Fuel Minimization of a Hybrid Electric Racing Car by Quasi-Pontryagin's Minimum Principle

Tong Liu, *Member, IEEE*, Lei Feng, *Member, IEEE*, and Wenyao Zhu

***Abstract*—This paper improves the fuel efficiency of a student-made parallel hybrid electric racing car whose internal combustion engine (ICE) either operates with peak efficiency or is turned off. The control to the ICE thus becomes a binary problem. Owing to the very limited computation resource onboard, the energy management strategy (EMS) for this car must have small time and space complexities. A computationally efficient controller that combines the advantages of dynamic programming (DP) and Pontryagin's minimum principle (PMP) is developed to run on a low-cost microprocessor. DP is employed offline to calculate the optimal speed trajectory, which is used as the reference for the online PMP to determine the real-time ICE on/off status and the electric motor (EM) torques. The normal PMP derives the optimal costate trajectory through solving partial differential equations. The proposed quasi-PMP (Q-PMP) method finds the costate from the value function obtained by DP. The fuel efficiency and computational complexity of the proposed controller are compared against several state of the art methods through both model-in-the-loop (MIL) and processor-in-the-loop (PIL) simulations. The new method reaches similar fuel efficiency as the explicit DP, but requires less than 1% onboard flash memory. The performance of the Q-PMP controller is compared between binary-controlled and continuously controlled ICEs. It achieves roughly 12% higher fuel efficiency for the binary ICE with only approximately 1/3 CPU utilization.**

***Index Terms*—Hybrid electric vehicle, Energy management strategy, Dynamic programming, Pontryagin's minimum principle, Binary controlled internal combustion engine.**

I. Introduction

HYBRID electric vehicles (HEVs) have shown compelling potentials on energy conservation and emission reduction as the electric motor (EM) on the powertrain allows the internal combustion engine (ICE) to operate with high efficiency. The EM can also convert the mechanical energy from the ICE or the brake into electric energy [1], [2]. By these means, the HEV enjoys better fuel economy than the conventional vehicle solely driven by ICE, and has much larger driving range than the pure electric vehicle [3], [4].

The ICE efficiency is typically determined by a 2-dimension (2D) efficiency map with its speed and torque as the inputs [5], [6]. It varies a lot during a trip due to the fast changes of its speed and torque output. With the aid of EM, the ICE has more opportunities to run at (or close to) its peak efficiency points. If the ICE torque with peak efficiency is not enough to satisfy the driving demand, the EM provides the additional torque; otherwise, if it is larger than the requirement, the excess torque is recuperated by the EM to generate electricity for future utilization [7], [8].

The HEV studied in this paper is a racing car for Shell Eco-marathon (SEM).[1] In this competition, the performance of an HEV is evaluated by the overall fuel efficiency and the terminal charge of the electric energy storage (EES) should be carefully protected. Motivated by the smart usage of the EM, the ICE in this hybrid electric racing car is configured to always work with peak efficiency when it is switched on. Therefore, the control to the ICE becomes a binary control on its on/off status rather than calculating the continuous torque value, and the computation complexity can be greatly reduced.

While this binary control simplifies computation, it brings in a challenge of avoiding frequent ICE on/off switches, because each switch consumes additional energy and results in time delay on the energy flow from the ICE to the drive wheels. Thus, an appropriate energy management strategy (EMS) for this racing car should not only maximize the energy efficiency but also regulate the operation of ICE on/off switches.

The past decades have witnessed the rapid development of a wide variety of EMS for HEV. According to [9]-[11], the popular EMS at present can be broadly classified into rule-based strategies and optimization-based ones. The first group can be further divided into deterministic rule-based and fuzzy rule-based EMS whilst the second group consists of global optimization and real-time optimization EMS.

Rule-based strategies are preferred mainly because of their low cost on development, rapid execution for online control, and wide applicability to the fast-changing driving conditions [12]-[14]. These controllers have shown their advantages on ICE efficiency improvement and robust operation on power mode switching but suffer the limitations of not realizing the maximal fuel efficiency or violating some system constraints.

Manuscript received October 27, 2020; revised March 20, 2021; accepted April 19, 2021. Date of publication April 27, 2021; date of current version July 8, 2021. This work was supported in part by Chinese Scholarship Council (CSC), and in part by Swedish Electromobility Centre (SEC) in the thematic area of system studies and methods. The project title is "Optimal Integration of Combustion Engines, and Electric Motors for HEVs." The work of Lei Feng was supported by KTH XPRES. The review of this article was coordinated by Prof. Hicham Chaoui. *(Corresponding author: Lei Feng.)*

Tong Liu and Lei Feng are with the Department of Machine Design, KTH Royal Institute of Technology, Brinellvägen 83, Stockholm 10044, Sweden (e-mail: tongliu@kth.se; lfeng@kth.se).

Wenyao Zhu is with the Division of Electronics and Embedded Systems, KTH Royal Institute of Technology, Electrum 229, Kista 16440, Sweden (e-mail: wenyao@kth.se).

Digital Object Identifier 10.1109/TVT.2021.3075729

[1] https://www.makethefuture.shell/en-gb/shell-eco-marathon

Global optimization methods, such as dynamic programming (DP) [15]-[17], Pontryagin's minimum principle (PMP) [18]-[20], simulated annealing [21], [22], particle swarm optimization (PSO) [23], [24], can generate a non-causal global optimal solution based on a priori information of the driving cycle. Resulting from the extremely high computation and memory overheads, these methods are usually not applicable for direct online implementation. In most cases, they are employed as benchmarks to estimate the performances of other online controllers, or as online solvers to solve an optimal control problem (OCP) with limited horizon [25], [26].

Model predictive control (MPC) [27]-[29] and equivalent consumption minimization strategy (ECMS) [30]-[32] are two typical real-time optimization methods. MPC usually relies on a planned or predicted speed trajectory as a reference and perhaps an estimated state of charge (SOC) trajectory as a constraint to balance the electricity utilization for the whole trip. In contrast, ECMS mainly focuses on seeking for the optimal equivalence factor by which the total torque demand on powertrain is well split into fuel and electric paths, and then a balance between fuel and electricity consumptions can be achieved. However, the optimal equivalence factor is generally unknown unless the entire driving cycle is known a priori.

Furthermore, the research on EMS extends from higher energy efficiency to many other topics, e.g., tailpipe emission [33], [34], vehicle drivability [35], [36], component aging [22], [37], electric charge sustainability [38], [39] as well as robust ICE on/off switches [40]. To our knowledge, the ICEs in most HEVs can provide continuous torque values within their feasible boundaries. As a result, the optimal torque split problems are solved as constrained nonlinear programming problems. For this reason, the online optimization for transient powertrain torque split usually relies on complex algorithms and thus entails heavy computing payload. Once the ICE operation can be limited to a narrow region, the online computation and memory overheads will be significantly reduced.

In [41], an explicit DP controller was developed to investigate the feasibility of such a parallel hybrid powertrain that contains two EMs and one ICE which can only work with peak efficiency or be turned off. The optimal ICE on/off status is determined by solving a binary optimization problem. Including the additional fuel consumption on ICE switching, this controller still produced better fuel economy than its counterpart with a general ICE configuration. However, this controller contains 2 look-up tables of 5 degrees of freedom (DOF) so that the total memory size is larger than 100 Mbytes while none of the commonly low-cost microprocessors can provide a flash memory larger than 2 Mbytes.

The main contribution of this paper is to develop a quasi-Pontryagin's minimum principle (Q-PMP) controller applied on the same HEV aiming to further reduce the memory overhead without obvious compromises on either fuel efficiency or computation overhead. In contrast to the normal PMP which derives the optimal costate trajectory through solving partial differential equation (PDE), this new method finds the real-time costate from the value function obtained by DP, which solves the OCP offline and provides an optimal speed trajectory along with the value function. The online Q-PMP controller takes the optimal speed trajectory as the reference, and determines the ICE on/off status and the EM torque by minimizing the Hamiltonian. Afterwards, a simple but effective secondary torque split algorithm is designed to allocate the EM torque to two EMs. The new controller relies on only 1 look-up table of 3 DOF so that the total memory overhead concerning the complete EMS decreases to merely 497 Kbytes. Therefore, a portable microprocessor *STM32L476RGT6* (1 Mbyte flash memory, 80 MHz CPU) is competent for running this control algorithm for online usage.

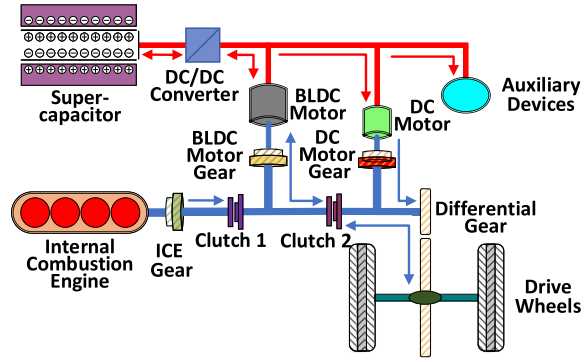


Fig. 1. Schematic Diagram of the Parallel HEV Powertrain.

Model-in-the-loop (MIL) and processor-in-the-loop (PIL) simulations for different EMS have been performed on a standard racing track. The results fully reflect the advantages of this Q-PMP controller. First, it improves the fuel efficiency by roughly 14% compared with a rule-based charge-depletion and charge-sustain (CDCS) controller, and roughly 7.5% compared with an adaptive ECMS (A-ECMS) controller. Secondly, it reaches a fuel efficiency almost identical to the explicit DP, but reduces onboard memory demand by 99.5%. In addition, the Q-PMP controller enjoys better and more robust performance than A-ECMS, and can avoid the arduous work on parameter tuning. Last but not least, the Q-PMP controller is applied to both binary and continuously controlled ICEs for the same HEV. It achieves roughly 12% higher fuel efficiency for the binary ICE with only approximately 1/3 CPU utilization.

The rest of this paper is organized as follows. Section II introduces the modeling of HEV and its powertrain. In Section III, the HEV racing problem is formulated as an OCP. The design procedure of the Q-PMP online controller is described in Section IV. Subsequently, simulation results by MIL and PIL are presented and discussed in Section V. Section VI concludes the whole paper and poses our future work.

II. Vehicle and Powertrain Modeling

Illustrated by Fig. 1, this hybrid electric racing car is a light-weight HEV and has a parallel powertrain. The essential parameters of this HEV are summarized in Table I. The unique features of this powertrain are listed below.

1) The powertrain contains 3 independent propelling components, namely an ICE fueled by gasoline, a DC motor, and a brush-less DC (BLDC) motor working as either an actuator or a generator.

TABLE I
ESSENTIAL PARAMETERS OF THE HEV

Parameter	Symbol	Value	Unit
Vehicular gross mass	M	216	kg
Lumped rotational inertial	J_t	0.66	$kg \cdot m^2$
Wheel radius	r	0.26	m
Frontal area	A_f	1.05	m^2
Air drag coefficient	c_d	0.25	/
Rolling resistance coefficient	c_r	0.011	/
Differential gear ratio	R_p	10	/
ICE gear ratio	R_{ce}	1.23	/
BLDC motor gear ratio	R_{bl}	1.05	/
DC motor gear ratio	R_{ce}	1.4	/
Average driveshaft efficiency	η_d	0.9	/
Nominal capacitance	C	107	F
Min. supercapacitor voltage	$V_{c,\min}$	40	V
Max. supercapacitor voltage	$V_{c,\max}$	50	V
Max. ICE torque	$T_{ce,\max}$	3.2	Nm
Max. ICE power	$P_{ce,\max}$	1.5	kW
Max. BLDC motor torque	$T_{bl,\max}$	2.6	Nm
Max. BLDC motor power	$P_{bl,\max}$	1.02	kW
Max. DC motor torque	$T_{dc,\max}$	0.42	Nm
Max. DC motor power	$P_{dc,\max}$	0.2	kW

2) The DC motor has higher energy efficiency but lower maximal torque than the BLDC motor.
3) A supercapacitor instead of a battery is selected as the EES thanks to its higher specific power. Therefore, the high power flow at peak times across the electric path is never a constraint.
4) The ICE is specially configured that it can only operate with peak efficiency or be turned off.
5) The ICE connects to the driveshaft through Clutch 1. Engine brake can thus be avoided by disengaging it when only EMs drive the HEV.
6) Clutch 2 can combine or separate torques from ICE and BLDC motor to drive wheels so that ICE charging can be conducted regardless of the HEV speed, even if the HEV stops.

This section presents a control-oriented dynamical model based on this specific HEV powertrain. This model is used to estimate both the fuel and electricity consumptions according to the HEV's movement and road condition.

A. Vehicle Longitudinal Dynamics

The longitudinal dynamic equations of the HEV are presented by (1)-(2).

$$M\dot{v}+\frac{J_t\dot{\omega}_t}{r} = \frac{T_t}{r}-\frac{1}{2}\rho_a A_f c_d v^2 - Mg(c_r\cos\alpha+\sin\alpha) \quad (1)$$

$$\omega_t = \frac{v}{r} \quad (2)$$

where M is the vehicular gross mass concerning both the HEV and the driver; v is the HEV longitudinal speed; ω_t is the angular speed of wheel shaft; r is the wheel radius; T_t is the net tractive torque at the wheel shaft; J_t reflects the lumped rotational inertial of the whole powertrain acting on the wheel shaft; α denotes the road slope angle whose value is dependent on y, the driving distance accumulated by v; other parameters includes the vehicle frontal area A_f, the air drag coefficient c_d, and the rolling resistance coefficient c_r.

T_t is the sum of torques from the fuel path T_f and electric path T_e, expressed by (3).

$$T_t = \begin{cases} (T_f + T_e)\cdot\eta_d\cdot R_p; & T_e \geq 0 \\ (T_f\cdot\eta_d + T_e/\eta_d)\cdot R_p; & T_e < 0 \end{cases} \quad (3)$$

where R_p is the differential gear ratio and η_d is the average value of the lumped energy efficiency on the driveshaft.

B. ICE Model

To estimate the total fuel consumption, the ICE is modeled as a quasi-static 2D energy efficiency map with its torque T_{ce} and angular speed ω_{ce} as inputs, and the efficiency η_{ce} as output, shown by Fig. 9.

Once Clutches 1 and 2 are engaged, ω_{ce} is determined by v via the ICE gear ratio R_{ce}, given as

$$\omega_{ce} = \omega_t\cdot R_p R_{ce}. \quad (4)$$

The nominal fuel mass flow rate $\dot{m}_f$ is thus defined as

$$\dot{m}_f = \frac{T_{ce}\omega_{ce}}{\eta_{ce}(T_{ce},\omega_{ce})\cdot Q_f} \quad (5)$$

where Q_f is the lower heating value of the gasoline.

Regarding the 4th feature of this powertrain, T_{ce} is not a free variable but relies on ω_{ce} to ensure peak efficiency. Hence, it is replaced by $T^*_{ce}(\omega_{ce})$. In this sense, T_f and $\dot{m}_f$ are determined by ω_{ce} and the ICE on/off status $s_{ce}\in\{0,1\}$ which is a binary variable with 1/0 representing on/off.

$$T_f = T^*_{ce}(\omega_{ce})\cdot s_{ce}\cdot R_{ce} \quad (6)$$

$$\dot{m}_f = \frac{T^*_{ce}(\omega_{ce})\cdot\omega_{ce}s_{ce}}{\eta_{ce}\left(T^*_{ce}(\omega_{ce}),\omega_{ce}\right)\cdot Q_f} \quad (7)$$

C. EM Models

Both DC and BLDC motors can provide torques, T_{dc} and T_{bl}, on the electric path through their respective gears, R_{dc} and R_{bl}. Their combination constitutes T_e.

$$T_e = T_{dc}\cdot R_{dc} + T_{bl}\cdot R_{bl} \quad (8)$$

Similar to the ICE model, quasi-static efficiency maps are introduced to analyze and calculate the electricity demand to power the two EMs, shown by Fig. 2. The currents to drive these two EMs, I_{dc} and I_{bl}, are expressed by (9)-(12).

$$I_{dc} = \frac{T_{dc}\omega_{dc}}{\eta_{dc}(T_{dc},\omega_{dc})\cdot V_{bus}} \quad (9)$$

$$I_{bl} = \begin{cases} \frac{T_{bl}\omega_{bl}}{\eta_{bl}(T_{bl},\omega_{bl})\cdot V_{bus}}; & T_{bl}\geq 0 \\ \frac{T_{bl}\omega_{bl}\cdot\eta_{bl}(T_{bl},\omega_{bl})}{V_{bus}}; & T_{bl}<0 \end{cases} \quad (10)$$

$$\omega_{dc} = \omega_t\cdot R_p R_{dc} \quad (11)$$

$$\omega_{bl} = \omega_t\cdot R_p R_{bl} \quad (12)$$

where ω_{dc} and ω_{bl} denote the rotational speeds of the DC and BLDC motors; η_{dc} and η_{bl} represent their own net efficiencies. When the amount of electric charge in the supercapacitor varies, the terminal voltage V_c also fluctuates; however, the EMs and other auxiliary devices require a stable voltage source. Hence, a

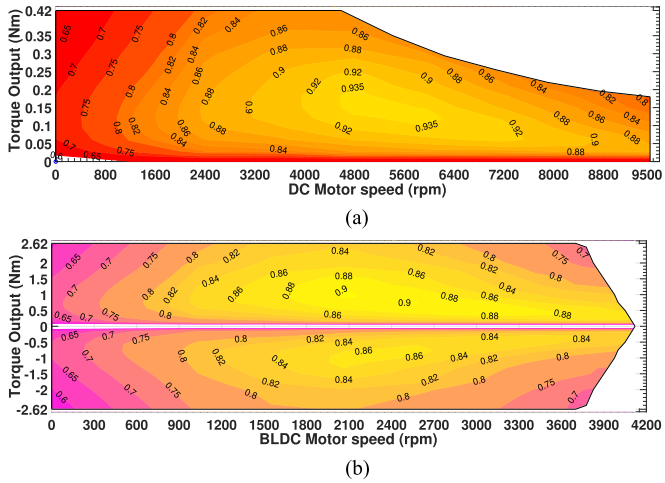


Fig. 2. EM Efficiency Maps. (a) DC Motor. (b) BLDC Motor.

DC/DC converter is employed to maintain a constant bus voltage V_{bus} for the electronics.

D. EES Model

As mentioned at the beginning of Section II, a supercapacitor is employed as the EES to support all the onboard electric appliances running during the whole race. For simplification, taking the average value of the electric power to all auxiliary devices P_{aux} into consideration, the EES dynamics can be illustrated by (13)-(16).

$$P_e = (I_{dc} + I_{bl}) \cdot V_{bus} + P_{aux} \tag{13}$$

$$I_c = \begin{cases} P_e/(V_c \cdot \eta_c); & P_e \geq 0 \\ P_e \cdot \eta_c/V_c; & P_e < 0 \end{cases} \tag{14}$$

$$\dot{V}_c = -I_c/C \tag{15}$$

$$SOC = V_c \cdot C/Q_0 \tag{16}$$

where P_e is the net power consumption on the electric path; I_c is the transient current flowing through the supercapacitor; η_c is the lumped efficiency including both the supercapacitor and the DC/DC converter; C and Q_0 denote the nominal capacitance and charge capacity of the supercapacitor; V_c is used as the indicator to the SOC due to their linear correlation.

Based on the HEV model defined above, the total fuel consumption M_f during a complete driving period t_e can be calculated by (17).

$$M_f = \int_0^{t_e} \dot{m}_f(\tau)\, d\tau \tag{17}$$

III. Optimal Control Problem Formulation

The HEV is developed for the SEM competition. According to the racing rules, each HEV must finish a given number of laps of a circular racing track within a time limit. The racing track consists of uphills, downhills, and curves. After each lap, the HEV must stop completely and the ICE should be switched off. At the end of the last lap, if the EES SOC is lower than its initial value, the ICE must be switched on again to recharge the EES until its SOC fully recovers. The total fuel consumption is recorded as the only index to grade an HEV's performance. On these grounds, this section formalizes such an HEV racing problem as an OCP. To reduce the complexity of this OCP, a model reduction method is presented to merge the two EMs into one.

A. Problem Formulation

The OCP is formulated in discrete-distance rather than discrete-time format because of the following two reasons.

1) The total driving time t_e is not a constant but a variable with an upper bound.
2) The road slope angle is dependent on the position along the track and available a priori, given as $\alpha(y)$.

The driving distance y is taken as the free variable and the driving time to reach y as an additional state variable $t(y)$. The complete driving track of distance Y is evenly divided into N stages of even distance $\Delta y = Y/N$, and $t(y)$ can be approximated by $t(k)$ if y is within the range $[k \cdot \Delta y, (k+1) \cdot \Delta y)$ with $k \in \{0, 1, \ldots, N-1\}$.

With a view to the possible fuel consumption to recharge the EES after racing, the terminal cost to the OCP can be expressed by (18).

$$m_{rc}(N) = \begin{cases} \frac{C \cdot (V_c^2(0) - V_c^2(N))}{2\eta_{rc} Q_f}; & V_c(N) < V_c(0) \\ 0; & V_c(N) \geq V_c(0) \end{cases} \tag{18}$$

where η_{rc} is the energy conversion efficiency from the fuel tank to the supercapacitor. During the final recharging, the HEV stops at the roadside, Clutch 1 is engaged and Clutch 2 is disengaged. Therefore, η_{rc} is estimated as a constant.

To avoid frequent ICE on/off switches, the extra energy consumption and time delay for each switch are introduced into the HEV model. In reality, the time delay, from the instant when the controller sends out the command to the instant when the switch is completely finished and Clutch 1 is fully engaged or disengaged, varies a lot under different operation scenarios. For a reasonable simplification, it is supposed that the ICE delay is one step in the discrete model. Therefore, at each stage, the potential equivalent fuel consumption on ICE on/off switch m_{sw} can be calculated as below.

$$s_{ce}(k+1) = u_{ce}(k) \tag{19}$$

$$m_{sw}(k) = \begin{cases} 0; & u_{ce}(k) = s_{ce}(k) \\ m^\star; & u_{ce}(k) \neq s_{ce}(k) \end{cases} \tag{20}$$

where $u_{ce} \in \{0, 1\}$ is the binary control command on ICE on/off status, and $m^\star$ is the necessary equivalent fuel consumption for once ICE on/off switch.

The discrete-distance nonlinear state-space model of this HEV is shown in the following. For all $k \in \{0, 1, 2, \ldots, N-1\}$,

$$\begin{cases} \mathbf{x}(k+1) = F(\mathbf{x}(k), \mathbf{u}(k)) \\ \mathbf{x} = [v \quad t \quad V_c \quad s_{ce}]^T \\ \mathbf{u} = [T_{dc} \quad T_{bl} \quad u_{ce}]^T. \end{cases} \tag{21}$$

The system dynamics is explicitly written as below.

$$v(k+1) = \sqrt{v^2(k) + 2\dot{v}(k)\Delta y} \tag{21a}$$

$$t(k+1) = t(k) + \Delta t(k) \tag{21b}$$

$$V_c(k+1) = V_c(k) + \dot{V}_c(k)\Delta t(k) \tag{21c}$$

$$\Delta t(k) = \frac{2\Delta y}{v(k) + \sqrt{v^2(k) + 2\dot{v}(k)\Delta y}} \tag{21d}$$

$$\mathbf{x}(0) = [0 \quad 0 \quad V_{c,\text{ini}} \quad 0]^T \tag{21e}$$

where Δt denotes the update of the driving time during one stage, $\mathbf{x}(0)$ represents the initial values of the state variables among which only V_c has a nonzero initial value $V_{c,\text{ini}}$. The expressions of $\dot{v}$ and $\dot{V}_c$ are given by (1)-(15) in Section II.

Then, the optimization objective J can be derived.

$$J(\mathbf{x}(0)) = \sum_{k=0}^{N-1} [\dot{m}_f(\mathbf{x}(k), \mathbf{u}(k))\Delta t(k) + m_{sw}(\mathbf{x}(k), \mathbf{u}(k))] + m_{rc}(\mathbf{x}(N)) \tag{22}$$

subject to (21) and the following,

$$0 \le v(k) \le v_{\max} \tag{22a}$$

$$0 \le t(k) \le t_{\max} \tag{22b}$$

$$V_{c,\min} \le V_c(k) \le V_{c,\max} \tag{22c}$$

$$V_{c,\text{tmin}} \le V_c(N) \le V_{c,\max} \tag{22d}$$

$$0 \le T_{dc}(k) \le T_{dc,\max}(v(k)) \tag{22e}$$

$$T_{bl,\min}(v(k)) \le T_{bl}(k) \le T_{bl,\max}(v(k)) \tag{22f}$$

where the subscripts $\min$ and $\max$ indicate the lower and upper bounds of each corresponding variable; $V_{c,\text{tmin}}$ denotes a special lower bound for V_c at the end of driving, which is close to $V_{c,\text{ini}}$ but much larger than $V_{c,\min}$. The torque boundaries of the two EMs are dependent on their rotational speeds ω_{dc} and ω_{bl} which are further dependent on v.

B. Model Reduction

A straightforward solution to the OCP (22) is to apply DP. The challenge to this is the excessively computational complexity, because the HEV model has 4 state and 3 control variables. To reduce its complexity, we merge two independent control variables T_{dc} and T_{bl} into one T_e as defined in (8), which represents the combined torque from the electric path. The maximal value of T_e is $T_{e,\max} = T_{bl,\max}R_{bl} + T_{dc,\max}R_{dc}$, because both EMs can provide drive torques simultaneously. The minimal value of T_e is $T_{e,\min} = T_{bl,\min}R_{bl}$, because only the BLDC motor can work as a generator.

$$T_{e,\min} \le T_e \le T_{e,\max} \tag{23}$$

For a positive T_e lower than $T_{e,\max}$, there are infinitely possible combinations of T_{dc} and T_{bl} in principle. To seek for the best one with maximal electrical efficiency, the general idea is to solve an optimization problem based on the two EM models.

In actuator mode, the lumped EM efficiency η_e is defined as the total output mechanical power divided by the total input electrical power, and the selection of T_{dc} should maximize η_e.

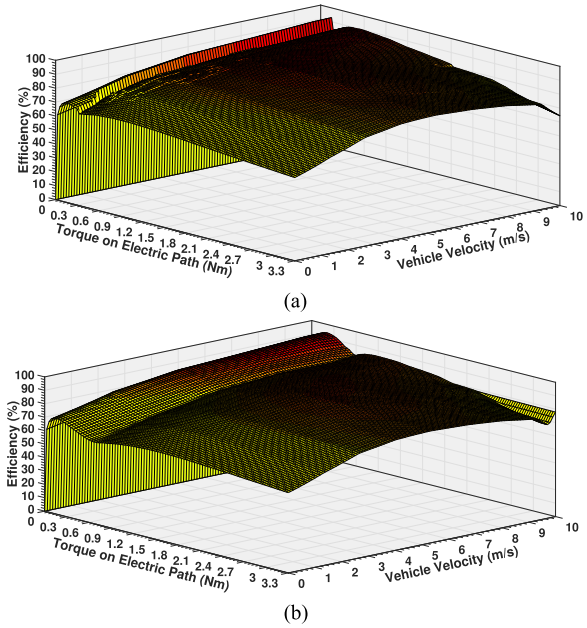


Fig. 3. Lumped EM Efficiency Map in Actuator Mode. (a) Explicit Optimization. (b) Simplified Algorithm.

$$\eta_e(T_{dc}, T_{bl}) = \frac{T_e \cdot \omega_t R_p}{(I_{dc} + I_{bl}) \cdot V_{bus}} = \frac{T_e}{\dfrac{T_{dc} \cdot R_{dc}}{\eta_{dc}(T_{dc}, \omega_{dc})} + \dfrac{T_{bl} \cdot R_{bl}}{\eta_{bl}(T_{bl}, \omega_{bl})}} \tag{24}$$

$$T_{dc}^* = \arg\max_{T_{dc}} \eta_e(T_{dc}, T_{bl}) \tag{25}$$

subject to

$$T_{bl} = (T_e - T_{dc}R_{dc})/R_{bl} \tag{25a}$$

$$0 \le T_{dc} \le T_{dc,\max}(v) \tag{25b}$$

$$0 \le T_{bl} \le T_{bl,\max}(v) \tag{25c}$$

When the optimal solution T_{dc}^* equals to T_e/R_{dc}, T_e is completely provided by the DC motor; conversely, when it equals to 0, DC motor is unused and only the BLDC motor provides T_e; otherwise, both EMs work together to satisfy T_e. At any time when T_{dc}^* is determined, the corresponding value of T_{bl}^* can be calculated by (25a). The optimal lumped EM efficiency η_e^* over all the admissible positive T_e and v are depicted by Fig. 3(a).

This optimization-based method can assure an optimal decision on positive torque split between two EMs at the expense of solving a nonlinear optimization problem. To reduce the online workload, a viable alternative is to solve the optimization problem in advance and save the solutions in a 2D look-up table, but this method will consume extra memory space of the onboard microprocessor. For that reason, this subsection proposes a simple but effective algorithm to find a close-to-optimal solution without solving the optimization problem (25). This algorithm saves both the computation time and memory overhead for an embedded system application.

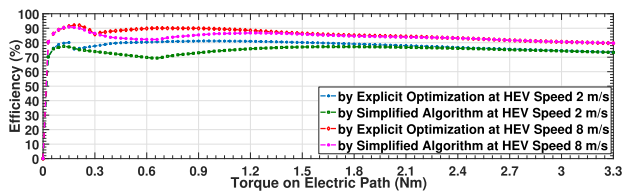


Fig. 4. Lumped EM Efficiencies at Specific HEV Speed.

The simplified algorithm exploits the 2nd powertrain feature that the DC motor has higher energy efficiency than the BLDC motor. Hence, the DC motor should be preferred to drive the HEV, and the BLDC motor is invoked only if $T_{dc,\max}$ is insufficient to satisfy T_e or T_e of negative value is in request. Under the constraint (23), T_{dc} is expressed by (26) and T_{bl} then follows (25a). By this way, η_e in actuator mode can be rapidly calculated by substituting T_{dc} and T_{bl} into (24), its values over all the admissible positive T_e and v are illustrated by Fig. 3(b).

$$T_{dc} = \begin{cases} 0; & T_e \leq 0 \\ T_{dc,\max}; & T_e \geq T_{dc,\max} R_{dc} \\ T_e/R_{dc}; & otherwise \end{cases} \tag{26}$$

To verify the accuracy of the simplified algorithm, η_e by both methods are compared in details. For safety concern, $v_{\max}$ is limited to $10\,m/s$. Within this range, both EMs can operate on their maximal torques so that the combined maximal torque is $3.3\,Nm$. To ensure the high precision of the efficiency maps, the test points are selected by uniform sampling with tiny intervals, $0.03\,Nm$ in T_e and $0.1\,m/s$ in v. Consequently, there are more than 10 000 test points in each 2D rectangular region of Figs. 3(a) and 3(b).

The root-mean-square error (RMSE) of all test points between two maps is 3.58% with a maximal error of 12%. The large errors are primarily located at the low power region when v is less than $3\,m/s$ and T_e lower than $1.5\,Nm$. The deviation is illustrated by Fig. 4 which compares η_e by two methods for different T_e when v is $2\,m/s$. Nevertheless, this region is barely used in practice because v is larger than $7\,m/s$ in most of the driving period to meet the time limitation for the race. Small values of v only exist at the beginning and ending phases of each lap. Furthermore, the average values of η_e of the whole maps in Figs. 3(a) and 3(b) are 81.2% and 79.6% respectively. Whereas, those values at the most used region, when v is larger than $7\,m/s$, are 83.5% and 83.0%. Fig. 4 also depicts η_e by two methods when v is $8\,m/s$. It can be seen that η_e at the high speed regions are higher than those at low speed regions, and their difference becomes negligible, especially when T_e is larger than $2\,Nm$. More importantly, in spite of the minor numeric deviation, the simplified algorithm greatly raises the computation efficiency.

With this simplification, T_{dc} and T_{bl} can be properly replaced by T_e so that the original OCP (22) is simplified to have only 2 control variables. The relevant expressions of system dynamics are updated as below.

$$\mathbf{u} = [T_e \quad u_{ce}]^T \tag{27}$$

$$P_e = I_e \cdot V_{bus} + P_{aux} \tag{28}$$

$$I_e = \begin{cases} (T_e\omega_t \cdot R_p)/(\eta_e \cdot V_{bus}); & T_e \geq 0 \\ (T_e\omega_t \cdot \eta_e \cdot R_p)/V_{bus}; & T_e < 0 \end{cases} \tag{29}$$

where I_e is the sum of currents flowing across two EMs.

IV. Q-PMP Online Controller Design

After formulating the OCP with a simplified HEV model, DP is employed to search for the optimal control sequence minimizing the objective (22). Owing to the curse of dimensionality, DP cannot be selected as the online EMS running on a low-cost microprocessor. To address this issue, this section proposes a computationally efficient EMS which combines the merits of both DP and PMP, and achieves a close-to-optimal fuel economy by a small amount of computation resources.

The essential idea to develop such an EMS is to decompose a complicated OCP into several sequentially correlated sub-problems. The solution to one sub-problem serves as the reference to others. Thus, the subsequent ones will have lower complexities and their solving processes require much less computation resources. Eventually, a suboptimal solution to the original OCP can be obtained. More importantly, these sub-problems can be solved online in real-time by an affordable microprocessor. In this case, the OCP (22) is divided into 4 sub-problems which successively reflect different optimization assignments. They are 1) finding the optimal HEV speed trajectory; 2) estimating the optimal costate of PMP; 3) determining the optimal ICE on/off status and EM torque; and 4) allocating the EM torque to DC and BLDC motors. Among them, sub-problem 1) is solved only once offline in advance with the prior knowledge of the racing track. Its solution is used as references for sub-problems 2) and 3) which are periodically solved online in real time. Sub-problem 4) is also iteratively solved online relying on the solution to sub-problem 3). The complete EMS architecture including both offline and online control functions is depicted by Fig. 5.

In Fig. 5, the three control levels, namely offline DP, online Q-PMP and actuator control, are covered by yellow, skyblue and magenta blocks from top to down. The offline DP finds the optimal HEV speed trajectory and the optimal value function. The online Q-PMP consists of three modules: the Speed Regulator regulates the HEV to follow the optimal speed trajectory by calculating the total torque demand on the powertrain; the Primary Torque Splitter determines the ICE on/off status and the supplementary torque from the electric path by PMP whose costate is derived from the value function; and the Secondary Torque Splitter further determines the torques from the DC and BLDC motors. The actuator control comprises three independent control units in parallel for implementing the online Q-PMP optimization decisions on ICE and two EMs. This paper focuses on the development of the control functions at top and middle levels, and the actuator control follows the standard approaches.

All signals connecting the control modules are classified into four types: the orange lines indicate the prior knowledge and the solutions from offline DP which are stored in the microprocessor; the blue ones express the real-time control instructions from online Q-PMP; the black ones reflect the real-time system states

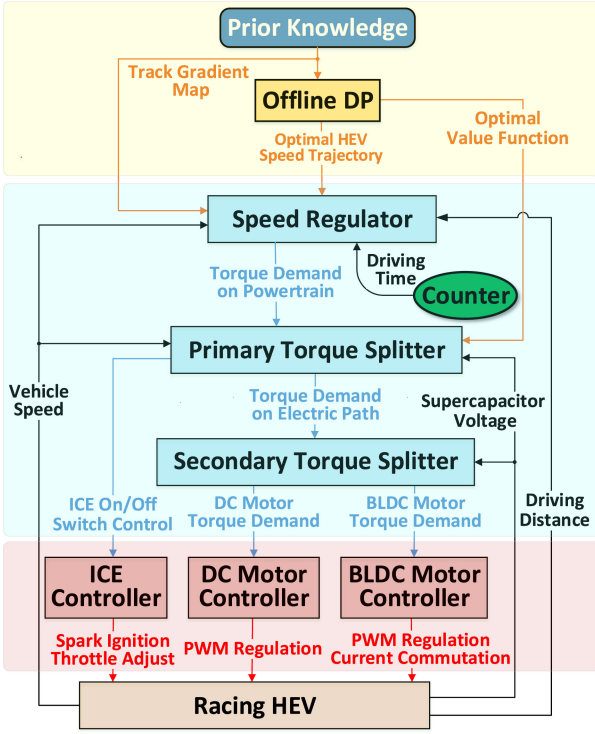


Fig. 5. Control Architecture of the HEV Racing Car.

TABLE II
VARIABLE DISCRETIZATION FOR DP

Symbol	Unit	Min. Value	Max. Value	Interval
v	m/s	0	10	0.5
t^1	s	$t_{\min}(k)$	$t_{\max}(k)$	1
V_c	V	40	50	0.2
s_{ce}	/	0	1	1
T_e	Nm	-2.6	3.3	0.1
u_{ce}	/	0	1	1
y	m	0	Y	4.488[2]

[1]The bounds for t here are dynamic values. [2]This value depends on the track SEM Europe 2016.

detected by the vehicular onboard sensors; and the red ones represent the actuator control instructions acting on corresponding physical components.

A. Offline DP Solution

This first step to DP is to discretize all continuous state and control variables. The resolution must be properly selected to balance the control performance and the computation load. Table II outlines the details of variable discretization. Since the variable t is monotonically increasing, dynamic bounds rather than fixed values are set for each step [41]. The step size of the free variable y varies from 4 to 5 m. Its exact value is determined by Y on different racing tracks.

The complete solving process takes nearly 8 hours in a high-performance desktop (Intel Core i7-8700 CPU, 3.7 GHz, 64 GB RAM), and generates 2 look-up tables of 5 DOF for the two control inputs in (27). The total size of the two tables exceeds 100 Mbytes. The tables are too large to be stored in a low-cost microprocessor.

The distance-based optimal HEV speed and driving time trajectories $v^*(k)$ and $t^*(k)$ $(k \in \{0, 1, \ldots, N\})$ can be picked up from the DP solutions. By combining them together, we can construct a time-based trajectory $v^*(t)$ as the reference for the Speed Regulator. Besides, a 5D look-up table of value function $V^*(v, t, V_c, s_{ce}, y)$ is included in the DP solution. Along the optimal trajectories of v^* and t^*, a 3D look-up table can be extracted out, denoted as $Y_y^*(V_c, s_{ce}) := V^*(v^*, t^*, V_c, s_{ce}, y)$, and saved into the onboard processor for calculating the optimal costate by the Primary Torque Splitter.

B. Speed Regulator

Located at the first layer of the online Q-PMP hierarchy, the Speed Regulator is mainly responsible for calculating T_t by which the HEV attempts to track $v^*(t)$.

For online control, periodical control mechanism is adopted with a fixed time interval t_s. Evenly divide the optimal total driving time $t^*(N)$ into Z steps by t_s, at the jth step $(j \in \{0, 1, 2, \ldots, Z-1\})$, $T_t(t_j)$ is determined by the real-time $\alpha(t_j)$, $v(t_j)$ and $\dot{v}(t_j)$ according to (1) and (2). $\alpha(t_j)$ is acquired by the track gradient map, $v(t_j)$ is detected by an onboard speed sensor, and $\dot{v}(t_j)$ is chosen to ensure that at next time step $v(t_{j+1})$ is equivalent to its optimal value $v^*(t_{j+1})$.

$$\dot{v}(t_j) = \left(v^*(t_{j+1}) - v(t_j)\right)/t_s \tag{30}$$

This method, however, is sensitive to the sensor noise of v in practice and numerically unstable owing to the small value of t_s. For that reason, a PID controller is employed instead to calculate $\dot{v}$, expressed as below.

$$\dot{v}^\star(t_j) = k_p \Delta v(t_j) + k_i \sum_{i=0}^{j} \Delta v(t_i) + k_d \left(\Delta v(t_j) - \Delta v(t_{j-1})\right) \tag{31}$$

$$\Delta v(t_j) = v^*(t_{j+1}) - v(t_j) \tag{32}$$

$$\dot{v}(t_j) = \begin{cases} \dot{v}_{\min}; & \dot{v}^\star(t_j) < \dot{v}_{\min} \\ \dot{v}_{\max}; & \dot{v}^\star(t_j) > \dot{v}_{\max} \\ \dot{v}^\star(t_j); & \text{otherwise} \end{cases} \tag{33}$$

where Δv is the deviation of v to its optimal value at next time step; $\dot{v}^\star$ is an intermediate result from the PID controller and may be truncated by the lower and upper bounds of the HEV acceleration for safety and feasibility concerns; k_p, k_i and k_d are three tunable parameters of the PID controller.

C. Primary Torque Splitter

Receiving T_t from the Speed Regulator, the Primary Torque Splitter splits it into fuel and electric paths by finding corresponding u_{ce} and T_e according to Q-PMP.

Assume that the HEV can perfectly follow $v^*(t)$ and the corresponding T_t is received from the Speed Regulator, neither v nor t is a free state variable to the OCP (22), and T_e is no longer a free control variable as well. Therefore, the HEV model and the OCP can be further simplified and then transformed to discrete-time format for online control design. The simplified

model is expressed by (34). For all $j \in \{0, 1, 2, \ldots, Z-1\}$,

$$\begin{cases} \bar{\mathbf{x}}(t_{j+1}) = \bar{F}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j)\right) \\ \bar{\mathbf{x}} = [V_c \quad s_{ce}]^T \\ \bar{\mathbf{u}} = u_{ce} \end{cases} \tag{34}$$

where the superscript $^-$ indicates the simplified variables and functions in the new model. The system dynamics and initial values of the state variables are given below.

$$V_c(t_{j+1}) = V_c(t_j) + \dot{V}_c(t_j) \cdot t_s \tag{34a}$$

$$s_{ce}(t_{j+1}) = u_{ce}(t_j) \tag{34b}$$

$$\bar{\mathbf{x}}(t_0) = [V_{c,\text{ini}} \quad 0]^T \tag{34c}$$

The cost function to be optimized is updated as (35), and the constraints (3), (22c)-(22f) still hold.

$$J\left(\bar{\mathbf{x}}(t_0)\right) = \sum_{j=0}^{Z-1} [\dot{m}_f\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j)\right) \cdot t_s + m_{sw}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j)\right)] + m_{rc}\left(\bar{\mathbf{x}}(t_Z)\right) \tag{35}$$

To solve this OCP by PMP, we define the Hamiltonian as,

$$\begin{aligned} &\mathcal{H}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j), \boldsymbol{\lambda}(t_{j+1})\right) \\ &= \dot{m}_f\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j)\right) \cdot t_s + m_{sw}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j)\right) \\ &\quad + \lambda_1(t_{j+1}) \cdot V_c(t_{j+1}) + \lambda_2(t_{j+1}) \cdot s_{ce}(t_{j+1}) \end{aligned} \tag{36}$$

where $\boldsymbol{\lambda} = [\lambda_1 \quad \lambda_2]^T$ is the costate vector.

The optimal control u_{ce}^* is given by minimizing the Hamiltonian $\mathcal{H}(\bar{\mathbf{x}}^*(t_j), \bar{\mathbf{u}}^*(t_j), \boldsymbol{\lambda}^*(t_{j+1}))$ with respect to

$$V_c^*(t_{j+1}) = \frac{\partial \mathcal{H}(\cdot)}{\partial \lambda_1(t_{j+1})} \tag{37}$$

$$s_{ce}^*(t_{j+1}) = \frac{\partial \mathcal{H}(\cdot)}{\partial \lambda_2(t_{j+1})} \tag{38}$$

$$\lambda_1^*(t_j) = \frac{\partial \mathcal{H}(\cdot)}{\partial V_c(t_j)} \tag{39}$$

$$\lambda_2^*(t_j) = \frac{\partial \mathcal{H}(\cdot)}{\partial s_{ce}(t_j)}. \tag{40}$$

The challenge of applying PMP is to find the trajectories of the optimal costate $\boldsymbol{\lambda}^*$. This is normally realized through solving the recursive equations (39) and (40). The unknown values of $\boldsymbol{\lambda}^*$ at the boundaries are iteratively searched by the shooting method [18] or an optimization algorithm [21]. This paper, however, avoids this complex procedure and exploits the relationship between $\boldsymbol{\lambda}^*$ and the value function achieved by DP. This is one reason that our method is called quasi-PMP (Q-PMP). Another reason will be explained later.

In optimal control theory, the Hamiltonian-Jacobi-Bellman (HJB) equation provides a necessary and sufficient condition for an optimal control to a dynamical system, and its solution, the value function, gives the optimal cost-to-go [5]. By Bellman's principle of optimality [42], along the optimal trajectory $\bar{\mathbf{x}}^*$, the HJB equation can be derived as

$$Y_{y(t_j)}^*\left(\bar{\mathbf{x}}(t_j)\right) - Y_{y(t_{j+1})}^*\left(\bar{\mathbf{x}}(t_{j+1})\right) = \min_{\bar{\mathbf{u}}} \mathcal{H}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j), \boldsymbol{\lambda}^*(t_{j+1})\right) \tag{41}$$

$$\boldsymbol{\lambda}^*(t_j) = \left.\frac{\partial Y_{y(t_j)}^*\left(\bar{\mathbf{x}}(t_j)\right)}{\partial \bar{\mathbf{x}}(t_j)}\right|_{\bar{\mathbf{x}}^*(t_j)} \tag{42}$$

where $Y_{y(t_j)}^*(\bar{\mathbf{x}}(t_j))$ is the optimal value function at the current state vector $\bar{\mathbf{x}}(t_j)$.

In (42), $\boldsymbol{\lambda}^*$ is calculated through the PDE of value function. However, the second state variable s_{ce} is a binary variable which is non-derivable. We have to approximate λ_2^* by the difference of value function at $s_{ce} = 1$ and $s_{ce} = 0$. That is the second reason that this method is named Q-PMP.

As presented in Section IV-A, the optimal value function along the optimal speed trajectory is stored in the 3D look-up table $Y_y^*(\bar{\mathbf{x}})$. Therefore, $\boldsymbol{\lambda}^*$ can be solved as below.

$$\lambda_1^*(t_j) = \begin{cases} \lambda_1^{-*}(t_j); \ T_e(t_{j-1}) \geq 0 \\ \lambda_1^{+*}(t_j); \ T_e(t_{j-1}) < 0 \end{cases} \tag{43}$$

$$\lambda_1^{-*}(t_j) = \frac{Y_{y(t_j)}^*\left(V_c(t_j) - \epsilon, s_{ce}(t_j)\right) - Y_{y(t_j)}^*\left(V_c(t_j), s_{ce}(t_j)\right)}{-\epsilon} \tag{44}$$

$$\lambda_1^{+*}(t_j) = \frac{Y_{y(t_j)}^*\left(V_c(t_j) + \epsilon, s_{ce}(t_j)\right) - Y_{y(t_j)}^*\left(V_c(t_j), s_{ce}(t_j)\right)}{\epsilon} \tag{45}$$

$$\lambda_2^*(t_j) = \frac{Y_{y(t_j)}^*\left(V_c(t_j), 1 - s_{ce}(t_j)\right) - Y_{y(t_j)}^*\left(V_c(t_j), s_{ce}(t_j)\right)}{1 - 2s_{ce}(t_j)} \tag{46}$$

λ_1^* has two values corresponding to charging or discharging the supercapacitor during the previous control stage. T_e is determined by (3) and (6). ϵ is a positive constant representing a small deviation on V_c. λ_2^* reflects the change to the value function if the ICE on/off status is switched.

According to PMP, the optimal control u_{ce}^* can be found by minimizing the Hamiltonian in (36).

$$u_{ce}^*(t_j) = \underset{u_{ce}(t_j)}{\operatorname{argmin}}\, \mathcal{H}\left(\bar{\mathbf{x}}(t_j), \bar{\mathbf{u}}(t_j), \boldsymbol{\lambda}^*(t_{j+1})\right) \tag{47}$$

Recall that u_{ce} is a binary variable, for a given T_t, the optimal control decision becomes a binary optimization problem that whether ICE switching (or keeping) on or off can give rise to a lower $\mathcal{H}(\cdot)$ based on current V_c, s_{ce} and y.

$$u_{ce}^*(t_j) = \begin{cases} 1; & \mathcal{H}(t_j)|_{u_{ce}=1} < \mathcal{H}(t_j)|_{u_{ce}=0} \\ 0; & \mathcal{H}(t_j)|_{u_{ce}=1} > \mathcal{H}(t_j)|_{u_{ce}=0} \\ s_{ce}(t_j); & \mathcal{H}(t_j)|_{u_{ce}=1} = \mathcal{H}(t_j)|_{u_{ce}=0} \end{cases} \tag{48}$$

If the values of the Hamiltonian are identical for both on and off ICE commands, the control decision is to keep the current status to reduce the number of switches for better HEV drivability and longer ICE lifespan.

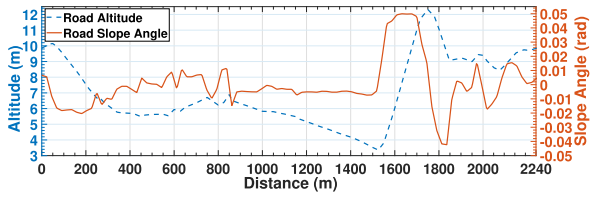


Fig. 6. Topographic Information of SEM Europe 2016.

To conclude, this Q-PMP controller highly relies on the value function to estimate the optimal costate $\boldsymbol{\lambda}$ and then determines the optimal operation on ICE status u_{ce}. Regarding the 3D look-up table of value function takes up only roughly 400 Kbytes onboard memory space, the total memory overhead for the complete EMS could be maintained to a designated amount so that a lot of portable microprocessors with limited flash memory can be selected to execute this EMS.

D. Secondary Splitter

To reduce the complexity of the original OCP (22), two EMs are merged into one in Section III-B. Given u_{ce}^* at current time interval is determined, T_e is known as well and ought to be split again into T_{dc} and T_{bl} by (26) and (25a) for implementation on real EMs.

Up to now, the online Q-PMP control design is completed. These optimal control decisions on each propelling component, namely u_{ce}, T_{dc} and T_{bl}, will be sent to their respective actuator controllers at the bottom level of the EMS architecture for further execution.

V. Test Results

This section firstly presents the performance of the Q-PMP controller through simulations with the data of a standard SEM racing track. To show the advantages of this proposed controller, its performance is compared with those by several state of the art methods. These advantages are further verified by PIL tests, which particularly prove that this controller can perform the control task in real-time on a resource-constrained microprocessor. The sensitivity of this controller with respect to modeling errors, sensor noises and process disturbances is also studied by Monte Carlo simulations. Finally, we show that the binary controlled ICE (BC-ICE) achieves higher fuel efficiency than the continuously controlled ICE (CC-ICE).

A. Racing Track

A standard racing track, SEM Europe 2016, is selected to design and test all relevant EMS. The topographic information, including the road altitude and slope angle, is depicted by Fig. 6. This track contains a polygon trajectory of 2240 meters long and every HEV is required to finish 8 laps within 43 minutes. To reflect the real driving scenarios, this track contains several uphills and downhills of varying slope angles. Steering occurs after approximately every 300 meters and lasts for roughly 20 meters. During these periods, it is assumed that c_r increases by 20% in average. In addition, a compulsory stop with ICE switched off is required at the end of each lap. Owing to the repetitive movement of this HEV, only one lap race is tested in simulation. To compensate the extra driving distance by turns during the lap, an extra 1% distance margin is added to the racing track. Furthermore, to prevent the HEV from violating the time constraint, a time margin of 2 minutes for recharging EES by ICE after driving should be subtracted from the maximally total driving time 43 minutes. Consequently, this HEV shall finish 2262 meters within 308 seconds.

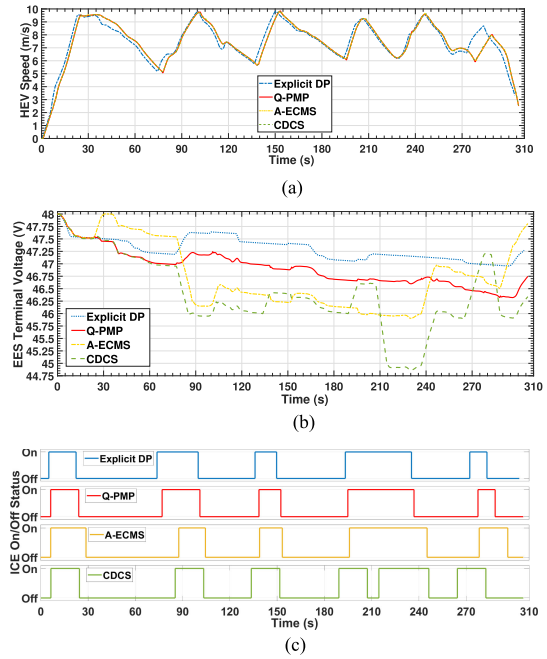


Fig. 7. MIL Simulation Results by Different EMS. (a) HEV Speed Trajectories. (b) EES Terminal Voltage Trajectories. (c) ICE On/Off Status Trajectories.

B. MIL Simulation Results

To evaluate the Q-PMP controller, it is compared with three state of the art control methods developed for the same HEV, namely explicit DP, A-ECMS, and CDCS. The primary properties of these EMS are briefly explained in Appendix.

MIL simulation is firstly carried out on MATLAB/Simulink to test the performance of the proposed Q-PMP controller and verify its advantages to other benchmark control methods. Fig. 7(a) depicts the HEV speed trajectories under four different control methods on the specified racing track. Evidently, all controllers speed up the HEV from standstill to a close-to-maximal speed at beginning and then maintain it a fairly high speed until the HEV approaches to the end. To satisfy the time constraint, the average speed is nearly $7.5\,m/s$. Further observation shows that the trajectories generated by Q-PMP, A-ECMS and CDCS are slightly different to that by the explicit DP but visually indistinguishable from each another of themselves. The reason is that, these three controllers employ the same Speed Regulator to track the same optimal speed trajectory whereas the explicit DP does not follow a fixed trajectory but calculates the ICE status and the lumped EM torque by checking 2 independent look-up tables.

TABLE III
EQUIVALENT FUEL EFFICIENCY BY DIFFERENT EMS

EMS	Explicit DP	Q-PMP	A-ECMS	CDCS
Driving Time (s)	303.8	306.4	306.4	306.3
Fuel Consumption on Lap (mL)	9.79	9.57	11.01	10.79
Final EES Voltage (V)	47.27	46.75	47.81	46.34
ICE On/Off (couple)	5	5	5	6
Total Fuel Consumption (mL)	10.25	10.36	11.13	11.84
Fuel Efficiency (km/L)	220.65	218.43	203.25	191.11

The major differences among Q-PMP, A-ECMS and CDCS are the principles to split torque between fuel and electric paths. Although their torque demands on powertrain are similar during most of the driving time, their supercapacitor voltage trajectories apparently differ from each other, shown by Fig. 7(b). For safe and efficient utility, the initial voltage before the race is set as $48\,V$, and the final voltage after one lap is expected to be no less than $46\,V$. All controllers can satisfy these constraints because the trajectories are always above $44\,V$ and all final values are larger than $46\,V$. The trajectory by Q-PMP resembles the one by the explicit DP because both of them have relatively small variations. In contrast, the other two by A-ECMS and CDCS vary in a larger range. Larger variations in supercapacitor voltage imply active and deep usage of the onboard EES; however, this is not the most efficient approach to reduce fuel consumption. This point will be elaborated in Table III.

Another focus point on MIL simulation is the analysis on ICE on/off status. The ICE on/off status influences not only the fuel consumption but also the ICE lifespan and the HEV drivability. Fig. 7(c) summarizes the ICE on/off trajectories by four EMS. All these methods can effectively manage ICE with a small number of switches: the proposed Q-PMP switches the ICE on/off 5 couples in total, identical to the explicit DP and A-ECMS, and 1 couple less than CDCS. In particular, there is no visible difference between the trajectories by Q-PMP and the explicit DP. Regarding their HEV speed trajectories are similar and the ICE is always operated at the speed coupled peak efficiency point, Q-PMP realizes an approximate control to the explicit DP on ICE utilization. By comparing Figs. 7(a) and 7(c), it can be found that, in most instances, the moments that the ICE is switched on/off by Q-PMP are synchronized with the starts of evident HEV accelerations/decelerations.

Combining all the information together, the fuel efficiency, measured by km/L, can be calculated and presented by Table III. The total fuel consumption for one lap is the sum of the fuel consumed during the lap and the fuel to recharge the supercapacitor afterwards. The equivalent fuel efficiency is defined as the total fuel consumption divided by the driving distance. The first row of Table III shows all EMS can control the HEV to complete the driving task before the prescribed time limit. Q-PMP reaches an equivalent fuel efficiency of $218.43\,km/L$, which is just a little bit less than $222.65\,km/L$ from the explicit DP but notably higher than those from A-ECMS and CDCS by 7.5% and 14.3% respectively.

MIL simulation has proved that the proposed Q-PMP controller outperforms A-ECMS and CDCS, and approximates to explicit DP in terms of the equivalent fuel efficiency. Moreover, the most significant advantage of Q-PMP compared to explicit DP is the capability of being implemented on a low-cost microprocessor and also meeting the real-time requirement. The comparison on the usage of computation resource is presented via PIL simulation in the next subsection.

TABLE IV
PIL RESULTS FOR DIFFERENT EMS

EMS	Explicit DP	Q-PMP	A-ECMS	CDCS
Fuel Efficiency (km/L)	221.68	213.76	197.68	187.89
Flash Memory Size (Kbyte)	111021	496.60	192.79	76.25
Maximum Running Time per Step (ms)	×	13.35	11.85	0.53
Average Running Time per Step (ms)	×	4.71	5.04	0.49
Average CPU Utilization (%)	×	4.71	5.04	0.49

C. PIL Simulation Results

This subsection verifies that the Q-PMP controller can run on a microprocessor with limited computation resource in real-time through PIL simulation. The test bench consists of a portable microprocessor *STM32L476RGT6* with ARM Cortex M4 32 b RISC core (up to 80 MHz frequency, 1 Mbyte flash memory and up to 128 Kbyte SRAM),[2] a host PC, and a USB cable. The complete EMS algorithm runs on this *STM32* microprocessor, and the HEV model runs on the host PC. Their communication follows the standard USART protocol.

At first, all the aforementioned EMS are converted into executable C code by Embedded Coder. The code generation procedure also generates the related *STM32* target file and CubeMX configuration file. Then, the generated C code is imported into an integrated development environment, e.g., Keil μVision. Finally, the complied EMS can be downloaded and run on the *STM32* microprocessor.

Apart from verifying the fuel efficiency generated by MIL simulation, PIL simulation mainly provides the information about the usage of the onboard computation resources for each EMS, including the computation overhead, memory demand, and CPU utilization. Each EMS runs with the identical t_s of 0.1 s, and the relevant results are listed in Table IV. Note that the explicit DP is only tested by the software-in-the-loop (SIL) simulation in which C code is generated and complied but only runs in the simulation environment of the host PC. Because this method requires more than 100 Mbyte memory space, it cannot be accepted by this *STM32* microprocessor. The other three EMS can be well implemented by this microprocessor as none of their memory overheads exceeds the upper limit of the flash memory and none of their CPU utilization surpasses 25% even in the worst case.

The values of fuel efficiency given in Table IV are slightly different from those in Table III. These small differences between MIL and PIL simulations stem from the round-off error and the end-to-end delay of PIL. First of all, in MIL simulation, all the variables have the double data type of high precision; however, to save memory overhead in PIL simulation, most of the variables are saved as the single data type, and the binary

[2] www.st.com/en/microcontrollers-microprocessors/stm32l476rg.html

variables, e.g., the ICE on/off status, have the unsigned-integer type. Furthermore, MIL simulation ignores the computation and communication delays of the controllers; but these delays cannot be avoided in PIL simulation. The delays are mainly determined by the running time to execute an EMS for once iteration, and thus further depends on the complexity of the EMS. The rule-based CDCS has the simplest algorithm so that its running time is the shortest and it consumes the least memory space. To solve the instantaneous control variables, A-ECMS needs to calculate the equivalence factor and Q-PMP must estimate the optimal costate at the start of each time interval. Therefore, they have the similar CPU utilization. The main difference between them is the method to obtain the equivalence factor and the costate. A-ECMS relies on a PID controller and a comparator, but Q-PMP checks a 3D look-up table. That is why Q-PMP requires much larger memory space.

D. Robustness Verification

The robustness of the relevant EMS is studied by running a large number of Monte Carlo simulations during which various combinations of system uncertainties, sensor noises and process disturbances are introduced into the HEV model. The details are explained as below.

1) The nominal value of the driver's weight is $70\,kg$, which is included in the HEV model and used by the design of each EMS. Since the real weight of a driver may vary, a constant disturbance of the uniform distribution in the range $(0,\ 2\,kg]$ is added to the vehicular gross weight M at each simulation.
2) The average auxiliary power consumption P_{aux} is a constant of $10\,W$ in previous simulations. For robustness test, it is replaced by a Gaussian distributed random value with mean 10 and variance 1, expressed as $P_{aux} \sim N(10, 1)$. P_{aux} is randomly generated before each simulation and then remains during this simulation.
3) A random signal with Gaussian distribution of mean 0 and variance 0.1 is adopted as sensor noise applied on the real-time HEV speed v measurement.
4) In practice, the ICE may not always reach its peak efficiency $\eta_{ce}(T^*_{ce}(\omega_{ce}), \omega_{ce})$. Therefore, a uniformly distributed random degradation percentage $\gamma \in [-2\%,\ 0]$ is added to $\eta_{ce}(\cdot)$ as a dynamic disturbance. By this way, $\eta_{ce}(\cdot)$ degrades by γ than its nominal value in the model.
5) Another dynamic disturbance ϱ, a uniformly distributed signal with bounds $\pm 5\%$, is imposed on the powertrain net tractive torque T_t. So that the real torque output on the wheel shaft becomes $T_t \cdot (1+\varrho)$.

The robustness of an EMS is indicated by the variations of the overall equivalent fuel efficiency and the final supercapacitor voltage referring to Table III. Since the explicit DP is infeasible to online implementation on the provided microprocessor, it is excluded from this test. Table V contains the relevant test results for Q-PMP, A-ECMS, and CDCS. All these three EMS can maintain their corresponding control performances with small standard deviations in both fuel efficiency and final supercapacitor voltage even though the mean values of these two indexes become slightly worse than those simulated in an ideal environment.

TABLE V
ROBUSTNESS TEST RESULTS FOR DIFFERENT EMS

EMS	Q-PMP	A-ECMS	CDCS
Mean Value of Fuel Efficiency (km/L)	211.25	197.31	173.09
Standard Deviation of Fuel Efficiency (km/L)	0.83	0.72	5.97
Variation of Fuel Efficiency to baseline (%)	-3.28	-2.92	-9.43
Mean Value of Final EES Voltage (V)	46.67	47.37	46.78
Standard Deviation of Final EES Voltage (V)	0.067	0.049	0.106
Variation of Final EES Voltage to baseline (%)	-0.17	-0.92	0.95

To the Q-PMP control, the real-time $\boldsymbol{\lambda}^*$ can be accurately calculated based on the assumption that the HEV can strictly follow the optimal speed trajectory. However, it is practically impossible because of the existence of all possible disturbances, including but not limited to those tested in this subsection. Once the HEV speed deviates from its optimal trajectory, the calculated $\boldsymbol{\lambda}$ is not optimal any more based on its present speed. Nevertheless, the Speed Regulator is able to regulate the HEV speed very close to the optimal trajectory. Therefore, the estimated $\boldsymbol{\lambda}$ is a good approximation of its optimum. Besides, thanks to the binary control on ICE operation, a small numeric error on $\boldsymbol{\lambda}$ will not result in a large error on ICE on/off switch. This explains why the performance by the Q-PMP control under robustness test degrades little from the one under MIL simulation.

The Q-PMP controller is slightly sensitive to perturbations because the mean value of fuel efficiency in Table V decreases by 3.28% and the mean of final supercapacitor voltage reduces by merely 0.17% compared to values in Table III. A-ECMS has similar reductions of 2.92% on the mean of fuel efficiency and 0.92% on the mean of final voltage. The CDCS controller suffers the worse robustness because its total number of ICE on/off switches often increases under the effects of multiple perturbations. Therefore, its fuel efficiency decreases obviously and is the worst of all three methods. To summarize, the Q-PMP controller enjoys decent robustness against the various system uncertainties. More importantly, as a candidate for online implementation, Q-PMP ensures a higher fuel efficiency than A-ECMS and CDCS.

E. Comparison to Continuously Controlled ICE

A motivation to explore the BC-ICE is the less computation overhead for torque split between the fuel and electric paths. This subsection further investigates whether the BC-ICE has higher energy efficiency than the CC-ICE.

To conduct a fair comparison, identical EMS should be selected to control the HEV with two types of ICE control methods. Given that Q-PMP is the proposed EMS in this paper, we conduct the same developing procedure described in Section III to design a new Q-PMP for the CC-ICE and test its performance by PIL simulation. The continuous control on ICE has two differences to the binary control: one is the ICE on/off status and another is

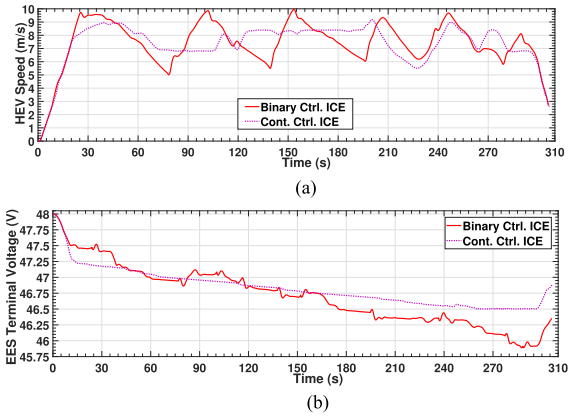


Fig. 8. PIL Simulation Results by Q-PMP on Different ICE Configurations. (a) HEV Speed Trajectories. (b) EES Terminal Voltage Trajectories.

TABLE VI
PIL SIMULATION RESULTS BY TWO VERSIONS OF Q-PMP

Configuration	BC-ICE	CC-ICE
Driving Time (s)	306.1	306.6
Final EES Voltage (V)	46.35	46.89
Fuel Consumption on Lap (mL)	9.54	11.19
Total Fuel Consumption (mL)	10.58	11.90
Fuel Efficiency (km/L)	213.76	190.14
Flash Memory Size (Kbyte)	496.60	305.06
Average CPU Utilization (%)	4.72	13.43

the ICE torque output. Under the continuous control, the ICE is switched on as long as its angular speed is over the idle speed, and switched off when its angular speed is below that. After the ICE is switched on, its torque output T_{ce} is selected between 0 and its admissible maximum. Consequently, s_{ce} is no longer an independent state variable and the binary control variable u_{ce} is replaced by a continuous one T_{ce}. And then, the HEV dynamical model is revised to one with 3 state variables $\mathbf{x} = [v \quad t \quad V_c]^T$ and 2 control variables $\mathbf{u} = [T_e \quad T_{ce}]^T$.

Figs. 8(a) and 8(b) respectively demonstrate the trajectories of HEV speed and supercapacitor voltage controlled by two versions of Q-PMP. Afterwards, Table VI summarizes the corresponding results from PIL simulations. In Fig. 8(a), the HEV with CC-ICE has similar speed trajectories to its counterpart at the beginning and ending phases, but enjoys a relatively flat trajectory during the middle phase. This is mainly because of the flexible torque output the CC-ICE can provide. Furthermore, the CC-ICE is only switched on/off once during a lap driving. Benefiting from the flexible ICE torque, the supercapacitor voltage trajectory from the HEV with CC-ICE is much smoother than the one of BC-ICE.

Table VI presents quantitative comparison of the two ICE control methods. In term of the equivalent fuel efficiency, the HEV with BC-ICE exceeds the one with CC-ICE by 12.4%. Although obtaining a higher final supercapacitor voltage by about 0.53 V, the HEV with CC-ICE consumes 17.3% more fuel than that with BC-ICE during the lap. This consequence can be further visually verified by Fig. 9. In this figure, the majority of operation points from BC-ICE, displayed by green circles, are concentrated around the peak efficiency curve of color red. There are still a small number of green circles located in

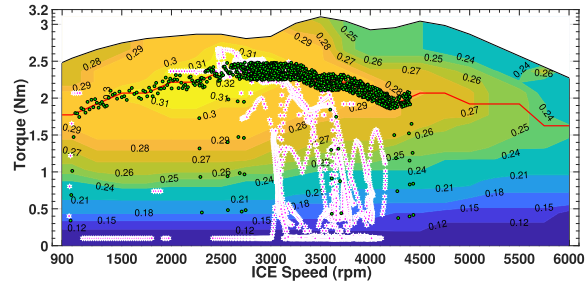


Fig. 9. Distribution of ICE Operation Points on Efficiency Map by Two Different Configurations.

lower efficiency regions, because of the ICE's transient behavior during the switch operations. By contrast, the operations points from CC-ICE, displayed by plum hexagrams, are more widely distributed over a larger region of various efficiencies. The fundamental cause to the larger fuel consumption by CC-ICE is that the ICE cannot be freely switched off by the EMS and thus a large proportion of the operation points are of lower efficiency.

From the perspective of online implementation, the Q-PMP controller based on CC-ICE requires roughly 40% less onboard flash memory than that of BC-ICE. Since the state variable s_{ce} is omitted in the HEV model with CC-ICE, once the optimal speed trajectory $v^*(t)$ is found out and used as a reference for online control, the only state variable left is V_c. Consequently, the look-up table of value function from DP has only 2 DOF and thus requires theoretically half memory size of the one for HEV with BC-ICE. However, the average CPU utilization of the new Q-PMP is about 2.85 times of its counterpart. This is because the real-time torque split in the Primary Torque Splitter is not a binary optimization problem. More complex algorithm is required to find a continuous T_{ce}^* for each periodic iteration. Therefore, the overall computation overhead greatly increases of necessity.

VI. CONCLUSION AND FUTURE WORK

Aiming at a parallel hybrid electric racing car with a BC-ICE, this paper proposes a Q-PMP control method to improve its fuel efficiency on a prescribed racing track. To obtain a close-to-optimal performance by a portable microprocessor with limited computation resource, offline DP solutions are fully exploited to support a computationally efficient online controller. On the one hand, DP provides an optimal HEV speed trajectory as the reference so that the original OCP is simplified to a great extent; on the another hand, a 3D look-up table of value function from the DP solutions is used to estimate the optimal costate for the Q-PMP controller which is responsible for determining the real-time ICE on/off status and the supplementary torque by the EMs.

The MIL and PIL simulation results exhibit the advantages of this proposed EMS when compared with several other methods: it has better fuel economy than the rule-based CDCS and the A-ECMS controllers by around 14% and 7.5%, respectively; and it realizes similar performance to the explicit DP at the expense of less than 1% memory overhead. In addition, the performance of Q-PMP is compared between HEVs with BC-ICE and CC-ICE.

The PIL simulation result indicates that the Q-PMP control for BC-ICE outperforms its counterpart by nearly 12% higher fuel economy but requires only approximately 1/3 CPU utilization.

According to the racing rules, each HEV requires charge sustain for any driving assignment. For this HEV, the size of the onboard EES, a supercapacitor, is conservatively determined to guarantee an adequate supply of onboard electric energy. In fact, the supercapacitor voltage varies within only a narrow region of its admissible range during the lap. It hints that the majority of the available electricity is useless but the large capacitance brings in extra weight to the HEV. In view of this, the future research will focus on the simultaneous optimization of the component sizing together with the energy management. By this means, a better fuel economy can be achieved by the combination of an appropriate EMS and a well-optimized powertrain with sufficient driving capability yet without redundant load.

APPENDIX

INTRODUCTION TO BENCHMARK EMS

A. Explicit DP

Explicit DP has been designed in our previous work [41]. The basic process consists of 3 steps: 1) applying DP to solve the OCP defined in Section III and generating 2 independent 5D look-up tables for online usage; 2) calculating real-time control commands u_{ce}^* and T_e^* by checking the look-up tables from the state information of v, t, V_c, s_{ce} and y; 3) splitting T_e^* into T_{dc} and T_{bl} according to (26) and (25a).

B. A-ECMS

The principle of A-ECMS is very similar to that of Q-PMP. The only difference is that the equivalence factor $\mathbf{s}$ is estimated by explicit functions of the state vector $\bar{\mathbf{x}}$ in (34). Referring to [18], [19] and [38], when SOC is the only state variable, $\mathbf{s}$ can be calculated by a PID controller with the SOC deviation from its reference as the input. Since there is another binary state variable s_{ce} in the OCP, $\mathbf{s}$ is a vector containing two elements, $\mathbf{s}=[s_1 \quad s_2]^T$. According to the real-time V_c, s_{ce} and $\mathbf{s}$, A-ECMS calculates the optimal control u_{ce}^* that minimizes the instantaneous equivalent fuel consumption m_{eq}. The relevant formulas are given below.

$$\begin{aligned} m_{eq}(t_j) &= \dot{m}_f\left(s_{ce}(t_j), u_{ce}(t_j)\right)\cdot t_s \\ &\quad + s_1(t_j)\cdot E_e\left(s_{ce}(t_j), V_c(t_j), u_{ce}(t_j)\right) \\ &\quad + s_2(t_j)\cdot m_{sw}\left(s_{ce}(t_j), u_{ce}(t_j)\right) \end{aligned} \tag{49}$$

$$\begin{aligned} E_e(t_j) &= \frac{C}{2}\cdot\left[V_c^2(t_j) - \left(V_c(t_j) + \dot{V}_c(t_j)\cdot t_s\right)^2\right] \\ &= -C\cdot V_c(t_j)\dot{V}_c(t_j)\cdot t_s - \frac{C\cdot\left(\dot{V}_c(t_j)\cdot t_s\right)^2}{2} \end{aligned} \tag{50}$$

$$\begin{aligned} s_1(t_j) &= s_{1,0} + K_p\Delta V_c(t_j) + K_i\sum_{i=0}^{j}\Delta V_c(t_i) \\ &\quad + K_d\left(\Delta V_c(t_j) - \Delta V_c(t_{j-1})\right) \end{aligned} \tag{51}$$

$$\Delta V_c(t_j) = V_c^*(t_j) - V_c(t_j) \tag{52}$$

$$s_2(t_j) = \begin{cases} s_{2,0}; & s_{ce}(t_j) = s_{ce}^*(t_j) \\ -s_{2,0}; & s_{ce}(t_j) \neq s_{ce}^*(t_j) \end{cases} \tag{53}$$

where $s_{1,0}$ and $s_{2,0}$ are positive constants and represent the initial guesses of s_1 and s_2 respectively; the optimal trajectories of supercapacitor voltage $V_c^*(t)$ and ICE on/off status $s_{ce}^*(t)$ come from the offline DP solutions; E_e is the consumption of electric energy during a control period t_s; ΔV_c is the current supercapacitor voltage deviation from its optimum; K_p, K_i and K_d in (51) are the proportional, integral and derivative gains of the PID controller to calculate s_1. For ICE status control, it is intended to follow its optimal trajectory by setting s_2 a positive value if s_{ce} at present matches the corresponding value on its optimal trajectory so that the ICE prefers to hold its current status; otherwise, a negative s_2 will promote a switch on ICE on/off status. All the tunable parameters in this controller are found by trial-and-error.

C. CDCS

The CDCS controller also relies on the given optimal speed trajectory to compute an instantaneous T_t. Two voltage thresholds V_1 and V_2 $(V_{c,\min} < V_1 < V_2 < V_{c,\max})$ are prescribed to divide the admissible range of supercapacitor voltage into 3 sections. When $V_c \geq V_2$, EMs are preferred to drive the HEV exclusively unless they cannot satisfy T_t; when $V_c \leq V_1$, ICE is switched on to provide torque for driving the HEV and recharge the supercapacitor if T_{ce} can satisfy T_t; and when V_c is located between V_1 and V_2, ICE prefers to maintain its current on/off status and the two EMs either assist the ICE if T_{ce} is insufficient, or convert the surplus power into electricity if T_{ce} is more than required. To avoid frequent ICE switches, a time constant t_{sw} is introduced as the minimum period during which the ICE must maintain its on/off status and reject any switch command from the superior control.

REFERENCES

[1] E. Silvas, T. Hofman, N. Murgovski, L. F. P. Etman, and M. Steinbuch, "Review of optimization strategies for system-level design in hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 57–70, Jan. 2017.

[2] K. V. Singh, H. O. Bansal, and D. Singh, "A comprehensive review on hybrid electric vehicles: Architectures and components," *J. Mod. Transp.*, vol. 27, no. 2, pp. 77–107, 2019.

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

[4] A. Sciarretta and L. Guzzella, "Control of hybrid electric vehicles," *IEEE Control Syst.*, vol. 27, no. 2, pp. 60–70, Apr. 2007.

[5] S. Onori, L. Serrao, and G. Rizzoni, *Hybrid Electric Vehicles: Energy Management Strategies*. London Heidelberg New York Dordrecht: Springer, 2016.

[6] L. Guzzella and C. H. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Berlin, Heidelberg: Springer-Verlag, 2nd edition, 2010.

[7] W. Liu, *Introduction to Hybrid Vehicle System Modeling and Control*. Hoboken, New Jersey, NJ, USA: John Wiley & Sons, Inc., 2013.

[8] S. Zhang, R. Xiong, and F. Sun, "Model predictive control for power management in a plug-in hybrid electric vehicle with a hybrid energy storage system," *Appl. Energy*, vol. 185, pp. 1654–1662, 2017.

[9] 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 Trans. Veh. Technol.*, vol. 66, no. 6, pp. 4534–4549, Jun. 2017.

[10] W. Enang and C. Bannister, "Modelling and control of hybrid electric vehicles (a comprehensive review)," *Renew. Sustain. Energy Rev.*, vol. 74, pp. 1210–1239, 2017.

[11] Y. Huang *et al.*, "A review of power management strategies and component sizing methods for hybrid vehicles," *Renew. Sustain. Energy Rev.*, vol. 96, pp. 132–144, 2018.

[12] M. Kim, D. Jung, and K. Min, "Hybrid thermostat strategy for enhancing fuel economy of series hybrid intracity bus," *IEEE Trans. Veh. Technol.*, vol. 63, no. 8, pp. 3569–3579, Oct. 2014.

[13] 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," *Appl. Energy*, vol. 185, pp. 1633–1643, 2017.

[14] S. G. Wirasingha and A. Emadi, "Classification and review of control strategies for plug-in hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 111–122, Jan. 2011.

[15] S. Zhang and R. Xiong, "Adaptive energy management of a plug-in hybrid electric vehicle based on driving pattern recognition and dynamic programming," *Appl. Energy*, vol. 155, pp. 68–78, 2015.

[16] Y. Yang, X. Hu, H. Pei, and Z. Peng, "Comparison of power-split and parallel hybrid powertrain architectures with a single electric machine: Dynamic programming approach," *Appl. Energy*, vol. 168, pp. 683–690, 2016.

[17] Y. Yang, H. Pei, X. Hu, Y. Liu, C. Hou, and D. Cao, "Fuel economy optimization of power split hybrid vehicles: A rapid dynamic programming approach," *Energy*, vol. 166, pp. 929–938, 2019.

[18] A. Nguyen, J. Lauber, and M. Dambrine, "Optimal control based algorithms for energy management of automotive power systems with battery/supercapacitor storage devices," *Energy Convers. Manag.*, vol. 87, pp. 410–420, 2014.

[19] S. Onori and L. Tribioli, "Adaptive pontryagin's minimum principle supervisory controller design for the plug-in hybrid GM chevrolet volt," *Appl. Energy*, vol. 147, pp. 224–234, 2015.

[20] N. Guo, J. Shen, R. Xiao, W. Yan, and Z. Chen, "Energy management for plug-in hybrid electric vehicles considering optimal engine ON/OFF control and fast state-of-charge trajectory planning," *Energy*, vol. 163, pp. 457–474, 2018.

[21] Z. Chen, C. C. Mi, B. Xia, and C. You, "Energy management of power-split plug-in hybrid electric vehicles based on simulated annealing and pontryagin's minimum principle," *J. Power Sources*, vol. 272, pp. 160–168, 2014.

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

[23] S. Y. Chen, Y. H. Hung, C. H. Wu, and S. T. Huang, "Optimal energy management of a hybrid electric powertrain system using improved particle swarm optimization," *Appl. Energy*, vol. 160, pp. 132–145, 2015.

[24] Z. Chen, R. Xiong, and J. Cao, "Particle swarm optimization-based optimal power management of plug-in hybrid electric vehicles considering uncertain driving conditions," *Energy*, vol. 96, pp. 197–208, 2016.

[25] C. Yang, S. Du, L. Li, S. You, Y. Yang, and Y. Zhao, "Adaptive real-time optimal energy management strategy based on equivalent factors optimization for plug-in hybrid electric vehicle," *Appl. Energy*, vol. 203, pp. 883–896, 2017.

[26] S. Xie, X. Hu, Z. Xin, and J. Brighton, "Pontryagin's minimum principle based model predictive control of energy management for a plug-in hybrid electric bus," *Appl. Energy*, vol. 236, pp. 893–905, 2019.

[27] C. Sun, X. Hu, S. J. Moura, and F. Sun, "Velocity predictors for predictive energy management in hybrid electric vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 3, pp. 1197–1204, May 2015.

[28] U. Sartori, F. Biral, E. Bertolazzi, and S. Onori, "On-line power management optimization of a hybrid electric vehicle with non linear MPC and battery re-charge equivalent cost," in *Proc. Annu. Conf. IEEE Ind.* Electron. Soc., Florence, Italy, 2016, pp. 5094–5100.

[29] Y. Zhou, A. Ravey, and M. C. Péra, "Multi-mode predictive energy management for fuel cell hybrid electric vehicles using markov driving pattern recognizer," *Appl. Energy*, vol. 258, no. 114057, pp. 1–17, 2020.

[30] J. T. B. A. Kessels, M. W. T. Koot, P. P. J. van denBosch, and D. B.Kok, "Online energy management for hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3428–3440, Nov. 2008.

[31] M. Khodabakhshian, L. Feng, and J. Wikander, "Improving fuel economy and robustness of an improved ECMS method," in *Proc. IEEE Int. Conf. Control Autom.*, Hangzhou, China, 2013, pp. 598–603.

[32] T. Liu, L. Feng, M. Hellgren, and J. Wikander, "Increasing fuel efficiency of a hybrid electric competition car by a binary equivalent consumption minimization strategy," in *Proc. IEEE Int. Conf. Autom. Sci. Eng.*, Munich, Germany, 2018, pp. 1–7.

[33] R. Johri and Z. Filipi, "Optimal energy management of a series hybrid vehicle with combined fuel economy and low-emission objectives," in *Proc. Inst. Mech. Eng., Part D: J. Automobile Eng.*, vol. 228, no. 12, pp. 1424–1439, 2014.

[34] T. Nüesch, A. Cerofolini, G. Mancini, N. Cavina, C. Onder, and L. Guzzella, "Equivalent consumption minimization strategy for the control of real driving NOx emissions of a diesel hybrid electric vehicle," *Energies*, vol. 7, no. 5, pp. 3148–3178, 2014.

[35] D. F. Opila, X. Wang, R. McGee, R. B. Gillespie, J. A. Cook, and J. W. Grizzle, "An energy management controller to optimally trade off fuel economy and drivability for hybrid vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 6, pp. 1490–1505, 2012.

[36] M. Khodabakhshian, L. Feng, and J. Wikander, "Optimization of gear shifting and torque split for improved fuel efficiency and drivability of HEVs," in *Proc. SAE* 20 13 *World Congr. Exhib.*, vol. 1461, Detroit, Michigan, USA, 2013, pp. 1–13.

[37] S. Bashash, S. J. Moura, J. C. Forman, and H. K. Fathy, "Plug-in hybrid electric vehicle charge pattern optimization for energy cost and battery longevity," *J. Power Sources*, vol. 196, no. 1, pp. 541–549, 2011.

[38] X. Tian, R. He, X. Sun, Y. Cai, and Y. Xu, "An ANFIS-based ECMS for energy optimization of parallel hybrid electric bus," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1473–1483, Feb. 2020.

[39] D. D. Tran, M. Vafaeipour, M. El Baghdadi, R. Barrero, J. Van Mierlo, and O. Hegazy, "Thorough state-of-the-art analysis of electric and hybrid vehicle powertrains: Topologies and integrated energy management strategies," *Renew. Sustain. Energy Rev.*, vol. 119, pp. 1–29, 2020.

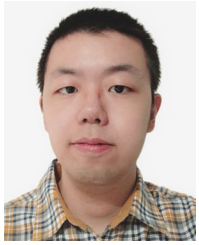
[40] M. Josevski and D. Abel, "Gear shifting and engine on/off optimal control in hybrid electric vehicles using partial outer convexification," in *Proc. IEEE Conf. Control Appl.*, Buenos Aires, Argentina, 2016, pp. 562–568.

[41] T L. Liu, M. Feng, Hellgren, and J.Wikander, "A binary controller to ensure engine peak efficiency for a parallel hybrid electric car," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Auckland, New Zealand, 2019, pp. 726–732.

[42] R. E. Bellman and E. S. Lee, "History and development of dynamic programming," *IEEE Control Syst. Mag.*, vol. 4, no. 4, pp. 24–28, Nov. 1984.

Tong Liu (Member, IEEE) received the B.Sc. degree in electronics information engineering, and the M.Sc. degree in traffic information engineering and control from Chang'an University, Xi'an, China, in 2010 and 2013, respectively. He is currently working toward the Ph.D. degree with the Department of Machine Design, KTH Royal Institute of Technology, Stockholm, Sweden. Since 2013, he has been a traffic engineer with China Communications Construction Company Ltd. His research interests include optimal control of dynamic process, energy management of new energy vehicles, and structure optimization of hybrid powertrain.

Lei Feng (Member, IEEE) received the B.S. and M.S. degrees from Xi'an Jiaotong University, Xi'an, China, in 1998 and 2001, respectively, and the Ph.D. degree from the Systems Control Group, the University of Toronto, Toronto, ON, Canada, in 2007. In 2012, he joined the Mechatronics and Embedded Control System Division, the KTH Royal Institute of Technology, Stockholm, Sweden, where he is currently an Associate Professor. His main research interests include energy management control of mechatronic systems, autonomous driving, verification and control synthesis of cyber-physical systems, and supervisory control of discrete event systems.



Wenyao Zhu received the B.S. degree in electrical and computer engineering from Shanghai Jiao Tong University, Shanghai, China, in 2018, and the M.S. degree in embedded systems from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2020. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology. His current research interests include edge computing and real-time systems.