Cooperative Power Management for Range Extended Electric Vehicle Based on Internet of Vehicles

Yuanjian Zhang[1], Bingzhao Gao[2], Jingjing Jiang[1], Chengyuan Liu[1], Dezong Zhao[3], Quan Zhou[4], Zheng Chen[5*] and Zhenzhen Lei[6**]

[1]Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough LE11 3TU, UK.
[2]School of Automotive Studies, Tongji University, Shanghai, 201804, China.
[3]James Watt School of Engineering, University of Glasgow, G12 8QQ, UK.
[4]Department of Mechanical Engineering, University of Birmingham, B15 2TT, UK
[5]Faculty of Transportation Engineering, Kunming University of Science and Technology, Kunming, 650500, China
[6]School of Mechanical and Power Engineering, Chongqing University of Science & Technology, Chongqing, 401331, China.
Corresponding Author: Zheng Chen (chen@kust.edu.cn) and Zhenzhen Lei (2010048@cqust.edu.cn)

Abstract: The dramatic progress in internet of vehicles (IoVs) inspires further development in electrified transportation, and abundant information exchanged in IoVs can be infused into vehicles to promote the controlling performance of electric vehicles (EVs) via vehicle-environment cooperation. In this paper, a cooperative power management strategy (PMS) is advanced for the range extended electric vehicle (REEV). To this end, the studied REEV is accurately modelled first, laying an efficient platform for strategy design. Based on the advanced framework of IoVs, the cooperative PMS is meticulously developed via incorporating the self-learning explicit equivalent minimization consumption strategy (SL-eECMS) and adaptive neuro-fuzzy inference system (ANFIS) based online charging management within on-board power sources in the REEV. The brand-new SL-eECMS achieves preferable balance between the optimal effect and instant implementation capability through integrating the improved quantum particle swarm optimization (iQPSO), and ANFIS grasps future driving status macroscopically, offering the predicted charging request for online charge management. The substantial simulations and hardware-in-the-loop (HIL) test manifest that the proposed cooperative PSMS can coherently and efficiently manage power flow within power sources in the REEV, highlighting its anticipated preferable performance.

Key words: Power management strategy (PMS), self-learning explicit equivalent minimization consumption strategy (SL-eECMS), adaptive neuro-fuzzy inference system (ANFIS), improved quantum particle swarm optimization (iQPSO), range extended electric vehicle (REEV).

NOMENCLATURE

Abbreviations

EV	electric vehicles	IoVs	internet of vehicles
PMS	power management strategy	REEV	range extended electric vehicle
ANFIS	adaptive neuro-fuzzy inference system	SL-eECMS	self-learning explicit equivalent minimization consumption strategy
PSO	particle swarm optimization	QPSO	quantum particle swarm optimization
iQPSO	improved quantum particle swarm optimization	EMS	energy management strategies
DP	dynamic programming	PMP	Pontryagin minimum principle
ECMS	equivalent consumption minimization strategy	MPC	model predictive control
EF	equivalent factor	MEC	Mobile edge computation
MECU	mobile edge computation units	VCU	vehicle control units
V2V	vehicle to vehicle	V2I	vehicle to infrastructure
ZE	zero emission	PE	pure electric
ND	normal driving	HD	hybrid driving
ED-LUT	energy distribution look-up table	APU	auxiliary power unit
CS	charging sustaining	PMSM	permanent magnet synchronous motors
SOC	state of charge	I2I	infrastructure to infrastructure
mmWave	millimeter-wave	BS	base stations
RF	radio frequency	CM-MECU	charging management MECU
ZE-MECU	ZE MECU	N-MECU	normal MECU
BSFC	brake-special fuel consumption	VP	Velocity profile

Symbols

P_{req}	required tractive power	c_2	acceleration coefficient
G	gravity	G_n	global best position
α	gradient	$f(x)$	fitness function
f	rolling resistance factor	S	feasible space
C_d	aerodynamic drag factor	$p_{i,n}$	local attractor
A	frontal area	$L_{i,n}$	characteristic length of potential well
v	vehicle speed	$u_{i,n+1}^{j}$	sequence of random numbers
a	acceleration	τ	contraction-expansion coefficient
ξ	correction coefficient of rotating mass	C_n^j	mean best position
η_t	transmission efficiency	ε	predefined ratio
m	vehicle mass	$\aleph$	Gaussian distribution
T_{em}	electric motor torque	X	optimized parameter
i_{fd}	final drive ratio	$P_{batt\_\min\_nor}$	minimum battery power
R_{wh}	wheel radius	$P_{batt\_\max\_nor}$	maximum battery power
η_{eng}	engine net efficiency	A_1	nonlinear parameter
ω_{eng}	rotating speed of engine	A_2	nonlinear parameter
Q_{lhv}	fuel lower heating value	B_1	nonlinear parameter
$\dot{m}_f$	fuel consumption rate	B_2	nonlinear parameter
ω_{em}	angular speed of electric motor	p_1	linear parameter
P_{em}	mechanical power of electric motor	p_2	linear parameter

η_{mot}	efficiency of electric motor in tractive mode	q_1	linear parameter
η_{gen}	efficiency of electric motor in generator mode	q_2	linear parameter
SOC	battery state of charge	r_1	linear parameter
R_{int}	internal resistance of battery	r_2	linear parameter
P_{batt}	battery power	I_1	mentioned input
Q_{batt}	battery capacity in Ah	I_2	mentioned input
P_{eng_de}	distributed optimal engine power	μ_{A_j}	membership function
$\dot{m}_{f,eqv}$	total instant equivalent fuel consumption	O_j^1	output of a layer
$\dot{m}_{ress}$	equivalent fuel consumption rate that converted from electricity usage	a_j	premise parameter
t	control step	b_j	premise parameter
P_{eng}	engine power	c_j	premise parameter
s	EF	p_j	consequent parameter
u^*	optimal control variable	q_j	consequent parameter
P	probability distribution	r_j	consequent parameter
Q	probability distribution	T_s	traffic status
M	metric space	V_s	vehicle status
p	order's Wasserstein distance	V_{ave}	average velocity
$\rho(x,y)$	distance function	A_{ave}	acceleration
x	sample	$T_{v=0}$	vehicle stop time
y	sample	v_{ave}	predicted average velocity
Θ	set	a_{ave}	acceleration of the target vehicle
$\prod(P,Q)$	probability measures	$\tau_{v=0}$	predicted stop time of the target vehicle
$\|f\|$	Lipschitz semi-norm	t_0	predicted initial time
$X_{i,n}$	position	t_d	predicted terminal time
$V_{i,n}$	velocity	E_{req_ze}	energy in ZE zone
i	particle	SOC_{ini_ze}	initial battery SOC in ZE zone
n	iteration	SOC_i	instant battery SOC in Approach zone
j	dimension space	E_{req_ap}	required energy for left driving in Approach zone
c_1	acceleration coefficient	W_b	battery capacity

I. INTRODUCTION

The aggravation of energy dilemma, environment pollution and global warming appeals technique innovation in all disciplines towards net-zero emission of transport [1-4]. As a major contributor to harmful gas emission and excessive fossil fuel consumption, public transportation bears the brunt of electrification, spurring the blooming of electric transportation solutions in recent years [5, 6]. Among these granted electrified vehicles (EVs), range

extended EV (REEV) represents one promising solution [6]. Equipped with multiple power sources, REEVs hold tremendous advantages in energy reservation and emission reduction without excessively scarifying driving mileage. The additional degree of freedom in energy consumption, counteractively, leads to the imperative requirement on power management strategies (PMSs) for optimally governing energy flow within hybrid powertrains of REEVs.

The studies on PMSs, also referred to as energy management strategies (EMSs), for hybrid powertrains have been lasting for tens of years, and a substantial of solutions have been advanced. Existing PMSs for hybrid powertrains can be divided into rule based ones (such as rule-based strategies [7] and fuzzy logic based methods [8]), global optimization-based PSMSs by employing algorithms like dynamic programming (DP) [9] and Pontryagin minimum principle (PMP) [10], instantaneous optimization-based PMSs (such as equivalent consumption minimization strategy (ECMS) [11] and model predictive control (MPC) [12]), as well as machine learning and deep learning based PMSs (like Q-learning [13] and deep Q-learning [14]). Despite the validated performance under certain conditions, existing solutions still hold some shortcomings, impeding their application performance promotion. In particular, the real-time application performance of some listed methods still needs to be improved. Global optimization-based PMSs seek the optimal solutions by traversing all candidate solutions within whole cycles, thus obviously increasing the cost of computation and the difficulties in real-time implementations. Rule-based methods rely on engineering practice largely and are difficult to adaptively cope with complex environments. In addition, the application effect of the presented methods in real time is restricted by the greedy dependence on information of future driving. Adaptive tuning of some inner parameters in core algorithms, e.g. equivalent factor (EF) in ECMS, demands the pre-knowledge of future driving, making it intractable in practical applications.

Among the discussed methods, ECMS possesses huge potential in real-time implementations with preferable control effect that is quite close to DP or PMP [15]. During the implementation of ECMS, the algorithm instantly searches the power distribution combination that contributes to the minimum equivalent fuel consumption (including the original fuel consumption and the equivalent fuel consumption converted from the consumed electric energy). The conversion from electric energy to equivalent fuel consumption is attained by employing EF [16], which is the representative index denoting efficiencies of hybrid powertrain in past, current and future driving

[16]. Therefore, prediction of future driving conditions is often incorporated into the implementation of ECMS [17], while huge computation burden is often imposed on the vehicle on-board hardware, thus discounting the real-time implementation performance of ECMS.

The adopted methods to distribute energy within power sources in hybrid powertrains, generally, can behave optimally with abundant future driving information and huge computation, appealing novel solutions proposed to furnish performance in real time. With the dramatically development of internet of vehicles (IoVs) [18], vehicle-environment cooperation provides new perspectives to prompt on-board power management [19]. The flexible information interaction schemes lower the difficulties of accurately acquiring future driving status [20], and advanced cooperation frameworks enable to share partial computation through vehicle's on-board hardware [21]. Mobile edge computation (MEC) is one of progressive asynchronous computation theories that encourages the appearance of novel vehicle-environment cooperation methods [22]. In the MEC based framework, mobile edge computation units (MECUs) at road side offload partial tasks from on-board vehicle control units (VCUs), and the equipped superior vehicle to vehicle (V2V) communication and vehicle to infrastructure (V2I) communication enrich the valuable environment information that can be accessed from vehicle side [23]. The MEC based vehicle-environment cooperation has been applied in charging management [24]. Even though there are some initial attempts of vehicle-environment cooperation by MEC, the MEC based applications in electric transportation for efficient driving, to the best of the authors' knowledge, are still rare.

Despite the technique innovation, new social-economic policies are made to accelerate the transition to net-zero emission [25, 26]. In some urban traffic networks, zero emission (ZE) zones are prescribed to reduce harmful exhaust gas emission [27]. To be specific, vehicles must operate in pure electric (PE) mode in ZE zones to reduce emission and noise due to some specific restrictions. On this account, the operation of EVs including REEV, can be divided into normal driving (ND) zone, approach zone and ZE zone when operating in urban traffic networks. The divided zones on operation route are illustrated in Fig. 1, and in the ND zone, REEV operates in normal state by cooperatively utilizing PE mode and hybrid driving (HD) mode. After REEV moves into approach zone, forced charge is implemented to replenish electricity for future driving in ZE zone. This compulsory operation guarantees sufficient energy saving of battery in ZE zone. In ZE zone, only PE mode is enabled, inducing engines in REEVs

to be turned off and electricity to be consumed. After vehicles pulling out ZE zone, normal operation state is activated again.

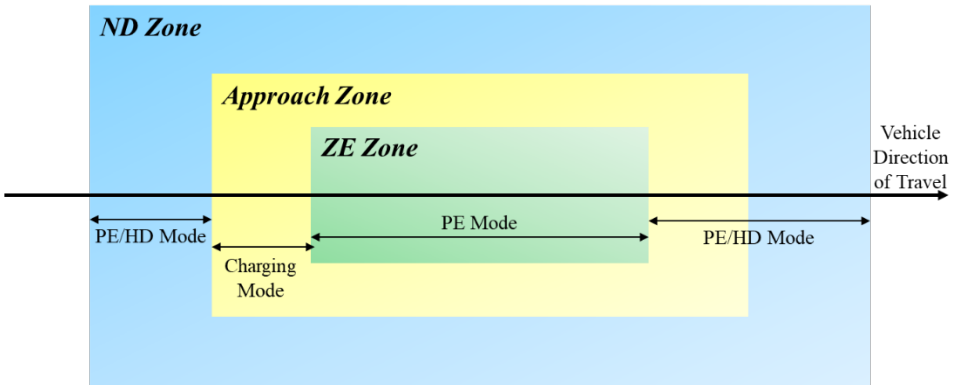


Fig. 1. Illustration on different operation zones in urban traffic network.

To move forward the state-of-the-art research in the literature, a novel PMS is presented in this study based on the advanced MEC framework in IoVs. In this cooperative PMS, self-learning explicit ECMS (SL-eECMS) accomplishes energy management with high quality and within the power sources in hybrid powertrain. The explicit control policies, denoted as energy distribution look-up table (ED-LUT) which is generated offline with optimality, are implemented into online control efficiently with better adaptabilities to driving environment. The adaptive neuro-fuzzy inference system (ANFIS), in parallel, predicts the charging levels properly, thereby attaining effective online charging management for future driving in ZE zones. The following four contributions developed by this study are added to current literature:

1. A novel control framework in IoVs is constructed to solidly support design and implementation of the raised cooperative PSMS.
2. The self-learning mechanism is incorporated into the novel SL-eECMS according to the Wassertern distance-based learning method. The explicit control policies for online implementation are updated, given the driving conditions judged by the Wassertern distance, to prompt the adaptabilities of the novel strategy to various driving environments.
3. The improved quantum particle swarm optimization (iQPSO) is integrated into the SL-eECMS to offline generate and online update ED-LUTs, strengthening optimality of SL-eECMS in real-time application.
4. The online charge management is realized with the support of ANFIS. By capturing traffic information shared through IoVs, ANFIS predicts charging levels corresponding to the forced pure electric driving in future, guaranteeing that sufficient electric energy can be prepared in advance.

The remainder of this paper is organized as follows. The studied REEV and the related model building are described in Section II. The designed cooperative PSMS is detailed in Section III, and Section IV evaluates the simulation and hardware-in-the-loop (HIL) test results and validates the performance of the raised method. The main conclusions are made in Section VI.

II. REEV AND MODEL CONSTRUCTION

2.1 The Studied REEV

REEVs, as the anagenesis of traditional HEVs, demonstrate better economy improvement, less emission reduction and longer driving mileage. The affirmed performance enhancement is obtained through equipping larger battery pack and better optimized ICE, as well as employing more advanced PMS. The studied REEV configuration is applied in a commercial public bus, and the configuration is shown in Fig. 2, where we can find that the electric motor itself outputs the tractive forces to satisfy various driving requirement. The battery pack, together with the auxiliary power unit (APU), provides the demanded tractive energy. The APU consists of an ICE and a generator, performing as the primary power source in charging sustaining (CS) stage. The ICE is connected to the generator through rigid coupling mechanism. In braking mode, the battery pack absorbs the regenerative braking energy recycled from electric motor. The detailed parameters of different components in REEV are given in Table 1. Note that the allowable range of battery SOC is from 0.2 to 0.8.

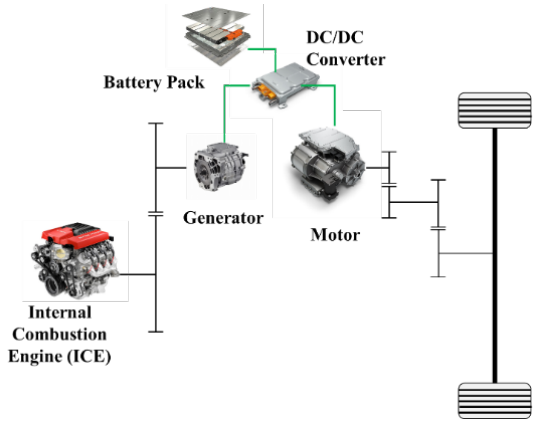


Fig. 2. The configuration of the studied REEV.

Table 1. Detailed parameters of each components in powertrain

Diesel Engine	Displacement	3.7 L
	Maximum Power	125 kW
	Maximum Torque	650 Nm
Motor/ Generator	Maximum Power	170 kW
	Maximum Torque	2800Nm

	Maximum Speed	3900 rpm
Battery	Type	Lithium-ion
	Capacity	92 kWh
	Nominal Voltage	645 V

2.2 REEV Model Construction

a. Vehicle Dynamics

The tractive power, generated based on the driving intention from driver, is outputted to overcome the driving resistance. The power balance at the output end of hybrid powertrains in REEV can be formulated, as:

$$P_{req} = \frac{v}{\eta_t}\left(\frac{Gfcos\alpha}{3600} + \frac{Gsin\alpha}{3600} + \frac{C_D A v^2}{76140} + \frac{\xi ma}{3600}\right) \quad (1)$$

where P_{req} denotes the required tractive power; G, α, f represents the gravity, gradient and rolling resistance factor; C_d, A, v is the aerodynamic drag factor, frontal area and vehicle speed, respectively; a means the acceleration; ξ, η_t, m is the correction coefficient of rotating mass, transmission efficiency, and vehicle mass, respectively. The transmission efficiency η_t considers the total loss of mechanical and electric path in the powertrain. At the wheel side of REEV, the corresponding torque balance can be expressed as:

$$F_t = \frac{T_{em} i_{fd} \eta_t}{R_{wh}} \quad (2)$$

where T_{em} is the electric motor torque, i_{fd} is the final drive ratio, and R_{wh} is the wheel radius.

b. Engine Model

Since the research focus in this study is to improve energy economy of REEV by optimal design on PSMS, the dynamic performance of engine is neglected. A static engine mode, including an efficiency map obtained through a benchmark test, is employed. Based on the static model, the engine efficiency can be described as:

$$\eta_{eng}\left(T_{eng}, n_{eng}\right) = \frac{T_{eng}\omega_{eng}}{Q_{lhv}\dot{m}_f} \quad (3)$$

where η_{eng} means the engine net efficiency, ω_{eng} represents the rotating speed of engine, Q_{lhv} denotes the fuel lower heating value, and $\dot{m}_f$ is the fuel consumption rate.

c. Motor/ generator model

Both the electric motor and generator in this study belong to permanent magnet synchronous motors (PMSMs). Likewise, the dynamic behaviors of motors are neglected, due to the optimization target in this paper. For the PMSM with tractive and generator mode, the relationship between torque and power can be described as:

$$P_{em} = \begin{cases} \frac{T_{em}\omega_{em}}{\eta_{mot}} & T_{em} > 0 \\ T_{em}\omega_{em}\eta_{gen} & T_{em} \leq 0 \end{cases} \tag{4}$$

where ω_{em} means the angular speed of electric motor; P_{em} denotes the mechanical power of electric motor; η_{mot} and η_{gen} represents the efficiency of electric motor in tractive mode and generator mode, respectively.

d. Battery model

For the battery, the performance of temperature and aging effect is ignored after carefully considering the optimization target and modelling complexity. A simplified equivalent circuit model, consisting of an open circuit voltage source, and an internal resistor connecting in series topology, is employed to characterize the battery electrical performance. The general relationship in the simple equivalent circuit model can be described as:

$$\dot{SOC} = -\frac{V_{oc} - \sqrt{V_{oc} - 4R_{int}P_{batt}}}{2R_{int}Q_{batt}} \tag{5}$$

where SOC means the battery state of charge (SOC), V_{oc} denotes the open circuit voltage of battery, R_{int} expresses the internal resistance of battery, P_{batt} represents the battery power, and Q_{batt} is the battery capacity in Ah.

III. Novel Cooperative Power Source Management Strategy

3.1 Novel Control Framework in IoVs

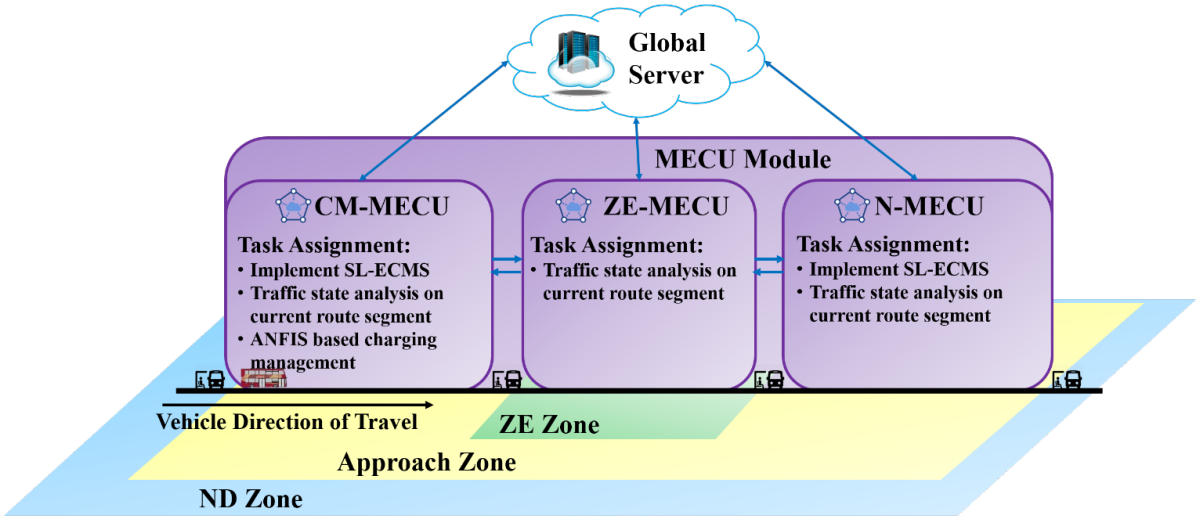
To develop the cooperative PMS, a special control framework is designed first, as shown in Fig. 3 (a). In this designed novel control framework, the on-board VCU in the REEV, together with MECUs at route side and global sever at cloud, accomplishes the cooperative management. The V2I, V2V and infrastructure to infrastructure (I2I) communication among on-board VCU, MECUs and global server is attained by the millimeter-wave (mmWave) communication with its ultra-wide band [28]. Besides, the base stations (BSs) and vehicle are assumed to be equipped with single radio frequency (RF) chain to reduce hardware complexity [29]. In daily applications, the roles of the mentioned control units can be summarized, as:

Global server: Global server unitedly activates MECUs that undertake tasks in cooperative control after subscribing the published route information from REEV.

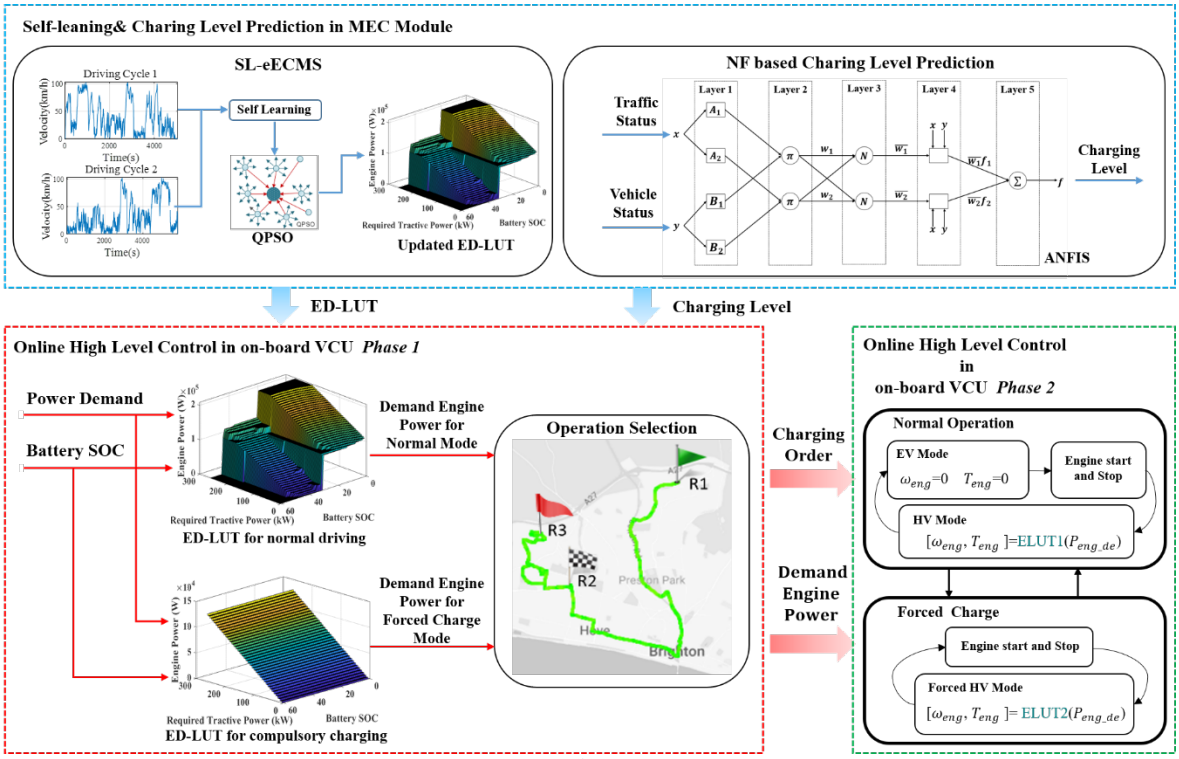
MECUs: MECUs, as the coordinative controllers, accomplish tasks related to vehicle-environment cooperation. Specifically, one MEC module includes three MECUs. Along with the vehicle travel direction, there are charging management MECU (CM-MECU), ZE MECU (ZE-MECU) and normal MECU (N-MECU). In CM-MECU, self-learning process in SL-eECMS based energy management and charging level prediction in online management are implemented by referring to the information from traffic state analysis. In ZE-MECU, only traffic state analysis is assigned. In N-MECU, self-learning process in SL-eECMS and traffic state analysis are delegated. The traffic state analysis in three MECUs focuses on different aspects. Traffic state analysis in CM-MECU collects and processes valuable information, and obtains the predicted velocity profile of the REEV on current route segment for self-learning in SL-eECMS. Nonetheless, ZE-MECU performs traffic state analysis and shares the evaluation results, including the predicted macroscopic traffic state, to CM-MECU to support partial charging management. In N-MECU, traffic state analysis, similar with that in CM-MECU, is required by SL-eECMS for the included self-learning process. Moreover, CM-MECU, ZE-MECU and N-MECU may consists of several sub-MECUs due to the limited serving area of each MECU.

On-board VCU: The main task of on-board VCU is to accomplish SL-eECMS based energy management and online charging management within power sources after subscribing information published from MECUs, which include the self-learning results for SL-eECMS and predicted charging level for future driving in ZE zone.

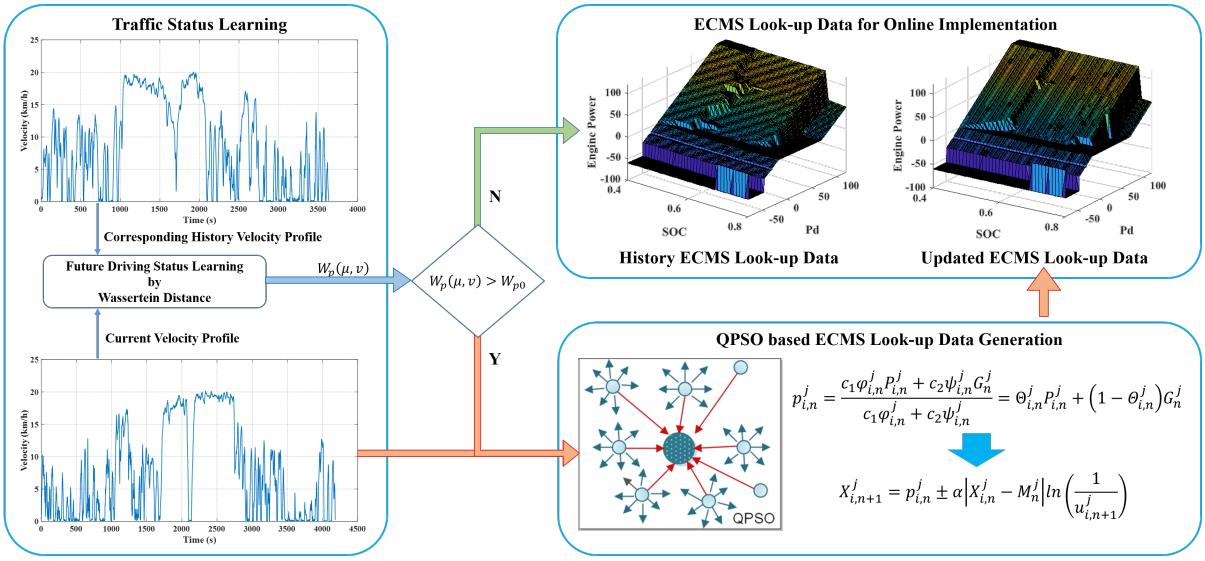
The presented MEC based cooperative control framework exhausts the powerful sensing and computing abilities of IoV, successfully allevitating computation load of vehicle and boosting vehicle-environment cooperation.


Global Server
MECU Module
CM-MECU
Task Assignment:
• Implement SL-ECMS
• Traffic state analysis on current route segment
• ANFIS based charging management
ZE-MECU
Task Assignment:
• Traffic state analysis on current route segment
N-MECU
Task Assignment:
• Implement SL-ECMS
• Traffic state analysis on current route segment
Vehicle Direction of Travel
ZE Zone
Approach Zone
ND Zone

(a)


Self-leaning& Charing Level Prediction in MEC Module
SL-eECMS
Driving Cycle 1
Driving Cycle 2
Velocity(km/h)
Time(s)
Self Learning
QPSO
Required Tractive Power (kW)
Battery SOC
Engine Power (W)
Updated ED-LUT
NF based Charing Level Prediction
Traffic Status
Vehicle Status
Layer 1
Layer 2
Layer 3
Layer 4
Layer 5
Charging Level
ANFIS
ED-LUT
Charging Level
Online High Level Control in on-board VCU Phase 1
Power Demand
Battery SOC
Demand Engine Power for Normal Mode
ED-LUT for normal driving
Demand Engine Power for Forced Charge Mode
ED-LUT for compulsory charging
Operation Selection
R1
R2
R3
Charging Order
Demand Engine Power
Online High Level Control in on-board VCU Phase 2
Normal Operation
EV Mode
ω_{eng}=0 T_{eng}=0
Engine start and Stop
HV Mode
$[\omega_{eng}, T_{eng}]$=ELUT1(P_{eng_de})
Forced Charge
Engine start and Stop
Forced HV Mode
$[\omega_{eng}, T_{eng}]$= ELUT2(P_{eng_de})

(b)



(c)

Fig. 3. Schematic of PMS design. (a) Illustration on the novel cooperative control framework. (b) Illustration on the novel cooperative PMS. (c) Implementation process of SL-eECMS.

3.2 Cooperative PMS for REEV

According to the constructed cooperative control framework, a cooperative PMS is innovatively developed for the power management of the designated REEV. Fig. 3 (b) exhibits the general execution process of the novel strategy. The brand-new strategy accomplishes efficient power management by comprehensively utilizing the super computation capacity in the cooperative control framework. Both the self-learning process in SL-eECMS and the charging level prediction in charging management are moved into MEC module, intensifying the whole application efficiency of the presented strategy in real time. In on-board VCU, the high-level control process can be divided into two phases, of which phase 1 optimally distributes the energy within the power sources by interpolating the generated ED-LUTs. Two types of ED-LUTs are applied: one is for normal driving, and the other one accounts for compulsory charge. Several ED-LUTs, corresponding to the various predicted charging levels, are prepared for compulsory charge. Besides, the charging order that consists of the charging activation signal is outputted in phase 1. The finally output energy distribution is for either normal driving or forced charge decided by the charging order. Phase 2 contributes to operation mode selection and generate control order to lower component controllers on the basis of the energy distribution and charging order from phase 1.

In the REEV, the operation selection between normal operation and forced charge is realized by referring to the charging order that is derived based on instant GPS coordinates and battery SOC in phase 1. In EV mode, the engine speed and torque, selected as the control order, are all set to zero. In HV and forced HV mode, the engine

is regulated to operate along the brake-special fuel consumption (BSFC) line. Hence, the engine speed and torque can be obtained by interpolating the look-up table of operation points in BSFC line (ELUTs 1 and 2 in Fig. 3 (b)) with the distributed optimal engine power (P_{eng_de} in Fig. 3 (b)).

3.2.1 Self-Learning Equivalent Minimization Consumption Strategy

A. Basic ECMS for Energy Management

ECMS has been widely accepted in instant applications, and can provide close effect to global optimization methods [15]. In the implementation, ECMS evaluates the impact on total equivalent fuel consumption from the candidate energy distribution combinations at every control step. The total fuel consumption rate $\dot{m}_{f,eqv}$ can be calculated as:

$$\dot{m}_{f,eqv}(t) = \dot{m}_f(t) + \dot{m}_{ress}(t) \tag{6}$$

where $\dot{m}_{ress}$ is the equivalent fuel consumption rate converted from electricity usage, t is the control step. $\dot{m}_f$ and $\dot{m}_{ress}$ can be respectively calculated by:

$$\begin{cases} \dot{m}_f(t) = \frac{P_{eng}(t)}{\eta_{eng}(t)Q_{lhv}} \\ \dot{m}_{ress}(t) = \frac{s(t)}{Q_{lhv}} P_{batt} \end{cases} \tag{7}$$

where P_{eng} and η_{eng} expresses the engine power and efficiency, respectively; s is the EF dominating the performance of ECMS, wherein the optimal control variable u^* at each step can be calculated as:

$$u^* = argmin[\frac{P_{req}(t)u}{\eta_{eng}(t)Q_{lhv}} + \frac{s}{Q_{lhv}} P_{req}(t)(1-u)] \tag{8}$$

In actual applications, two main manners are widely leveraged in the literature, which are implicit [30] and explicit ECMS [31]. In implicit EMCS [30], Eqns. (6) and (7) are directly applied to calculate optimal solutions for each step. In explicit ECMS, an ED-LUT is calculated offline by evaluating all possible candidates over a grid that is composed of discrete tractive power and battery SOC. Afterwards, the ED-LUT is implemented online to distribute energy within powertrain instantaneously. The application effect of implicit ECMS can be discounted if the amount of discrete candidate solutions is too large. In vehicle's on-board hardware, it is quite labored to evaluate a large scale of candidate solutions with small control steps (e.g. 0.01s). As a consequence, the control step and discrete step should not be too small when applying implicit ECMS. By contrast, the application of explicit ECMS is quite simple just by implementing ED-LUT online, making it more suitable for real-time control. Even though

implicit ECMS behaves slight worse than explicit ECMS in real-time applications, the implicit solving manner makes it rather suitable for adaptive ECMSs, and the EF regulation manner can be freely designed in implicit ECMS. Since the ED-LUT in explicit ECMS is generated offline and reluctant to be tuned instantaneously, the flexibility of explicit ECMS to various driving conditions becomes slightly worse than implicit ECMS.

B. Novel SL-eECMS for Optimal Power Source Management

In this study, a novel SL-eECMS is designed, with the aim of enhancing the adaptability of ECMS to driving environment without scarifying capability in instant applications. Fig. 3 (c) illustrates the implementing process of SL-eECMS. The explicit ECMS is treated as the basic control algorithm in the SL-eECMS. To ensure optimality and adaptability, the iQPSO algorithm is employed to generate ED-LUTs corresponding to different driving conditions. Moreover, a self-learning process is integrated in the designed SL-eECMS to select the most appropriate ED-LUTs for current driving condition. In particular, history ED-LUTs is generated offline by cooperatively exploiting explicit ECMS and iQPSO for different driving conditions that are obtained via the K-means cluster method according to the collected data in real traffic [32]. In case of the obvious variation in driving conditions, self-leaning process, including driving condition learning and ED-LUT updating, is designed accordingly. In the self-learning process, the driving condition of current route segment is compared with history driving condition that is employed to generate history ED-LUT by the Wasserstein distance [33]. New ED-LUT is ordered if difference between current and history driving condition surpasses than the permitted threshold. The driving condition learning and ED-LUT updating is completed in MEC module. The initially employed history ED-LUTs are stored in on-board VCU, while the updated ED-LUT is shared to on-board VCU from MEC module via V2I communication. Furthermore, ED-LUTs for forced charge are not updated in the self-learning process. Wasserstein distance is a distance function defined between two probability distributions on a given metric space, which can detect difference between two probability distributions [33]. For any two probability distributions P and Q on metric space M, the $p-$order's Wasserstein distance can be formulated as:

$$W_P(P,Q) = \left(\inf_{\pi \in \prod(P,Q)} \int \rho(x,y)^p \, d\pi(x,y) \right)^{\frac{1}{p}} \tag{9}$$

where $\rho(x, y)$ is the distance function, x and y are the samples from set Θ, $\prod(P, Q)$ is the probability measures on $\Theta \times \Theta$ with margins P and Q. In existing applications, L_1-Wasserstein distance shows satisfying flexibility in functional analysis [23]. The general equation that describes L_1-Wasserstein distance can be presented as:

$$W_P(P, Q) = \sup_{\|f\| \leq 1} \{E_{x \sim P}[f(x)] - E_{x \sim Q}[f(x)]\} \tag{10}$$

where $\|f\|$ is the Lipschitz semi-norm, as:

$$\|f\| = \sup |f(x) - f(y)| / \rho(x, y) \tag{11}$$

In this paper, L_1-Wasserstein distance is taken into the self-learning process in the SL-eECMS. During operations in certain route segment, the on-duty MECU compares the difference between historical and current driving conditions of next route segments via the L_1-Wasserstein distance. Particularly, the probability distribution of history velocity profile, corresponding to certain driving condition, applied to generate history ED-LUT and stored in on-board VCU, is instantly transmitted to the MECU and contrastively analyzed together with the probability distribution of velocity profile in next route segment. The history velocity profile that is related to certain driving condition is constructed by the method introduced in [34], and the velocity profile of vehicle driving in next route segment can be predicted by the method in our former work [35]. The history ED-LUT is requested to be updated in the on-duty MECU by iQSPO if the calculated L_1-Wasserstein distance is larger than the preset threshold. The self-learning process strives to insure the most appropriate ED-LUT is prepared for new route segment before the REEV moves in. At the beginning of trip, the REEV will select a generic ED-LUT in power source management.

3.2.2 Improved Quantum Particle Swarm Optimization Based ED-LUT Generation

A. iQPSO for ED-LUT Optimization

In PSO, each individual is treated as a particle in a N-dimension space. The independent particle, actually, represents a solution of the optimization problem, and flows through the defined searching space to approximate the optimal solution by updating its position and flow velocity [36]. The position $X_{i,n}$ and velocity $V_{i,n}$ of the ith particle $(1 \leq i \leq M)$ at nth $(1 \leq i \leq N)$ iteration can be respectively defined:

$$\begin{cases} X_{i,n} = \left(X_{i,n}^1, X_{i,n}^2, \ldots X_{i,n}^N\right) \\ V_{i,n} = \left(V_{i,n}^1, V_{i,n}^2, \ldots V_{i,n}^N\right) \end{cases} \tag{12}$$

During the optimal solution searching, the particle updates velocity and position by referring to the personal best position and global best position of the whole swarm, as:

$$\begin{cases} V_{i,n+1}^{j} = V_{i,n}^{j} + c_1\varphi_{i,n}^{j}\left(P_{i,n}^{j} - X_{i,n}^{j}\right) + c_2\psi_{i,n}^{j}\left(G_n^{j} - X_{i,n}^{j}\right) \\ X_{i,n+1}^{j} = X_{i,n}^{j} + V_{i,n+1}^{j} \end{cases} \tag{13}$$

where $j = 1,2,\dots,N$ denotes the dimension space, c_1 and c_2 are the acceleration coefficients, φ and ψ are the uniformly distributed random variables, $P_{i,n}$ is the personal best position, G_n is the global best position. The general optimization problem by PSO can be described, as:

$$\min\ f(x), \textit{ s.t. } X \in S \subseteq R^N \tag{14}$$

where $f(x)$ is the fitness function to evaluate the effect of each iteration, S is the feasible space. The trajectory analysis in [37] proves that PSO algorithm may achieve convergence in optimization if individual particle converges to its local attractor $p_{i,n} = \left(p_{i,n}^{1}, p_{i,n}^{2}, \dots, p_{i,n}^{N}\right)$, as:

$$p_{i,n}^{j} = \frac{c_1\varphi_{i,n}^{j}P_{i,n}^{j} + c_2\psi_{i,n}^{j}G_n^{j}}{c_1\varphi_{i,n}^{j} + c_2\psi_{i,n}^{j}} = \Theta_{i,n}^{j}P_{i,n}^{j} + \left(1 - \Theta_{i,n}^{j}\right)G_n^{j} \tag{15}$$

The local attractor in (15) indicates that each particle is updated to ensure convergence by following $P_{i,n}$ and G_n towards the local attractor. From the perspective of Newtonian dynamics, each particle in the process of convergence moves to its local attractor with its kinect energy decreasing to zero. Consequently, each particle can be seen as one flying point with attraction potential centered at $p_{i,n}$ in Newtonian space. Extending the case to quantum mechanics, each particle can be regarded as one moves in a quantum potential filed in an N-dimension Hilbert space. After endowing particle with quantum behaviors, QPSO can be designed accordingly.

In QPSO algorithm, the quantum state of particle is described by the wave function rather than position and velocity in original PSO [38]. The probability density function that specifies flying position of each particle can be derived from Schrödinger equation [38]. The position of each particle can be measured by employing Monte Carlo inverse transformation [38]. Accordingly, the position of one particle in QPSO can be expressed, as:

$$X_{i,n+1}^{j} = p_{i,n}^{j} \pm \frac{L_{i,n}^{j}}{2} ln\left(\frac{1}{u_{i,n+1}^{j}}\right) \tag{16}$$

where $L_{i,n}$ is the characteristic length of potential well, and $u_{i,n+1}^{j}$ is the sequence of random numbers. The value of $L_{i,n}$ can be determined by following equation, as:

$$L_{i,n}^{j} = 2\tau\left|X_{i,n}^{j} - C_{n}^{j}\right| \tag{17}$$

where τ is the contraction-expansion coefficient, $C_n^j = (C_n^1, C_n^2, \ldots C_n^N)$ is the mean best position that is the average of $P_{i,n}$. Substituting (17) into (16), the manner to update position of each particle in QPSO can be rewritten, as:

$$X_{i,n+1}^{j} = p_{i,n}^{j} \pm \alpha\left|X_{i,n}^{j} - C_{n}^{j}\right| ln\left(\frac{1}{u_{i,n+1}^{j}}\right) \tag{18}$$

Compared with normal PSO, the particle updating in QPSO focuses on only position and omits velocity updating. As a result, the complexity of algorithm and swarm iteration is reduced, and the calculation efficiency is prompted. Despite the promising advantage, the premature ripening of particles still exists in the stage close to convergence, trapping solutions into local optima. To avoid falling into local optima, a Gaussian distribution based mutation is added in particle updating. That is, a perturbation M_n^j=$(M_n^1, M_n^2, \ldots M_n^N)$ is considered in the position updating, as:

$$M_{n}^{j} = C_{n}^{j} + \varepsilon\aleph \tag{19}$$

where ε is the predefined ratio, $\aleph$ denotes a Gaussian distribution. Hence, the position updating manner in the improved QPSO can be written as:

$$X_{i,n+1}^{j} = p_{i,n}^{j} \pm \alpha\left|X_{i,n}^{j} - M_{n}^{j}\right| ln\left(\frac{1}{u_{i,n+1}^{j}}\right) \tag{20}$$

B. ED-LUT Optimization

In explicit ECMS, the solutions are generated offline by traversing all possible combinations of discrete tractive power and battery SOC. As discussed in literature, EF in ECMS is sensitive to controlling performance and deserves be tuned accordingly. To eliminate the constraint on explicit ECMS in adaptive control, iQPSO is preferred to generate ED-LUTs for different driving conditions. In actual applications, EF values corresponding to different battery SOC are optimized under different driving conditions. Hence, particles in iQPSO represents EF values under different battery SOC. The optimization problem by the iQPSO is, within the given driving cycle, to minimize total values of the fitness function, which can be written as:

$$f(x) = \sum \frac{P_{eng}}{\eta_{eng} Q_{lhv}} + \frac{s(X)}{Q_{lhv}} P_{batt} \tag{21}$$

where X is the optimized parameter. During the EF optimization, the subjected constraints can be expressed as:

$$\begin{cases} SOC_{min} \le SOC \le SOC_{max} \\ P_{batt\_min} \le P_{batt} \le P_{batt\_max} \\ T_{eng\_min} \le T_{eng} \le T_{eng\_max} \\ \omega_{eng\_min} \le \omega_{eng} \le \omega_{eng\_max} \\ T_{em\_min} \le T_{em} \le T_{em\_max} \\ \omega_{em\_min} \le \omega_{em} \le \omega_{em\_max} \end{cases} \tag{22}$$

In the ED-LUT optimization for normal operation, constraints on battery can be further expressed, as:

$$\begin{cases} P_{batt\_min} = P_{batt\_min\ \_nor}(SOC) \\ P_{batt\_max} = P_{batt\_max\ \_nor}(SOC) \end{cases} \tag{23}$$

where $P_{batt\_min\ \_nor}$ and $P_{batt\_max\ \_nor}$ respectively denotes the minimum and maximum battery power under certain battery SOC for normal operation. Similarly, the constraints on battery in the ED-LUT optimization for forced charge can be also described, as:

$$\begin{cases} P_{batt\_min} = P_{batt\_min\ \_cha}(SOC) \\ P_{batt\_max} = 0 \end{cases} \tag{24}$$

where $P_{batt\_min\ \_nor}$ denotes the minimum battery power under certain battery SOC for forced charge.

3.2.3 Adaptive Neuro-Fuzzy Inference System Based Charging Management

A. Adaptive Neuro-Fuzzy Inference System (ANFIS)

The ANFIS is a hybrid method that combines the fuzzy intelligence and neural network, which is quite qualified in solving complex and nonlinear problems. In ANFIS, the neural network is utilized to estimate the parameters of membership functions, and efficiently tunes the conversation of human intelligence in various conditions. Assuming ANFIS holds two inputs and one output, the two fuzzy if-then rules of a Sugeno fuzzy model in ANFIS can be described as [39]:

Rule 1: if x is A_1 and y is B_1; then $f = p_1x + q_1y + r_1$;

Rule 2: if x is A_2 and y is B_2; then $f = p_2x + q_2y + r_2$;

where A_1, A_2, B_1 and B_2 are the nonlinear parameters; p_1, p_2, q_1, q_2, r_1 and r_2 are the linear parameters.

The ANFIS generally consists of five layers, which is referred to fuzzification, product, normalized, defuzzification and output layer. The transformation and calculation in the five layers can be:

Layer 1: The relationship between the two inputs and the corresponding outputs are described by the membership function, as:

$$\begin{cases} O_j^1 = \mu_{A_j}(I_1)\ j = 1,2 \\ O_j^1 = \mu_{B_j}(I_2)\ j = 1,2 \end{cases} \tag{25}$$

where I_1 and I_2 are the two mentioned inputs, μ_{A_j} is the membership function, O_j^1 is the output of this layer. The membership function can be expressed, as:

$$\mu_{A_j}(x) = \frac{1}{1+\left[\left(\left(x-c_j\right)/a_j^2\right)\right]^{bj}}\ j = 1,2 \tag{26}$$

where a_j, b_j, c_j are the premise parameters.

Layer 2: The weight of a rule w_j for next layer is calculated by multiplicative operator, as:

$$O_j^2 = w_j = \mu_{A_j}(I_1) \cdot \mu_{B_j}(I_2)\ j = 1,2 \tag{27}$$

Layer 3: The weight of rule at j th node is normalized, as:

$$O_j^3 = \overline{w_j} = \frac{w_j}{w_1+w_2}\ j = 1,2 \tag{28}$$

Layer 4: The contribution of each rule toward the final output is computed, as:

$$O_j^4 = \overline{w_I} z_i = \overline{w_I}\left(p_j I_1 + q_j I_2 + r_j\right) \tag{29}$$

where p_j, q_j, and r_j are the consequent parameter.

Layer 5: The overall output of ANFIS is obtained by employing a summing function, as:

$$O_j^5 = \sum_j \overline{w_I} z_i = \sum_j \frac{w_j}{w_1+w_2}\left(p_j I_1 + q_j I_2 + r_j\right) \tag{30}$$

The ANFIS is trained via supervised learning. To regulate the parameters in ANFIS, a hybrid algorithm is exploited by integrating the gradient descent method and least square method, of which the first method is in charge of estimating the premise parameters (a_j, b_j, c_j), and the latter one determines the consequent linear parameters (p_j, q_j, r_j). In the forward training process, the least square method captures the consequent parameters while keeping the premised parameters fixed. Then, the error between the predicted values and raw data are propagated backward. The least square method identifies the premise parameters by minimizing the quadratic cost function. Moreover, in the backward propagation, consequent parameters remain unchanged.

B. Charing Level Prediction in Online Charging Management

To satisfy the requirement of zero emission driving, sufficient electricity must be replenished during vehicle driving in Approach zone. To charge battery online properly, the consumed electricity in ZE zone needs to be predicted. Nevertheless, it is impractical to precisely predict energy consumption under the limits of massive computation, complicated traffic data distribution and powertrain dynamic. Instead of accurately predicting the utilized electricity, CM-MECU covers Approach zone only forecasts the charging level by ANFIS macroscopically. The inputs of the ANFIS based predictor are vectors related to traffic and vehicle status, while the output of the ANFIS based predictor are three different charging levels: low, medium and high. Diversified ED-LUT, according to the identified charging level, is implemented in explicit ECMS during the charging stage afterward. In the charging level prediction, traffic and vehicle status vectors are chosen by binary Dragonfly method [40], as:

$$\begin{cases} T_s = (V_{ave}, A_{ave}, \mathrm{T}_{v=0}, N_{rs}, N_{bs}) \\ V_s = (v_{ave}, a_{ave}, \tau_{v=0}) \end{cases} \tag{31}$$

where T_s and V_s denotes the traffic and vehicle status, respectively; V_{ave}, A_{ave} and $\mathrm{T}_{v=0}$ means the average velocity, acceleration and vehicle stop time of all the vehicles on certain route segment, respectively; v_{ave} and a_{ave} expresses the predicted average velocity and acceleration of the target vehicle on certain route segment; and $\tau_{v=0}$ represents the predicted stop time of the target vehicle on certain sector. v_{ave}, a_{ave} and $\tau_{v=0}$ can be obtained in CM-MECU by method in our previous work based on the shared traffic data [35]. Likewise, V_{ave}, A_{ave} and $\mathrm{T}_{v=0}$ can be statistically calculated in CM-MECU. The defined charging levels are obtained via clustering energy consumption results that are calculated by:

$$E_{req\_ze} = \int_{t_0}^{t_d} P_{req}(t)dt \tag{32}$$

where t_0 is the predicted initial time when the REEB drives into ZE zone, t_d is the predicted terminal time that the REEB leaves ZE zone, and E_{req_ze} is the required energy in ZE zone. t_0 and t_d can also be captured by method in [35]. To clearly categorize charging levels, velocity profiles collected from real driving in ZE zone are applied to calculate electricity consumption by (32), preparing for the K-mean clustering based charging level classification [32]. Then, typical driving cycles for generating ED-LUTs in forced charge are derived by the method introduced in [34] on the strength of driving profiles that are homologous to diverse charging levels. In online charging management, one particular ED-LUT for certain charging level is implemented in on-board VCU.

In other words, the predicted charging levels, named as “low”, “medium” and “high”, will instruct the adaptation degree of different ED-LUTs, in which the EF values corresponding to different charging level are properly optimized. Consequently, various final charging level can be obtained to make sure there is enough electric power to drive the vehicle in the ZE zone.

As described previously, the forced charge is activated according to instant geographic position and battery SOC. In other words, the vehicle switches into the forced charge mode on the condition that it is located in Approach zone, and the initial battery SOC in ZE zone is smaller than the defined value. The initial battery SOC in ZE zone can be predicted, as:

$$SOC_{ini\_ze} = SOC_i - \frac{E_{req\_ap}+E_{req\_ze}}{3.6W_b} \tag{33}$$

where SOC_{ini_ze} and SOC_i denotes the initial battery SOC in ZE zone and instant battery SOC in Approach zone, respectively; E_{req_ap} is the required energy for the remaining driving in Approach zone, which can be calculated by the same method described in (32); and W_b is the battery capacity in kWh. In online charging management, the future traffic state is predicted every 30s.

Indeed, each ED-LUT exploits one fixed EF, which has been optimized by the QPSO algorithm offline. If the driving condition is detected to change significantly by calculating the Wassertein Distance, the ED-LUT will be updated by generating a new optimal EF value. The application of ED-LUT is to make sure the online application capacity of ECMS can be maximized, thus avoiding the repeating solution searching in each online control step. In the next step, substantial simulations and corresponding discussions are conducted to examine the performance of the devised control framework.

IV. Simulation and Evaluation

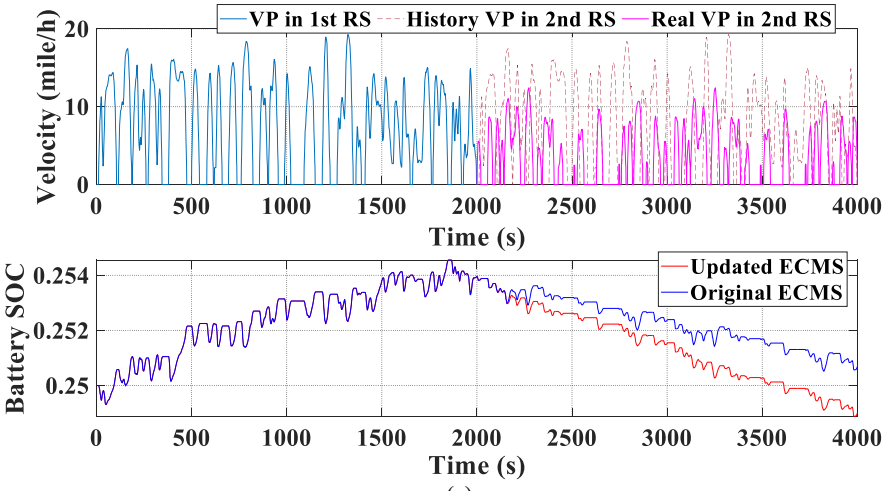
In this section, comprehensive simulations are performed to validate the performance of the proposed cooperative PSMS in adaptive energy management and online charging management. The simulation evaluation can be divided mainly into two parts: one is focused on the adaptive energy management by the SL-eECMS, and the other one pays attention to the ANFIS based online charging management. Note that during the assessment, VP denotes velocity profile, and RS means the route segment. Moreover, LC, MC and HC represent low, medium and high level charging level. The simulation is conducted on a workstation with an i7-8700 processor and 16

gigabyte memory. In the evaluation test, the virtual scenarios installed with IoVs is constructed by incorporating PreScan and Matlab/Simulink toolbox. The built virtual scenarios provide simulation on control application in IoVs. Note that all the initial battery SOC is set to 0.25 to enable charging process can be activated to comprehensively assess the performance of the raised solution.

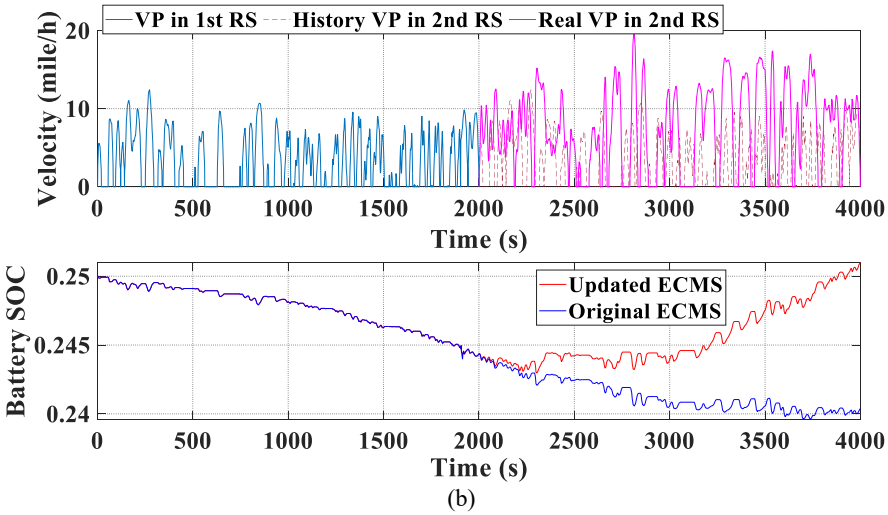
A. Assessment on SL-eECMS Based Energy Management

To investigate the capability of the designed SL-eECMS, two case studies are performed. In each case study, the first 2000 s indicates that the REEV operates in current route segment, while it drives into next route segment after 2000 s. The velocity profiles for case studies are presented in Fig. 4. In the first case, real velocities of next route segment are obviously smaller than the history velocities for generation of ED-LUT. While, real velocities of next route segment are rather larger than the history velocities in case study 2.

As shown in Fig. 4, when the driving condition of next route segment presents noteworthy differences, the proposed SL-eECMS updates the ED-LUT timely. In the first case study, instantaneous driving velocity of next route segment tends to be slower, discouraging the engine's participation due to poor efficiency at low speed. Hence, the updated ED-LUT urges battery to output more electricity, contributing to energy saving by more efficient powertrain operation. In the second case, the velocities of next route segment are much higher than history velocities. The high driving velocities favour engine's power and battery's charging because of higher operation efficiencies. The proposed SL-eECMS detects the difference and updates the ED-LUT. Thus, more electricity is charged into battery during 2000 s to 4000 s in next route segment in case study 2.



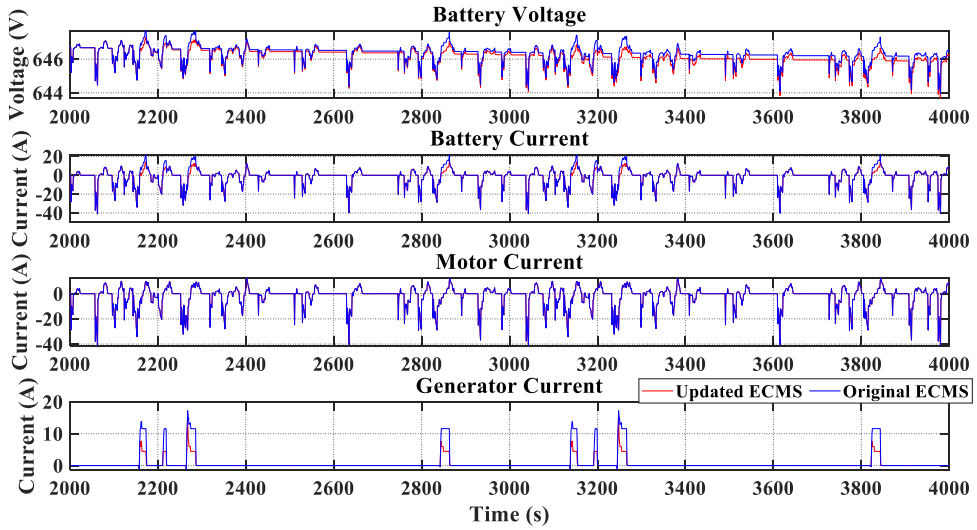
(a)



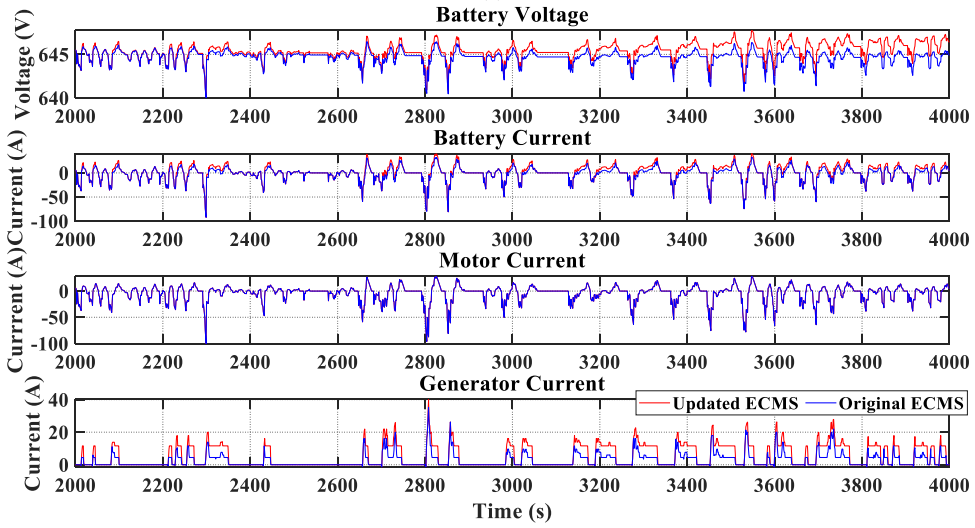
(b)

Fig. 4. Comparison between SL-eECMS and Original ECMS. (a) Self-learning process of SL-EMCS in case 1. (b) Self-learning process of SL-EMCS in case 2. VP denotes the velocity profile, RS means the route segment, Original ECMS expresses common explicit ECMS.

Fig. 5 exhibits the component performance by the SL-eECMS in two cases. As the velocity profiles before 2000 s remain the same in two cases, Fig. 5 only show the component performance from 2000 s to 4000 s. The analysis shown in Fig. 4 (a) reveals that more electric energy in next route segment is recommended in the first case study. As a consequence, the generator current, which describes the operation of engine and the variation of battery voltage by the updated ED-LUT, is smaller than that by the history ED-LUT. Simultaneously, the battery current is larger than that by the original ED-LUT. In case study 2, the engine is ordered to operate with higher proportion by new ED-LUT for the improved efficiency. Consequently, the generator current in next route segment by new ED-LUT is much larger than that by the saved ED-LUT. Owing to the charged electricity, the battery voltage by new ED-LUT in next sector becomes higher. Besides, the resulted negative battery current supplied by the fresh ED-LUT is smaller, expressing that more electricity is replenished by APU. Fig. 6 highlights the engine operation points projected on the engine-fuel-consumption contour, and further evaluates the performance of the raised SL-eECMS among benchmarks. As can be clearly illustrated, the performance of SL-eECMS performs better than original ECMS, with more engine operation points distributing in the low fuel consumption regions. The performance of SL-eECMS, judged by the engine operation points in higher efficient zones, behaves more closed to DP, compared with the original ECMS, leading to a promising energy-saving performance.



(a)



(b)

Fig. 5. Component performance compare in two cases. (a) Component performance in case 1. (b) Component performance by in case 2. Original ECMS expresses common explicit ECMS.

Fig. 7 illustrates the comparison in fuel consumption by different methods in two case studies. Original ECMS denotes the common explicit ECMS without integrating the self-learning process. As the driving conditions are all the same at first 2000 s in two cases, there is no difference in fuel consumption by various methods. After 2000 s, the diving conditions of next route segment present prominent difference, compared with the historical conditions, inducing ED-LUT updating in SL-eECMS. In the first case study, more electricity is consumed by the updated ED-LUT to avoid engine operation in poor efficiencies, thus saving certain fuel consumption. Even though the engine is suggested to operate more by the updated ED-LUT in the second case study, the fuel consumption by the new ED-LUT is still less than that by the history ED-LUT. The reason leads to less fuel consumption lies in that the iQPSO based generation method adds optimal knowledge into EF tuning, thereby capturing the most

appropriate EF vector for the specific driving condition. Table 2 lists the numerical results of energy consumption and calculation time by different methods, further assessing the capability of SL-eECMS.

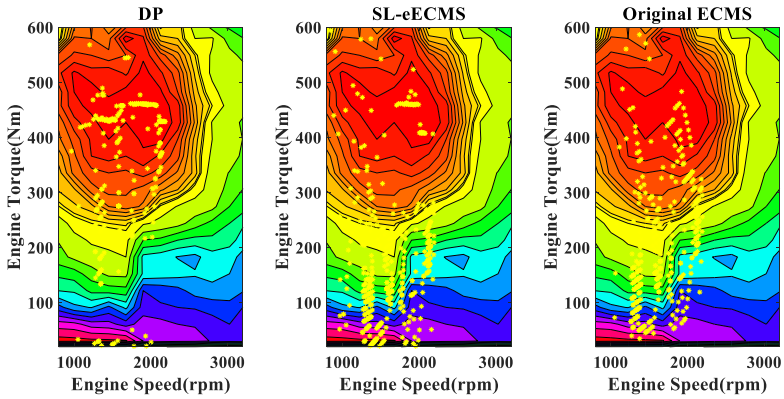


Fig. 6. Compare of engine operation points by different methods.

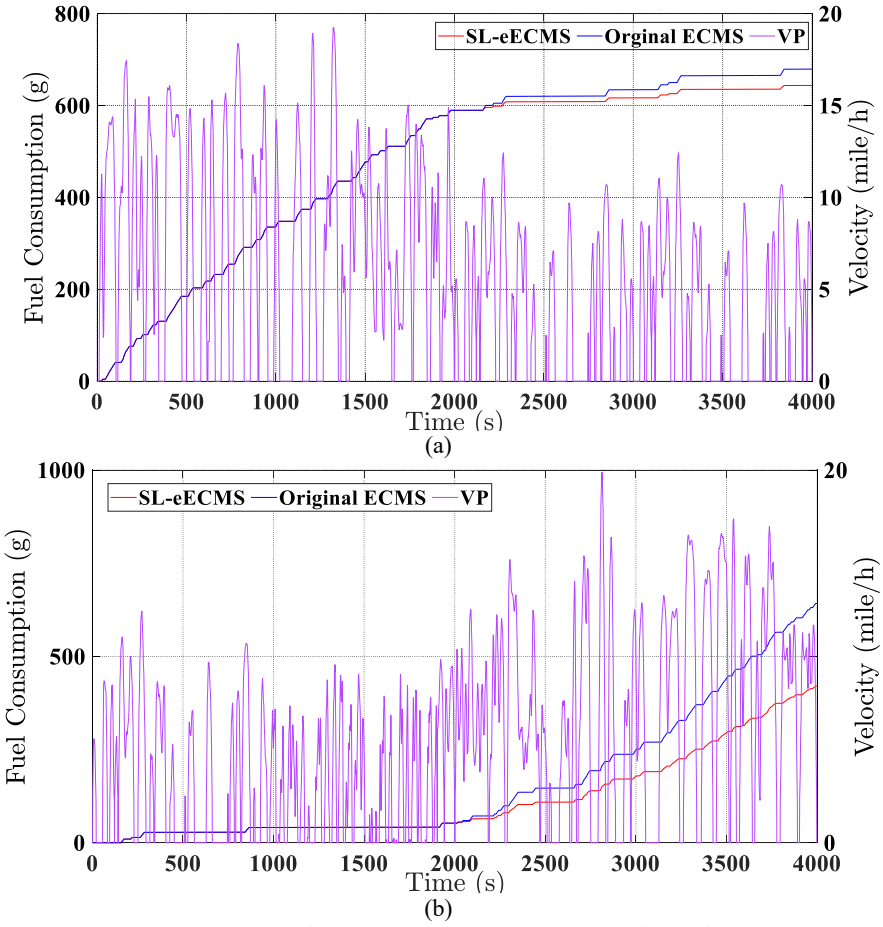


Fig. 7. Fuel consumption compare in two cases. (a) Fuel consumption compare in case 1. (b) Fuel consumption compare in case 2. VP denotes the velocity profile, Original ECMS expresses common explicit ECMS.

Table 2 Numerical results of energy consumption and calculation time by different methods

Case	Method	Fuel Consumption (g)	Equivalent Fuel Consumption (L/100km)	Fuel Economy Improvement (%)	Total Calculation Time (s)	Step Calculation Time (s)
1	DP	626.1	14.25	6.06	8864	2.216
	SL-eECMS	643.7	14.87	1.97	2932	0.733

	Original ECMS	679.3	15.17	-	384	0.096
2	DP	609.6	14.17	9.81	8852	2.213
	SL-eECMS	620.2	14.59	7.13	2944	0.736
	Original ECMS	642.1	15.71	-	392	0.098

Note: Original ECMS expresses common explicit ECMS, and DP means the dynamic programming as the benchmark.

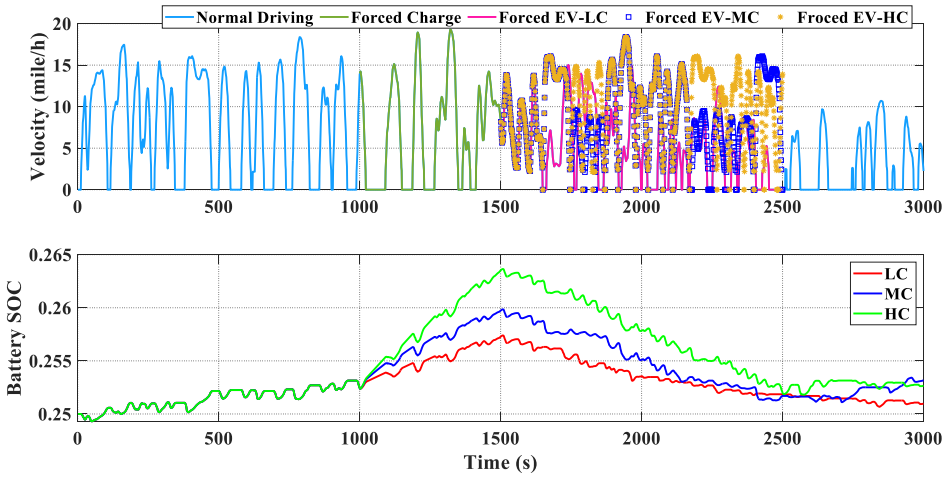
In Table 2, the equivalent fuel consumption denotes the sum of the original fuel consumption and the equivalent fuel converted from the consumed electricity. In addition, the calculation time means the CPU processing time, and the step calculation time is calculated by dividing total calculation time over simulation steps. In the simulation, the sampling time and control step are set to 1 s. According to the numerical results listed in Table 2, the proposed SL-eECMS achieves better fuel economy than common explicit ECMS in both case studies. This is because SL-eECMS is enabled with adaptabilities to various driving conditions after integrating the self-learning process. The enhanced adaptability to various driving environment narrows the gap between DP and ECMS. The optimality of SL-eECMS is raised to 95.8% of that by DP, which is a significant improvement, compared with the common explicit ECMS. Moreover, SL-eECMS is proved to be qualified in instantaneous optimization in practical applications. The step calculation time by SL-eECMS is smaller than 1 s, manifesting that the updated ED-LUT can be prepared before vehicle moves into next route segment and consequently satisfies the requirement on control step in real time. The common explicit ECMS, although it costs less time than SL-eECMS, cannot be suitable for all driving conditions, highlighting worse performance than the novel SL-eECMS.

4.2 Evaluation on ANFISI Based Charging Management

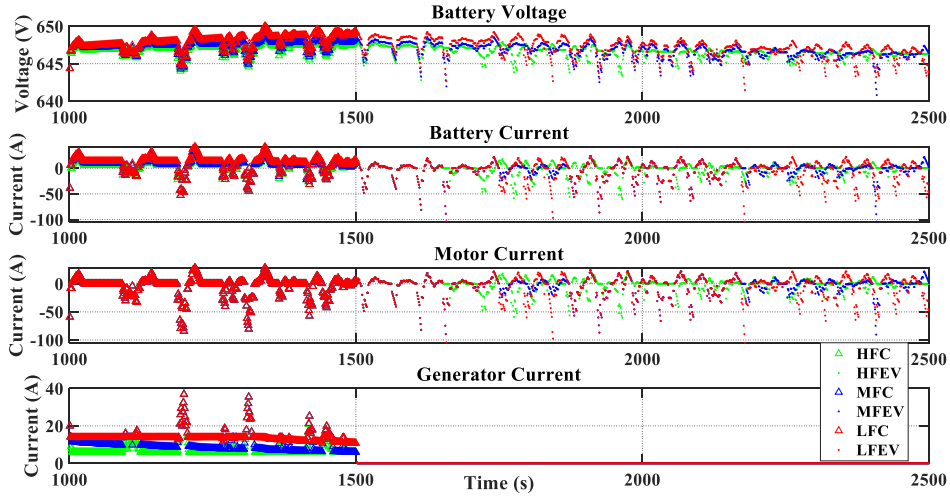
Requested by the specific requirement, the forced charge for future driving in ZE zone sparks the particular design of online charging management. To evaluate the performance of the developed online charging management method, an exclusive driving cycle is designed, as shown in Fig. 8 (a). The driving cycle is dived into four parts: normal driving 1 (0 s-1000 s), forced charge (1000 s-1500 s), forced EV mode (1500 s-2500 s) and normal driving 2 (2500 s-3000 s). In forced EV sector, there are three different velocity profiles corresponding to diverse driving requirements in ZE Zone and disparate charging levels.

As illustrated in Fig. 8 (a), the forced charge is activated when the vehicle drives into the Approach Zone. The predicted charging levels, referring to as different driving requirement in ZE Zone, informs the on-board VCU to charge battery with diverse levels, resulting in different battery SOC increase from 1000 s to 1500 s. The designed charging management method avoids the exhaustive discharge in ZE zone by the ANFIS based charging

prediction, thereby enabling the terminal battery SOC in ZE zone close to the initial battery in Approach Zone. Fig. 8 (b) shows component operations under different charging levels and driving requirement. In the forced charge, the generator current under HC is larger than MC and LC, providing larger charging power by APU. With larger charging power, the battery voltage under high charging level are charged to higher values. The indexes of battery and generator prove that the online charging management can meet the specific driving requirement under different driving conditions in ZE zone.



(a)



(b)

Fig. 8. Compare in charging management. (a) Comparison of battery SOC in different charging management. Forced EV-LC means the velocity profile in ZE zone corresponding to low charging level, Forced EV-MC expresses the velocity profile in ZE zone corresponding to medium charging level, and Forced EV-HC represents the velocity profile in ZE zone corresponding to high charging level. (b) Comparison of component operation in different charging management. HFC, MFC and LFC respectively means forced charge corresponding to high, medium and low charging level; HFEV, MFEV and LFEV respectively denotes forced pure electric drive on different driving conditions that are corresponding to high, medium and low charging level.

Table 3 compares the energy consumption and calculation time in charging management. The eECMS-LC, MC and HC respectively means the explicit ECMS based charging management strategy under low, medium and

high charging levels. In the comparison, the rule-based method that charges battery to certain SOC value is adopted as the benchmark. Apparently, the raised online charging management strategy can charge battery efficiently under different charging levels. During online charging, APU operates to output the tractive power and charge battery simultaneously. The proposed method not only enables that enough battery can be replenished, but also regulates APU to drive vehicle effectively. Compared with the rule-based method, the proposed charging management strategy can save fuel consumption by up to 42.8%. The satisfied performance of the presented charging management strategy owes to the optimal solution search by the explicit ECMS and accurate charging level prediction by the ANFIS. The prediction time listed in Table 3 shows that ANFIS can predict charging levels within 1 s, making it suitable for real-time applications.

Table 3 Numerical results of energy consumption and calculation time in charging management

Method	Fuel Consumption (g)	Fuel Reduction (%)	Prediction Time (s)
eECMS-LC	269.6	42.8	0.619
eECMS-MC	316.9	32.8	0.621
eECMS-HC	389.2	17.4	0.617
Rule based method	471.6	-	-

Note: eECMS-LC, eECMS-MC and eECMS-HC respectively means the explicit ECMS methods for low, medium and high level charging.

Through the sufficient simulation evaluation, it can be concluded that the proposed method improves the adaptability of common explicit ECMS to driving environment, and attains better fuel economy for REEV. Furthermore, the ANFIS based charging level prediction ideally directs online charging management with the instruction from explicit ECMS, further contributing to energy saving of REEV.

4.3 Evaluation in HIL test

To further validate the raised method in fast optimal solving, a HIL test is conducted on the platform, as shown in Fig. 9. The HIL test platform is made up of three host personnel computers (PCs) and one real-time controller. The control strategies are compiled in host PC 1 and downloaded to the real-time dSpace based controller. The vehicle model, including component sub-models, is executed in PC 2 and PC 3. The information of ND, Approach and ZE zones shown in Fig. 1 and driving cycles shown in Fig. 4 and Fig. 8 (a) are combined into three test scenarios, which are supplied by host PC 1. The communication between the controller and host PCs is attained via the CAN bus communication.

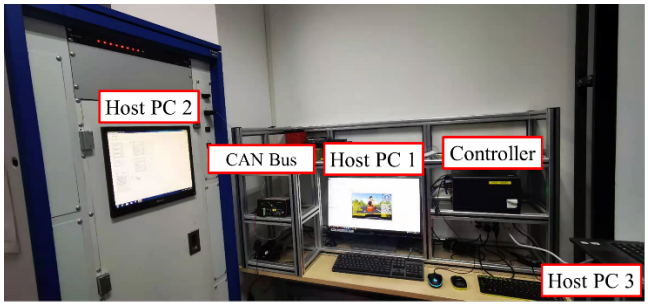


Fig. 9. HIL test platform.

The HIL test is respectively performed on the same driving cycles shown in Fig. 4 and Fig. 8 (a). Table 4 lists the test results of the proposed SL-eECMS in energy management, and Table 5 demonstrates the assessment of the raised method in charging management. As can be found from Table 4, the raised SL-eECMS performs better than original ECM, with 4.8% fuel saving in Case 1 study, and also outperforms the rule-based strategy in real time, with 6.9% reduction in fuel consumption. In Table 4, the one step time cost means the one-step optimal solution solving time. As can be found, the raised SL-eECMS can obtain optimal solution for one-step control in less than 0.01 s, satisfying the requirement for real-time implementation. In Table 5, the proposed on-board charging management method performs well under different charging levels, compared with the rule-based strategy in practical situations. The optimal solution solving speed is also acceptable, with one step time cost less than 0.01 s under various charging levels, proving the online application potential of the raised algorithm.

Table 4 Numerical results of energy consumption and calculation time by different methods

Case	Method	Fuel Consumption (g)	Fuel Reduction (%)	One Step Time Cost (s)
1	Rule based Strategy	701.2	-	0.0056
	SL-eECMS	652.3	6.9	0.0093
	Original ECMS	687.1	2.1	0.0087
2	Rule based Strategy	667.2	-	0.0055
	SL-eECMS	626.2	6.1	0.0096
	Original ECMS	649.3	2.7	0.0086

Table 5 Numerical results in HIL test

Method	Fuel Consumption (g)	Fuel Reduction (%)	One Step time Cost (s)
eECMS-LC	271.3	44.1	0.0091
eECMS-MC	319.5	34.1	0.0093
eECMS-HC	392.3	19.0	0.0095
Rule based method	484.5	-	0.0067

V. CONCLUSIONS

This paper presents a novel cooperative power source management strategy by virtue of Internet of Vehicles. The developed cooperative strategy incorporates instantaneous energy management and charging management, accomplishing efficient power source management. The self-learning explicit equivalent minimization

consumption strategy, integrating the improved quantum particle swarm optimization, is preferred to complete energy management, and the adaptive neuro-fuzzy inference system is applied in online charging management. Simulation and hardware-in-the-loop test evaluation demonstrates the anticipated performance of the raised method. In energy management, the presented method behaves better than common equivalent minimization consumption strategy, and the optimality raised by the proposed method reaches 95.8% of that by global dynamic programming. In charging management, proper online battery charge is performed according to the predicted charging level. By the novel developed method, the fuel consumption during online charge can be saved up to 42.8%.

In the future work, we plan to perform the research on understanding the interaction between wireless transmission quality and implication of the cooperative control strategies, trying to minimize the communication time delay and maximize the efficiency of information transformation. Besides, vehicle-environment cooperation methods that incorporate more information into vehicle control will be carefully investigated, and the impact on vehicle performance from driving behaviours will also be studied.

ACKNOWLEDGEMENT

The work is funded by the National Natural Science Foundation of China (No. 52002046 and 52272395) in part, Chongqing Fundamental Research and Frontier Exploration Project (No. CSTC2019JCYJ-MSXMX0642) in part, and Science and Technology Research Program of Chongqing Municipal Education Commission (No. KJQN201901539) in part.

REFERENCE

[1] Lin, Boqiang, and Junpeng Zhu. "Impact of energy saving and emission reduction policy on urban sustainable development: Empirical evidence from China." Applied Energy 239 (2019): 12-22.
[2] Ling, Zaili, et al. "Sulfur dioxide pollution and energy justice in Northwestern China embodied in West-East Energy Transmission of China." Applied Energy 238 (2019): 547-560.
[3] Pavić, Ivan, Hrvoje Pandžić, and Tomislav Capuder. "Electric vehicle based smart e-mobility system–Definition and comparison to the existing concept." Applied Energy 272 (2020): 115153.
[4] Zhang, Yaoli, et al. "Mobile charging: A novel charging system for electric vehicles in urban areas." Applied Energy 278 (2020): 115648.
[5] Al-Ogaili, Ali Saadon, et al. "Estimation of the energy consumption of battery driven electric buses by integrating digital elevation and longitudinal dynamic models: Malaysia as a case study." Applied Energy 280 (2020): 115873.
[6] López-Ibarra, Jon Ander, et al. "Plug-in hybrid electric buses total cost of ownership optimization at fleet level based on battery aging." Applied Energy 280 (2020): 115887.
[7] B. V. Padmarajan, Andrew McGordon, and Paul A. Jennings., " Blended rule-based energy management for PHEV: System structure and strategy.," IEEE Transactions on Vehicular Technology 65.10 (2015), pp. 8757-8762.
[8] Li, Ji, et al. "Driver-identified supervisory control system of hybrid electric vehicles based on Spectrum-guided fuzzy feature extraction." IEEE Transactions on Fuzzy Systems (2020).

[9] Zhou, Wei, et al. "Dynamic programming for new energy vehicles based on their work modes Part II: Fuel cell electric vehicles." Journal of Power Sources 407 (2018): 92-104.
[10] S. Xie, et al. , "Pontryagin's minimum principle based model predictive control of energy management for a plug-in hybrid electric bus.," Applied energy 236 (2019), pp. 893-905.
[11] J. Li, et al. , "Research on Equivalent Factor Boundary of Equivalent Consumption Minimization Strategy for PHEVs.," IEEE Transactions on Vehicular Technology (2020).
[12] Xie, Shaobo, et al. "Model predictive energy management for plug-in hybrid electric vehicles considering optimal battery depth of discharge." Energy 173 (2019): 667-678.
[13] Liu, Chang, and Yi Lu Murphey. "Optimal power management based on Q-learning and neuro-dynamic programming for plug-in hybrid electric vehicles." IEEE Transactions on Neural Networks and Learning Systems (2019).
[14] Zou, Runnan, et al. "A Self-adaptive Energy Management Strategy for Plug-in Hybrid Electric Vehicle based on Deep Q Learning." Journal of Physics: Conference Series. Vol. 1576. No. 1. IOP Publishing, 2020.
[15] Yang, Chao, et al. "Adaptive real-time optimal energy management strategy based on equivalent factors optimization for plug-in hybrid electric vehicle." Applied Energy 203 (2017): 883-896.
[16] Onori, Simona, Lorenzo Serrao, and Giorgio Rizzoni. Hybrid electric vehicles: Energy management strategies. London: Springer, 2016.
[17] Zhang, Yuanjian, et al. "Energy management strategy for plug-in hybrid electric vehicle integrated with vehicle-environment cooperation control." Energy (2020): 117192.
[18] Guo, Longhua, et al. "A secure mechanism for big data collection in large scale internet of vehicle." IEEE Internet of Things Journal 4.2 (2017): 601-610.
[19] Xu, Biao, et al. "V2I based cooperation between traffic signal and approaching automated vehicles." 2017 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2017.
[20] Wang, Yujie, et al. "Multiple-grained velocity prediction and energy management strategy for hybrid propulsion systems." Journal of Energy Storage 26 (2019): 100950.
[21] Dolui, Koustabh, and Soumya Kanti Datta. "Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing." 2017 Global Internet of Things Summit (GIoTS). IEEE, 2017.
[22] Abbas, Nasir, et al. "Mobile edge computing: A survey." IEEE Internet of Things Journal 5.1 (2017): 450-465.
[23] Zhang, Fengqi, Junqiang Xi, and Reza Langari. "Real-time energy management strategy based on velocity forecasts using V2V and V2I communications." IEEE Transactions on Intelligent Transportation Systems 18.2 (2016): 416-430.
[24] Cao, Yue, et al. "MEC Intelligence Driven Electro-Mobility Management for Battery Switch Service." IEEE Transactions on Intelligent Transportation Systems (2020).
[25] Rogelj, J., Schaeffer, M., Meinshausen, M., Knutti, R., Alcamo, J., Riahi, K., & Hare, W. (2015). Zero emission targets as long-term global goals for climate protection. Environmental Research Letters, 10(10), 105007.
[26] van Soest, H. L., den Elzen, M. G., & van Vuuren, D. P. (2021). Net-zero emission targets for major emitting countries consistent with the Paris Agreement. Nature communications, 12(1), 1-9.
[27] de Bok, M., Tavasszy, L., & Thoen, S. (2020). Application of an empirical multi-agent model for urban goods transport to analyze impacts of zero emission zones in The Netherlands. Transport Policy.
[28] Hemadeh, Ibrahim A., et al. "Millimeter-wave communications: Physical channel models, design considerations, antenna constructions, and link-budget." IEEE Communications Surveys & Tutorials 20.2 (2017): 870-913.
[29] X. Gao, L. Dai, and A. M. Sayeed, "Low RF-Complexity Technologies to Enable Millimeter-Wave MIMO with Large Antenna Array for 5G Wireless Communications," IEEE Communications Magazine, vol. 56, no. 4, pp. 211-217, Apr 2018.
[30] Lei, Zhenzhen, et al. "An adaptive equivalent consumption minimization strategy for plug-in hybrid electric vehicles based on traffic information." Energy 190 (2020): 116409.
[31] Paganelli, Gino, et al. "Control development for a hybrid-electric sport-utility vehicle: strategy, implementation and field test results." Proceedings of the 2001 American Control Conference.(Cat. No. 01CH37148). Vol. 6. IEEE, 2001.
[32] Likas, A., Vlassis, N., & Verbeek, J. J. (2003). The global k-means clustering algorithm. Pattern recognition, 36(2), 451-461.
[33] Zhang, Ming, et al. "Wasserstein Distance guided Adversarial Imitation Learning with Reward Shape Exploration." arXiv preprint arXiv:2006.03503 (2020).
[34] Hung, W. T., Tong, H. Y., Lee, C. P., Ha, K., & Pao, L. Y. (2007). Development of a practical driving cycle construction methodology: A case study in Hong Kong. Transportation Research Part D: Transport and Environment, 12(2), 115-128.
[35] Zhang, Yuanjian, et al. "Optimal energy management strategy for parallel plug-in hybrid electric vehicle based on driving behavior analysis and real time traffic information prediction." Mechatronics 46 (2017): 177-192.
[36] Sun, Jun, et al. "Quantum-behaved particle swarm optimization: Analysis of individual particle behavior and parameter selection." Evolutionary computation 20.3 (2012): 349-393.

[37] Clerc, Maurice, and James Kennedy. "The particle swarm-explosion, stability, and convergence in a multidimensional complex space." IEEE transactions on Evolutionary Computation 6.1 (2002): 58-73.
[38] Fu, Yangguang, Mingyue Ding, and Chengping Zhou. "Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV." IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 42.2 (2011): 511-526.
[39] Adedeji, Paul A., et al. "Hybrid adaptive neuro-fuzzy inference system (ANFIS) for a multi-campus university energy consumption forecast." International Journal of Ambient Energy (2020): 1-10.
[40] Mafarja, Majdi M., et al. "Binary dragonfly algorithm for feature selection." 2017 International Conference on New Trends in Computing Sciences (ICTCS). IEEE, 2017.