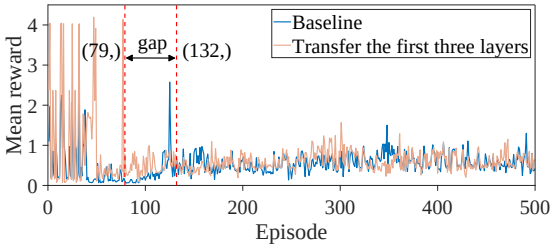
TABLE V
PERFORMANCE EVALUATION OF TRANSFER

	Baseline	The first layer	The first two layers	The first three layers	All layers
Jumpstart (CNY)	18.2	16.69	9.65	4.48	8.79
Number of outliers	15	5	9	11	21
Fuel economy (%)	98.3	98.4	98.6	98.4	98.4
Convergence efficiency(episodes)	200	61	45	39	47
Generalization performance(CNY)/	17.27 ± 0.26	17.33 ± 0.20	17.26 ± 0.23	17.36 ± 0.21	17.25 ± 0.26
$P_{fuel}(\%)$	95.3	95.0	95.7	95.1	95.0

Fuel economy or P_{fuel}: energy consumption ratio of DP to the baseline or the transferred EMS.

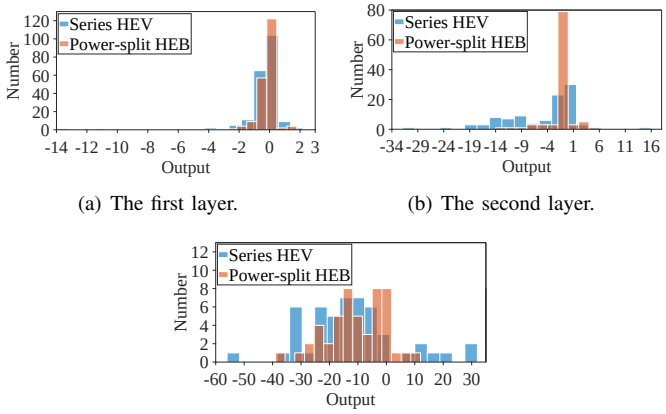


(a) Comparison of CNY cost.



(b) Comparison of mean reward.

Fig. 18. Comparison between the baseline and the transferred EMS.



(a) The first layer. (b) The second layer.

(c) The third layer.

Fig. 19. Output distributions of different layers without activation function.

TABLE VI
EUCLIDEAN DISTANCE OF OUTPUT BETWEEN SERIES HEV AND PLUG-IN HEB.

Output	Euclidean distance
Output of the first layer (1×200)	20.9
Output of the second layer (1×100)	82.7
Output of the third layer (1×50)	146.1
Action (1×2)	168.0

the specific knowledge of a particular HEV type is stored in the last layer. From the transfer from Prius to the power-split HEB or the series HEV, it can be found that only transfer the first layer is also beneficial for the rapid development of different types of HEV EMSs. In order to further explain this phenomenon, we analyze the output distributions of different layers between the series HEV and the power-split HEB. For comparison, the baselines of the two HEVs are utilized and tested with the same input states. The similarity degrees of outputs between the two HEVs are measured by Euclidean distance in equation (11).

From Fig. 19 and table VI, it can be seen that the similarity degrees of outputs decrease with the depth increase of neural network layers. In other words, the similarity degrees of neural network parameters decrease with the increasing pertinence of different layers to tasks. Hence, these general representations of driving cycles occur regardless of the specific type of HEV. This property of deep neural networks also makes the fact that the method transferring all layers performs worse in terms of robustness.

VI. CONCLUSION AND FUTURE WORK

To the best of our knowledge, this is the first paper to propose transfer learning (TL) for energy management of different types of hybrid electric vehicles (HEVs). We relate the principle that different HEVs share certain commonalities which can be utilized by the deep reinforcement learning (DRL) based energy management systems. We show how the developing cycle can be shortened by transferring knowledge between different energy management systems. With the transferred knowledge, the generalization performance with respect to the changes in vehicle parameters is improved, and the fuel economy is maintained.

We also investigate knowledge transfer between two vehicles with significantly different types of powertrains, in which the control variables are different from each other. Surprisingly, improved results are also reported.

Our hypothesis is that deep neural network is able to learn the common knowledge for the optimal assignment between fuel and electric power sources. Sharing this knowledge stored in its internal representations to the other energy management will be beneficial to the other types of energy management systems.

However, in this research, the knowledge transfer between different HEVs must be conducted under the same neural network architecture, i.e. the same number of neurons and hidden layers. In our future work, the transfer learning between heterogeneous network architectures will be explored, so as to make better use of general knowledge. Our final goal is to develop a general energy management system for any types of HEVs. To achieve this, we will explore the applications of more advanced machine learning techniques.

ACKNOWLEDGMENT

This work was supported by IVADO, and National Natural Science Foundation of China [Grant No.51705020 & No.61620106002]. Any opinions expressed in this paper are solely those of the authors and do not represent those of the sponsors. The authors would like to thank the reviewers for their corrections and helpful suggestions. The authors also thank Yong Wang from Beijing Institute of Technology and Wenchang Li from Xiamen university for data collection.

REFERENCES

[1] W. Enang and C. Bannister, "Modelling and control of hybrid electric vehicles (a comprehensive review)," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 1210–1239, 2017.

[2] C. M. Martinez, X. Hu, D. Cao, E. Velenis, B. Gao, and M. Wellers, "Energy management in plug-in hybrid electric vehicles: Recent progress and a connected vehicles perspective," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4534–4549, 2016.

[3] M. Sabri, K. Danapalasingam, and M. Rahmat, "A review on hybrid electric vehicles architecture and energy management strategies," *Renewable and Sustainable Energy Reviews*, vol. 53, pp. 1433–1442, 2016.

[4] P. Zhang, F. Yan, and C. Du, "A comprehensive analysis of energy management strategies for hybrid electric vehicles based on bibliometrics," *Renewable and Sustainable Energy Reviews*, vol. 48, pp. 88–104, 2015.

[5] N. Rotering and M. Ilic, "Optimal charge control of plug-in hybrid electric vehicles in deregulated electricity markets," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1021–1029, 2010.

[6] Y. Li, H. He, J. Peng, and H. Wang, "Deep reinforcement learning-based energy management for a series hybrid electric vehicle enabled by history cumulative trip information," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7416–7430, 2019.

[7] J. Peng, H. He, and R. Xiong, "Rule based energy management strategy for a series–parallel plug-in hybrid electric bus optimized by dynamic programming," *Applied Energy*, vol. 185, pp. 1633–1643, 2017.

[8] A. M. Ali and D. Söffker, "Towards optimal power management of hybrid electric vehicles in real-time: A review on methods, challenges, and state-of-the-art solutions," *Energies*, vol. 11, no. 3, p. 476, 2018.

[9] T. Hofman, M. Steinbuch, R. Van Druten, and A. Serrarens, "Rule-based energy management strategies for hybrid vehicles," *International Journal of Electric and Hybrid Vehicles*, vol. 1, no. 1, pp. 71–94, 2007.

[10] J. Guo, H. He, and C. Sun, "Arima-based road gradient and vehicle velocity prediction for hybrid electric vehicle energy management," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5309–5320, 2019.

[11] F. Zhang, X. Hu, R. Langari, and D. Cao, "Energy management strategies of connected hevs and phevs: Recent progress and outlook," *Progress in Energy and Combustion Science*, vol. 73, pp. 235–256, 2019.

[12] J. Wu, H. He, J. Peng, Y. Li, and Z. Li, "Continuous reinforcement learning of energy management with deep q network for a power split hybrid electric bus," *Applied energy*, vol. 222, pp. 799–811, 2018.

[13] H. Tan, H. Zhang, J. Peng, Z. Jiang, and Y. Wu, "Energy management of hybrid electric bus based on deep reinforcement learning in continuous state and action space," *Energy Conversion and Management*, vol. 195, pp. 548–560, 2019.

[14] Y. Wu, H. Tan, J. Peng, H. Zhang, and H. He, "Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus," *Applied Energy*, vol. 247, pp. 454–466, 2019.

[15] G. Du, Y. Zou, X. Zhang, Z. Kong, J. Wu, and D. He, "Intelligent energy management for hybrid electric tracked vehicles using online reinforcement learning," *Applied Energy*, vol. 251, p. 113388, 2019.

[16] L. Li, X. Wang, and J. Song, "Fuel consumption optimization for smart hybrid electric vehicle during a car-following process," *Mechanical Systems and Signal Processing*, vol. 87, pp. 17–29, 2017.

[17] Z. Chen, B. Xia, C. You, and C. C. Mi, "A novel energy management method for series plug-in hybrid electric vehicles," *Applied Energy*, vol. 145, pp. 172–179, 2015.

[18] C. Xiang, F. Ding, W. Wang, and W. He, "Energy management of a dual-mode power-split hybrid electric vehicle based on velocity prediction and nonlinear model predictive control," *Applied energy*, vol. 189, pp. 640–653, 2017.

[19] J. S. Martinez, R. I. John, D. Hissel, and M.-C. Péra, "A survey-based type-2 fuzzy logic system for energy management in hybrid electrical vehicles," *Information Sciences*, vol. 190, pp. 192–207, 2012.

[20] A. Lazaric, "Transfer in reinforcement learning: a framework and a survey," in *Reinforcement Learning*. Springer, 2012, pp. 143–173.

[21] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International Conference on Artificial Neural Networks*. Springer, 2018, pp. 270–279.

[22] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1717–1724.

[23] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[24] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[25] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[26] J. Rajendran, P. Prasanna, B. Ravindran, and M. M. Khapra, "Adaapt: A deep architecture for adaptive policy transfer from multiple sources," *arXiv preprint arXiv*, vol. 1510, 2015.

[27] T. Chen, A. Murali, and A. Gupta, "Hardware conditioned policies for multi-robot transfer learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 9333–9344.

[28] E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Actor-mimic: Deep multitask and transfer reinforcement learning," *arXiv preprint arXiv:1511.06342*, 2015.

[29] P. Ammanabrolu and M. O. Riedl, "Transfer in deep reinforcement learning using knowledge graphs," *arXiv preprint arXiv:1908.06556*, 2019.

[30] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[31] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, p. 9, 2016.

[32] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[33] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[34] R. H. Staunton, C. W. Ayers, L. Marlino, J. Chiasson, and B. Burress, "Evaluation of 2004 toyota prius hybrid electric drive system," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), Tech. Rep., 2006.

[35] F. Millo, L. Rolando, and M. Andreata, "Numerical simulation for vehicle powertrain development," in *Numerical Analysis-Theory and Application*. IntechOpen, 2011.

[36] S. Onori, L. Serrao, and G. Rizzoni, *Hybrid electric vehicles: Energy management strategies*. Springer, 2016.

[37] Z. Yang, K. E. Merrick, H. A. Abbass, and L. Jin, "Multi-task deep reinforcement learning for continuous action control." in *IJCAI*, 2017, pp. 3301–3307.

[38] D. Bahdanau, P. Brakel, K. Xu, A. Goyal, R. Lowe, J. Pineau, A. Courville, and Y. Bengio, "An actor-critic algorithm for sequence prediction," *arXiv preprint arXiv:1607.07086*, 2016.
[39] R. Lian, J. Peng, Y. Wu, H. Tan, and H. Zhang, "Rule-interposing deep reinforcement learning based energy management strategy for power-split hybrid electric vehicle," *Energy*, vol. 197, p. 117297, 2020.
[40] R. Islam, P. Henderson, M. Gomrokchi, and D. Precup, "Reproducibility of benchmarked deep reinforcement learning tasks for continuous control," *arXiv preprint arXiv:1708.04133*, 2017.
[41] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *Journal of machine learning research*, vol. 10, no. Jan, pp. 1–40, 2009.
[42] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.
[43] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.
[44] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1633–1685, 2009.

Renzong Lian received the B.S. degree in vehicle engineering from Fuzhou University, Fuzhou, China, in 2017. He is pursuing the M.S. degree in the School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China. His current research interests include the machine learning and energy management of the hybrid electric vehicles.

Huachun Tan (M'13) received the Ph.D. degree in electrical engineering from Tsinghua University, Beijing, China, in 2006. He is with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing from Sep. 2009 to June 2018. He is now a Professor with the School of Transportation, Southeast University, Nanjing, China. He has published 100 papers, his research interests include image engineering, pattern recognition, and intelligent transportation systems.

Jiankun Peng received the Ph.D. degree in mechanical engineering from Beijing Institute of Technology, Beijing, China, in 2016. He is currently a Post Doctorate with the National Engineering Laboratory for Electric Vehicles, Beijing Institute of Technology, Beijing, China. He has published more than 50 papers, his research interests include energy management and optimization for electrified vehicles, connected and automated vehicle-highway system, as well as optimal decision making for vehicle chassis X-by-wire.

Qin Li received the Bachelor's degree from the Department of Transportation Engineering, Beijing Institute of Technology, Beijing, China, in 2014. She is pursuing the Ph.D. degree with the School of Mechanical Engineering, Beijing Institute of Technology, and was a visiting student with Department of Civil & Environmental Engineering, University of Wisconsin-Madison from Oct. 2017 to Oct, 2018. Her research interests include machine learning and intelligent transportation systems.

Yuankai Wu received the PhD's degree from the School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China, in 2019. He was a visit PhD student with Department of Civil & Environmental Engineering, University of Wisconsin-Madison from Nov. 2016 to Nov, 2017. He is a Postdoc researcher with Department of Civil Engineering and Applied Mechanics of McGill University, supported by Institute For Data Valorization (IVADO). His research interests include intelligent transportation systems, intelligent energy management and machine learning.