SCIENCE CHINA
Technological Sciences

• Article •

May 2017 Vol.60 No.5: 737–748
doi: 10.1007/s11431-016-0640-2

MPC-based energy management with adaptive Markov-chain prediction for a dual-mode hybrid electric vehicle

XIANG ChangLe, DING Feng, WANG WeiDa*, HE Wei & QI YunLong

School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China

Received August 30, 2016; accepted December 20, 2016; published online January 12, 2017

The energy management strategy is an important part of a hybrid electrical vehicle design. It is used to improve fuel economy and to sustain a proper battery state of charge by controlling the power components while satisfying various constraints and driving demands. However, achieving an optimal control performance is challenging due to the nonlinearities of the hybrid powertrain, the time varying constraints, and the dilemma in which controller complexity and real-time capability are generally conflicting objectives. In this paper, a real-time capable cascaded control strategy is proposed for a dual-mode hybrid electric vehicle that considers nonlinearities of the system and complies with all time-varying constraints. The strategy consists of a supervisory controller based on a non-linear model predictive control (MPC) with a long sampling time interval and a coordinating controller based on linear model predictive control with a short sampling time interval to deal with different dynamics of the system. Additionally, a novel data based methodology using adaptive Markov chains to predict future load demand is introduced. The predictive future information is used to improve controller performance. The proposed strategy is implemented on a real test-bed and experimental trials using unknown driving cycles are conducted. The results demonstrate the validity of the proposed approach and show that fuel economy is significantly improved compared with other methods.

hybrid electric vehicle, dual-mode, energy management, Markov chains, model predictive control

Citation: Xiang C L, Ding F, Wang W D, et al. MPC-based energy management with adaptive Markov-chain prediction for a dual-mode hybrid electric vehicle. Sci China Tech Sci, 2017, 60: 737–748, doi: 10.1007/s11431-016-0640-2

1 Introduction

Hybrid electric vehicles (HEVs) represent a short-term approach to reducing pollutant emissions and improving the fuel efficiency of automobiles [1]. Compared with conventional vehicles, HEVs add additional energy sources and combine the output power with the internal combustion engine. On the market, there are currently different types of HEVs designed for different purposes. Typically, a dual-mode power split configuration is widely used in heavy duty vehicles, such as in SUVs and trucks. This can provide more power and higher efficiency over the entire vehicle velocity range [2,3].

The achievable improvement in HEVs strongly depends on the adopted energy management strategy (EMS) [4]. The primary objective of an EMS is to minimise fuel consumption while satisfying all physical constraints and imposed driving demands [5]. Early energy management strategies are mostly based on heuristic rules inspired by engineering intuition. The heuristics based on analysis of power flow in a hybrid powertrain and human experience is used to design deterministic rules to split the requested power between power sources [6,7]. In ref. [8] there is a report of a rule-based strategy to turn on/off the engine depending on the current SOC level. These methods have the advantages of easy implementation and little computation but lack flexibility when faced with various conditions. A fuzzy logic controller is an ex-

*Corresponding author (email: wangwd0430@163.com) tension of the conventional rule-based controller, which has better robustness and adaptation [9,10]. Other researchers [11] uses load power and SOC to develop a fuzzy logic controller to distribute the requested power to two power sources. However, the performance of this type of method is highly dependent on engineering experience and the solution is not optimal.

To achieve better performance, many optimisation-based strategies have been developed by researchers using advanced intelligent algorithms [12,13], such as dynamic programming (DP) [14,15], simulated annealing [16], particle swarm optimisation [17], Pontryagin's minimum principle (PMP) [18], genetic algorithms [19], and neural networks [20]. The DP algorithm is an effective method with which to obtain globally optimal solutions. Other workers [21] have proposed a DP-based strategy to balance the optimisation of fuel economy and drivability. However, this technique requires full knowledge of the entire driving cycle in advance, so it can only be used in off-line simulations. Others [22] employ PMP techniques to minimise overall fuel consumption and to ensure the final SOC matches the value at the beginning of the cycle, however, in addition, the PMP cannot be directly implemented on-line because the co-states in the Hamiltonian function must be located by iterative calculations.

A suitable method for real time optimisation is offered by model predictive control (MPC) that includes methodologies to predict the future load demand of the vehicle [23–25]. The MPC-based controller can obtain an optimal solution in real time and is capable of implementation with limited computation and memory resources. Others [26] have proposed an MPC-based strategy to obtain real-time optimal fuel economy. However, no a priori knowledge of the future load demand is considered. Elsewhere, an exponentially varying velocity predictor has been used to provide an intuitive understanding of how velocity prediction affects fuel economy [27]. A neural network has been used to predict future vehicle velocities [28]. Li et al. [29] classify driving behaviour to predict future behaviours. Ref. [30] develops an EMS with drive cycle prediction which is suitable for vehicles operating on the same route. However, these methods are strongly dependent on the driving cycles and are not suitable for real-world situations.

In this paper, an online cascaded MPC-based controller with a novel methodology predicting future load demand is proposed. Assuming that no environment detecting sensor and no future driving information is available, the future load is predicted based on Markov chains with only historical and current information. The Markov-chain transition possibility is computed from a comprehensive dataset and is updated in real-time to adapt to changes in driving conditions. The prediction is used to improve the performance of the controller. The cascaded control concept distributes the control problem to two separate controllers, the supervisory controller and the coordinating controller. The supervisory controller is based on non-linear model predictive control (NMPC) to optimise the fuel economy and the SOC trajectory. The coordinating controller is based on linear model predictive control (LMPC) and applies the demands from the supervisory controller to the powertrain: the effectiveness of the proposed EMS is demonstrated on a real test-bed.

2 Energy management strategy

In this section, the design of the proposed EMS is described. After an overview of the controller architecture, the control-oriented models are developed and the constraints are summarised. Subsequently, the supervisory and the coordinating controller are described in detail.

2.1 Architecture of the control system

The EMS is proposed for a dual-mode HEV which consists of an internal combustion engine and two electric motor generators (MGs). The configuration of the powertrain is shown in Figure 1. The engine and the MGs are controlled by the torque commands: T_{e_cmd}, T_{MG1_cmd}, and T_{MG2_cmd}, respectively.

The main objectives of the EMS are to minimise fuel consumption and to sustain an ideal battery SOC while satisfying all constraints and imposed driving demands. The complex problem is addressed by employing a cascade control system because the time constants of the powertrain differ significantly between the slow dynamic of the battery's SOC and the fast dynamic of the rotational speeds and torques. The cascaded control system includes an NMPC-based supervisory controller for the slow dynamic and an LMPC-based coordinating controller for the fast dynamic. An overview of the EMS is schematically depicted in Figure 2.

The supervisory controller is designed to improve fuel economy and to hold the SOC at a desired value, while the predictive torque demand T_{dmd} and velocity demand V_{dmd} from the Markov-chain predictor are considered in the

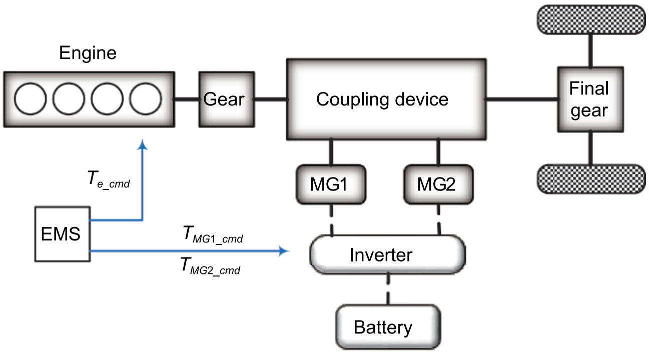

Figure 1 (Color online) The powertrain configuration.

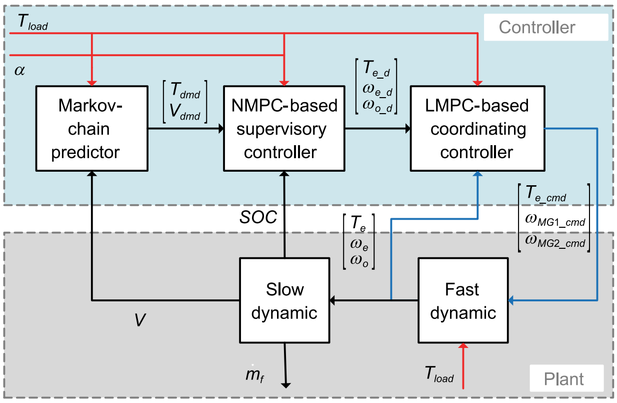

Figure 2 (Color online) Energy management strategy scheme.

optimisation. The sampling time interval of the supervisory controller is set to be t_{m_s}=1 s. Demand engine torque T_{e_d}, demand engine speed ω_{e_d}, and demand output speed ω_{o_d} are the outputs from the supervisory controller and also the inputs to the coordinating controller. Based on the actual state variables engine torque T_e, engine speed ω_e, and output speed ω_o, the coordinating controller controls the inputs through the manipulated variables engine torque command T_{e_cmd}, MG1 torque command T_{MG1_cmd}, and MG2 torque command T_{MG2_cmd}. A short sampling time interval t_{s_s}=100 ms of the coordinating controller is used so that sufficient powertrain dynamics can be guaranteed for load disturbances. The vehicle velocity V and battery SOC are fed back to the controllers. Note that, during its operation, only the driver's pedal position α and the load disturbance T_{load} are available, while the future load trajectory is, a priori, unknown.

2.2 System models for controller design

As shown in Figure 3, the coupling device consists of three planetary gear sets (PGs), a clutch, and a brake. The input of the coupling device is connected to the engine through a gearbox and the output power drives the vehicle though the final gear. Two modes are provided by operating the clutch and the brake for larger power and higher efficiency over the entire vehicle velocity range [31]. In mode 1, the brake is engaged and the clutch is released. In mode 2, on the contrary, the clutch is engaged and the brake is released. In a PG, due to the mechanical connection afforded through the gear teeth, the rotational speeds of the sun gear, the ring gear, and the carrier gear satisfy the following relationship:

$$\omega_s + k\omega_r - (1+k)\omega_c = 0, \tag{1}$$

where ω_s, ω_r, and ω_c are the rotational speeds of the sun gear, the ring gear, and the carrier gear; k is the PG's inherent parameter, obtained by

$$k = \frac{Z_r}{Z_s}, \tag{2}$$

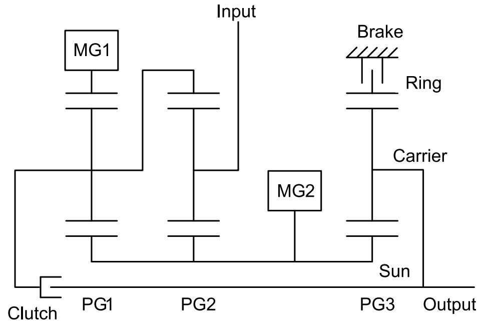

Figure 3 Structure of the coupling device.

where Z_r and Z_S are the number of teeth on the ring gear and the sun gear.

Through the coupling device, power coupling can be accomplished such that the engine operating point becomes independent of the vehicle operation. For the design of the supervisory controller, fast dynamics such as the inertial losses of the engine and the MGs are ignored. Assuming that all the connecting shafts in the powertrain are rigid, based on the lever analogy [5], the relationships between the components' speeds and torques can be described as follows, for mode 1

$$\begin{bmatrix} \omega_{MG1\_d} \\ \omega_{MG2\_d} \end{bmatrix} = \begin{bmatrix} \dfrac{(1+k_1)(1+k_2)}{k_1k_2} & -\dfrac{(1+k_1+k_2)(1+k_3)}{k_1k_2} \\ 0 & 1+k_3 \end{bmatrix} \times \begin{bmatrix} \omega_{i\_d} \\ \omega_{o\_d} \end{bmatrix}, \tag{3}$$

$$\begin{bmatrix} T_{MG1\_d} \\ T_{MG2\_d} \end{bmatrix} = \begin{bmatrix} -\dfrac{k_1k_2}{(1+k_1)(1+k_2)} & 0 \\ -\dfrac{1+k_1+k_2}{(1+k_1)(1+k_2)} & \dfrac{1}{1+k_3} \end{bmatrix} \begin{bmatrix} T_{i\_d} \\ T_{o\_d} \end{bmatrix}, \tag{4}$$

and for mode 2

$$\begin{bmatrix} \omega_{MG1\_d} \\ \omega_{MG2\_d} \end{bmatrix} = \begin{bmatrix} -\dfrac{1+k_2}{k_1} & \dfrac{1+k_1+k_2}{k_1} \\ 1+k_2 & -k_2 \end{bmatrix} \begin{bmatrix} \omega_{i\_d} \\ \omega_{o\_d} \end{bmatrix}, \tag{5}$$

$$\begin{bmatrix} T_{MG1\_d} \\ T_{MG2\_d} \end{bmatrix} = \begin{bmatrix} -\dfrac{k_1k_2}{(1+k_1)(1+k_2)} & \dfrac{k_1}{1+k_1} \\ -\dfrac{1+k_1+k_2}{(1+k_1)(1+k_2)} & \dfrac{1}{1+k_1} \end{bmatrix} \begin{bmatrix} T_{i\_d} \\ T_{o\_d} \end{bmatrix}, \tag{6}$$

where ω_{i_d}, ω_{MG1_d}, and ω_{MG2_d} are the speeds of the input, the MG1 and MG2; k_1, k_2 and k_3 are the inherent parameters of the PG1, the PG2 and PG3; T_{i_d}, T_{MG1_d}, T_{MG2_d} and T_{o_d} are the torques of the input, the MG1, the MG2, and the output. There are also four kinematic equality constraints between the speeds and the torques

$$\omega_{e\_d} = i_q\omega_{i\_d}, \tag{7}$$

$$V_{dmd} = \frac{r_w \omega_{o\_d}}{i_f}, \tag{8}$$

$$T_{e\_d} = \frac{T_{i\_d}}{i_q}, \tag{9}$$

$$T_{dmd} = i_f T_{o\_d}, \tag{10}$$

where i_q is the gear ratio of the front drive; i_f is the gear ratio of the final drive, and r_w is the radius of the wheels.

An empirical map of the engine, as shown in Figure 4, obtained experimentally, is used to relate the fuel consumption to the engine speed and torque as follows:

$$\dot{m}_f = f_e(\omega_{e\_d}, T_{e\_d}), \tag{11}$$

where m_f is the fuel consumption and f_e is a non-linear look-up table.

The battery's state of charge (SOC) is the main state variable in this supervisory controller. It needs to be defined so as to reflect battery energy status and can be calculated as follows:

$$\dot{\mathrm{SOC}} = -\frac{I_{\mathrm{batt}}}{C_{\mathrm{batt}}}, \tag{12}$$

where I_{batt} is the battery current and C_{batt} is the battery capacity. Due to the requirement of online calculation to obtain the SOC trajectory in the prediction horizon, the computational demand of dynamic models is significantly higher than for more precise models. Therefore, a simplified internal resistance battery model [32], as depicted in Figure 5, is used which results in

$$P_{\mathrm{batt}} = V_{\mathrm{oc}}(\mathrm{SOC}, t_{\mathrm{batt}}) I_{\mathrm{batt}} - I_{\mathrm{batt}}^2 R_{\mathrm{batt}}(\mathrm{SOC}, t_{\mathrm{batt}}), \tag{13}$$

where P_{batt} is the battery power, $V_{\mathrm{oc}}(\mathrm{SOC}, t_{\mathrm{batt}})$ is the battery open circuit voltage, and $R_{\mathrm{batt}}(\mathrm{SOC}, t_{\mathrm{batt}})$ is the battery internal resistance, while the values of $V_{\mathrm{oc}}(\mathrm{SOC}, t_{\mathrm{batt}})$ and $R_{\mathrm{batt}}(\mathrm{SOC}, t_{\mathrm{batt}})$ are derived from look-up tables that are extracted from experimental data. One should note that a positive value of P_{batt} indicates that the battery is discharging and a negative value indicates that the battery is charging. The battery power flows through the inverter to supply power to the MGs, and the relationship is governed by

$$P_{\mathrm{batt}} = T_{MG1\_d}\,\omega_{MG1\_d}\,\eta_{MG1}^{k_{MG1}} + T_{MG2\_d}\,\omega_{MG2\_d}\,\eta_{MG2}^{k_{MG2}}, \tag{14}$$

where η_{MG1} and η_{MG2} are the efficiencies of the MGs and when the battery is discharged, the exponents k_{MG1} and k_{MG2} are equal to –1, and, when the battery is charged, the exponents are both equal to 1.

Based on eqs. (1)–(14), choosing the input $\boldsymbol{u}_m$, state $\boldsymbol{x}_m$, output $\boldsymbol{y}_m$, and disturbance $\boldsymbol{v}_m$ of the system by

$$\boldsymbol{x}_m = [\mathrm{SOC}],\ \boldsymbol{u}_m = \begin{bmatrix} T_{e\_d} \\ \omega_{e\_d} \end{bmatrix},\ \boldsymbol{v}_m = \begin{bmatrix} V_{dmd} \\ T_{dmd} \end{bmatrix},\ \boldsymbol{y}_m = \begin{bmatrix} \mathrm{SOC} \\ \dot{m}_f \end{bmatrix}, \tag{15}$$

the discrete non-linear model for supervisory controller design can be represented by

$$\boldsymbol{x}_m(k+1) = f_m(\boldsymbol{x}_m(k), \boldsymbol{u}_m(k), \boldsymbol{v}_m(k)), \tag{16}$$

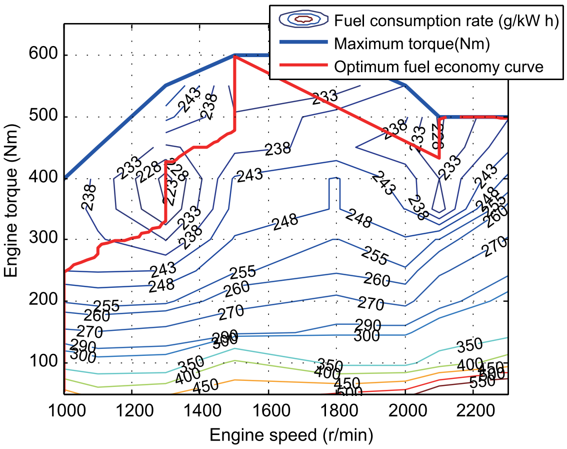

Figure 4 (Color online) Contour map of the engine's fuel consumption rate.

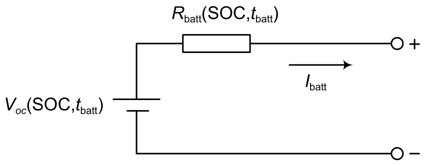
$R_{\mathrm{batt}}(\mathrm{SOC},t_{\mathrm{batt}})$
+
I_{batt}
$V_{\mathrm{oc}}(\mathrm{SOC},t_{\mathrm{batt}})$
–

Figure 5 The internal resistance battery model.

$$\boldsymbol{y}_m(k+1) = g_m(\boldsymbol{x}_m(k), \boldsymbol{u}_m(k), \boldsymbol{v}_m(k)). \tag{17}$$

Meanwhile a corresponding model of the fast system dynamics of the plant for the coordinating controller is required. Based on the principle of first-order lag elements [33–35], the fast dynamic of the engine torque can be modelled by

$$\tau_e \dot{T}_e + T_e = T_{e\_cmd}, \tag{18}$$

where τ_e is the time constant of the engine torque. Note that the MGs' response speeds are much faster than those of the engines, so the dynamics of the MGs' torques are neglected. Furthermore, based on the principle of angular momentum [2], the fast dynamics of the rotational speeds can be modelled as follows (for mode 1):

$$T_e + \frac{(1+k_1)(1+k_2)}{k_1 k_2 i_q} T_{MG1\_cmd} = J_e \dot{\omega}_e, \tag{19}$$

$$\begin{aligned} &-\frac{(1+k_1+k_2)(1+k_3)}{k_1 k_2} T_{MG1\_cmd} \\ &+(1+k_3) T_{MG2\_cmd} - T_{load} = J_p \dot{\omega}_o, \end{aligned} \tag{20}$$

and for mode 2

$$T_e - \frac{1+k_2}{k_1 i_q} T_{MG1\_cmd} + \frac{1+k_2}{i_q} T_{MG2\_cmd} = J_e \dot{\omega}_e, \tag{21}$$

$$\frac{1+k_1+k_2}{k_1} T_{MG1\_cmd} - k_2 T_{MG2\_cmd} - T_{load} = J_p \dot{\omega}_o, \tag{22}$$

where J_e and J_p are the engine's inertia and the powertrain's total inertia. Based on eqs. (18) to (22), choosing the input $\boldsymbol{u}_s$, state $\boldsymbol{x}_s$, output $\boldsymbol{y}_s$, and disturbance $\boldsymbol{v}_s$ of the system from

$$\boldsymbol{x}_s=\begin{bmatrix}T_e\\ \omega_e\\ \omega_o\end{bmatrix},\ \boldsymbol{u}_s=\begin{bmatrix}T_{e\_cmd}\\ T_{MG1\_cmd}\\ T_{MG2\_cmd}\end{bmatrix},\ \boldsymbol{v}_s=T_{\text{load}},\ \boldsymbol{y}_s=\begin{bmatrix}T_e\\ \omega_e\\ \omega_o\end{bmatrix},\tag{23}$$

the discrete linear state space coordinating controller model may be given by

$$\boldsymbol{x}_s(k+1)=\boldsymbol{A}_s\boldsymbol{y}_s(k)+\boldsymbol{B}_s\boldsymbol{u}_s(k)+\boldsymbol{E}_s\boldsymbol{v}_s(k),\tag{24}$$

$$\boldsymbol{y}_s(k)=\boldsymbol{C}_s\boldsymbol{x}_s(k),\tag{25}$$

is obtained, where for mode 1

$$\begin{gathered}\boldsymbol{A}_s=\begin{bmatrix}-\dfrac{1}{\tau_e} & 0 & 0\\ \dfrac{1}{J_e} & 1 & 0\\ 0 & 0 & 1\end{bmatrix},\\ \boldsymbol{B}_s=\begin{bmatrix}\dfrac{1}{\tau_e} & 0 & 0\\ 0 & \dfrac{(1+k_1)(1+k_2)}{k_1k_2i_qJ_e} & 0\\ 0 & -\dfrac{(1+k_1+k_2)(1+k_3)}{k_1k_2J_p} & \dfrac{1+k_3}{J_p}\end{bmatrix},\\ \boldsymbol{E}_s=\begin{bmatrix}0\\ 0\\ -\dfrac{1}{J_p}\end{bmatrix},\\ \boldsymbol{C}_s=\begin{bmatrix}1\\ 1\\ 1\end{bmatrix},\end{gathered}\tag{26}$$

and for mode 2

$$\begin{gathered}\boldsymbol{A}_s=\begin{bmatrix}-\dfrac{1}{\tau_e} & 0 & 0\\ \dfrac{1}{J_e} & 1 & 0\\ 0 & 0 & 1\end{bmatrix},\\ \boldsymbol{B}_s=\begin{bmatrix}\dfrac{1}{\tau_e} & 0 & 0\\ 0 & \dfrac{1+k_2}{k_1i_qJ_e} & \dfrac{1+k_2}{i_qJ_e}\\ 0 & -\dfrac{1+k_1+k_2}{k_1J_p} & -\dfrac{k_2}{J_p}\end{bmatrix},\\ \boldsymbol{E}_s=\begin{bmatrix}0\\ 0\\ -\dfrac{1}{J_p}\end{bmatrix},\\ \boldsymbol{C}_s=\begin{bmatrix}1\\ 1\\ 1\end{bmatrix}.\end{gathered}\tag{27}$$

2.3 Supervisory controller

The supervisory controller is designed to solve the energy management problem which can be viewed as a constrained non-linear dynamic optimisation problem. An NMPC technique is used here to address the problem online to minimise fuel consumption and maintain an ideal battery SOC. Also, rapid fluctuations of the control variables, engine torque, and engine speed, should be avoided which may cause the highly coupled system instability, thus, the objective function J_m to be optimised is defined by

$$\begin{aligned}J_m=&\sum_{i=0}^{N_p-1}(w_s(\mathrm{SOC}(k+i+1)-\mathrm{SOC}_r)^2+w_m(\dot{m}_f(k+i))^2)\\ &+\sum_{i=0}^{N_p-1}(w_t(\Delta T_{e\_d}(k+i))^2+w_w(\Delta\omega_{e\_d}(k+i))^2)\\ &+w_h(\mathrm{SOC}(k+N_p)),\end{aligned}\tag{28}$$

where N_p is the prediction horizon, SOC_r is the reference SOC value, and $\Delta\omega_{e\_d}$ are the incremental inputs of the demand engine torque and the demand engine speed, w_m, w_s, w_t, w_w and w_h are the penalty function weights, and $w_h(\mathrm{SOC}(k+N_p))$ is used here to penalise the deviation of SOC at the end of the prediction horizon from a reference value. The optimisation problem is subject to a set of inequality constraints arising from components' characteristics

$$\begin{aligned}&\mathrm{SOC}_{\min}\le\mathrm{SOC}\le\mathrm{SOC}_{\max},\\ &T_{e\_d\_\min}\le T_{e\_d}\le T_{e\_d\_\max},\\ &\Delta T_{e\_d\_\min}\le\Delta T_{e\_d}\le\Delta T_{e\_d\_\max},\\ &T_{MG1\_\min}\le T_{MG1}\le T_{MG1\_\max},\\ &T_{MG2\_\min}\le T_{MG2}\le T_{MG2\_\max},\\ &\omega_{e\_d\_\min}\le\omega_{e\_d}\le\omega_{e\_d\_\max},\\ &\Delta\omega_{e\_d\_\min}\le\Delta\omega_{e\_d}\le\Delta\omega_{e\_d\_\max},\\ &\omega_{MG1\_\min}\le\omega_{MG1}\le\omega_{MG1\_\max},\\ &\omega_{MG2\_\min}\le\omega_{MG2}\le\omega_{MG2\_\max},\end{aligned}\tag{29}$$

where $*_{\max}$ and $*_{\min}$ denote the maximum and minimum bounds which may be time-variant.

Therefore, based on eqs. (15)–(17), (28) and (29), the final constrained optimal control problem follows (formally):

$$\begin{gathered}\min_{u_m(k)}J_m(\boldsymbol{x}_m(k),\boldsymbol{u}_m(k))\\ s.t.:\begin{cases}\boldsymbol{x}_m(k+1)=f_m(\boldsymbol{x}_m(k),\boldsymbol{u}_m(k),\boldsymbol{v}_m(k)),\\ \boldsymbol{u}_{m\_\min}(k)\le\boldsymbol{u}_m(k)\le\boldsymbol{u}_{m\_\max}(k),\\ \boldsymbol{y}_{m\_\min}(k)\le\boldsymbol{y}_m(k)\le\boldsymbol{y}_{m\_\max}(k),\end{cases}\end{gathered}\tag{30}$$

where $\boldsymbol{u}_{m\_\min}$, $\boldsymbol{u}_{m\_\max}$, $\boldsymbol{y}_{m\_\min}$, and $\boldsymbol{y}_{m\_\max}$ are boundaries according to the constraints of eq. (29).

The non-linear receding horizon optimisation problem is solved with the sampling time interval t_{m_s} using forward dynamic programming. More specifically, in the supervisory controller, the following actions are performed at each sampling time (k) and are repeated by receding the prediction horizon on each step forward.

2.4 Coordinating controller

As shown in Figure 2, the coordinating controller is designed to apply the reference r_s, the demand engine torque T_{e_d}, the demand engine speed ω_{e_d}, and the demand output speed ω_{o_d}, to the plant using the manipulated variables T_{e_cmd}, T_{MG1_cmd}, and T_{MG2_cmd}. Since the prediction model eqs. (23)–(27) is linear, a standard LMPC method, with a quadratic cost function, is used. The optimisation problem can be formulated, in discrete time, as

$$\begin{aligned} \min_{u_s(k)} J_s = {} & \boldsymbol{Q}_s \sum_{i=0}^{N_s-1} (\boldsymbol{y}_s(k+i+1\mid k) - \boldsymbol{r}_s(k+i+1\mid k))^2 \\ & + \boldsymbol{R}_s \sum_{i=0}^{N_s-1} \Delta \boldsymbol{u}_s(k+i\mid k)^2 + \varepsilon^2 \\ s.t.: & \begin{cases} \boldsymbol{x}_s(k+1) = \boldsymbol{A}_s \boldsymbol{x}_s(k) + \boldsymbol{B}_s \boldsymbol{u}_s(k) + \boldsymbol{E}_s \boldsymbol{v}_s(k), \\ \boldsymbol{u}_{s\_\min}(k) \le \boldsymbol{u}_s(k) \le \boldsymbol{u}_{s\_\max}(k), \\ \boldsymbol{y}_{s\_\min}(k) \le \boldsymbol{y}_s(k) \le \boldsymbol{y}_{s\_\max}(k), \end{cases} \end{aligned} \tag{31}$$

where N_s is the prediction horizon of the coordinating controller, $\Delta \boldsymbol{u}_s$ are the incremental inputs, $\boldsymbol{Q}_s$ and $\boldsymbol{R}_s$ are the weighting factors, ε is the slack variable used to avoid infeasibility, and $\boldsymbol{u}_{s\_\min}$, $\boldsymbol{u}_{s\_\max}$, $\boldsymbol{y}_{s\_\min}$ and $\mathbf{y}_{s\_\max}$ are boundaries according to

$$\begin{aligned} & T_{e\_\min} \le T_e \le T_{e\_\max}, \\ & T_{e\_cmd\_\min} \le T_{e\_cmd} \le T_{e\_cmd\_\max}, \\ & \Delta T_{e\_cmd\_\min} \le \Delta T_{e\_cmd} \le \Delta T_{e\_cmd\_\max}, \\ & T_{MG1\_cmd\_\min} \le T_{MG1\_cmd} \le T_{MG1\_cmd\_\max}, \\ & \Delta T_{MG1\_cmd\_\min} \le \Delta T_{MG1\_cmd} \le \Delta T_{MG1\_cmd\_\max}, \\ & T_{MG2\_cmd\_\min} \le T_{MG2\_cmd} \le T_{MG2\_cmd\_\max}, \\ & \Delta T_{MG2\_cmd\_\min} \le \Delta T_{MG2\_cmd} \le \Delta T_{MG2\_cmd\_\max}, \\ & \omega_{e\_\min} \le \omega_e \le \omega_{e\_\max}. \end{aligned} \tag{32}$$

In the coordinating controller, a quadratic programming technique is used to solve the optimisation problem eq. (31) online with sampling time interval t_{s_s}. At each sampling time (k), the following actions are performed and repeated at subsequent sampling times.

3 Markov-chain stochastic prediction

In the supervisory controller of the EMS, the optimisation task is solved online with the predictive velocity demand and torque demand provided in each receding horizon. Although, in the NMPC algorithm, only the first element of the control sequence is applied, a good prediction of the future load demand can improve controller performance. Here, assuming that no radar, global positioning system, or similar devices are available and future driving profiles are completely unknown, a novel methodology based on adaptive Markov chains is proposed to predict the future load demand in the short-term.

Markov chains are series of transition probabilities from one of the limited states at instant k to another state of all possible states at instant $k+1$. In this paper, based on the different values of the driver's pedal position, $\alpha\leqslant20\%$, $20\%<\alpha\leqslant40\%$, $40\%<\alpha\leqslant60\%$, $60\%<\alpha\leqslant80\%$ and $80\%<\alpha\leqslant100\%$, five corresponding one-stage Markov-chain models are developed. The states of each Markov chain are defined on a discrete-valued domain given by vehicle velocity $\overline{V}$ (0–30 m/s) and vehicle acceleration $\bar{a}$ (–1.5–1.5 m/s^2). Suppose the vehicle velocity and the vehicle acceleration are discretised into p and q intervals and indexed by i and j, respectively, then each Markov-chain process is defined by a transition probability matrix $\boldsymbol{T}$ with

$$T_{ij} = \Pr\left[a_{k+n+1} = \bar{a}_j \mid V_{k+n} = \overline{V}_i\right], \tag{33}$$

where $i \in \{1,\ldots,p\}$, $j \in \{1,\ldots,q\}$, $n \in \{1,\ldots,N_p\}$ are time-based indices in the receding horizon of the supervisory controller and T_{ij} is the probability of vehicle acceleration transitioning to a_j in the next time step, given $V_{k+n}=V_i$ in the current time step.

At first, the transition probabilities of the Markov chains given by eq. (33) are computed from a comprehensive dataset considering both highway and urban driving scenarios which are extracted from standard driving cycles (HWFET, UDDS, FTP-72, US06, NEDC). Supposing that $p=30$ and $q=20$, an illustration of the transition probabilities in the Markov chain models, for driver pedal positions $\alpha\leqslant20\%$, is provided in Figure 6.

Since the driving conditions change over time, the Markov chains need to be updated online to adapt to each new situation; at any step k, when $a_k=a_j$ and $V_{k-1}=V_i$, the Markov-chain transition probability is updated by

$$T(k)_{ij} = T(k-1)_{ij} + \lambda, \tag{34}$$

where λ is a predefined constant. The whole column of the matrix in which the transition is observed is also updated. The procedure for the use of this adaptive Markov chain algorithm is summarised in Algorithm 1.

Algorithm 1 Adaptive Markov chain algorithm

```
At any step $k$;
get measurements $V_{k-1}$ and $V_k$;
calculates $a_k$;
if $a_k=a_j$ and $V_{k-1}$ and $V_i$
set $T(k)_{ij}=T(k-1)_{ij}+\lambda$;
for all $s \in \{1,\ldots,q\}$ and $s\neq j$ do;
set $T(k)_{is}=T(k-1)_{is}-\lambda/q$;
end for.
```

In Figure 7, the effect of the adaptive algorithm is shown: the original Markov-chain transition probabilities, shown in Figure 6, are updated online and after running two more urban driving cycles (UDDS), the matrix is developed as shown in Figure 7.

Note that the assumption that the vehicle velocity and acceleration satisfy the Markov property is validated by computing the residuals between the values acquired from the

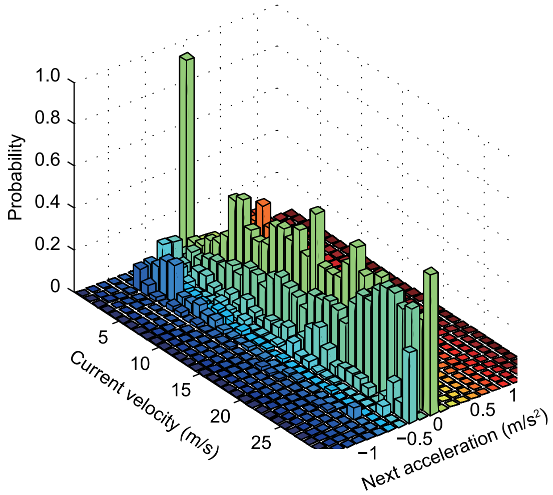

Figure 6 (Color online) Markov-chain transition probabilities for driver pedal positions $\alpha \leqslant 20\%$.

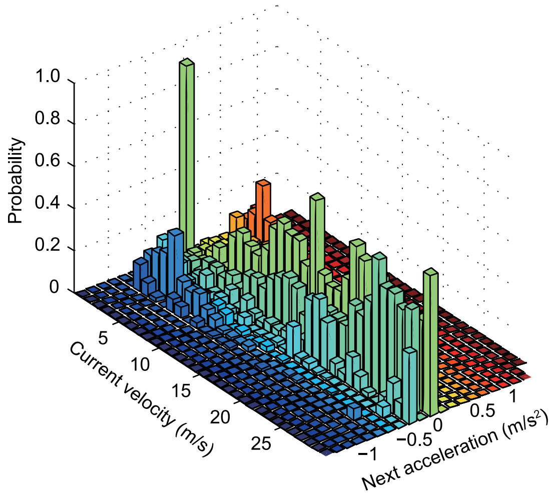

Figure 7 (Color online) Markov-chain transition probabilities for driver pedal positions $\alpha \leqslant 20\%$ after adaption.

model and the driving cycles. The autocorrelation of the residuals exceeds the 95% confidence interval which statistically confirms the assumption [36].

So, with the adaptive Markov chains, the future load demand over the predictive horizon of the supervisory controller can be obtained, as shown in Figure 8. In detail, with the sampling time interval t_{m_s}, the following actions are performed at each sampling time (k)

(1) Based on the driver's current pedal position, a different Markov-chain model is selected, and the model is assumed to stay the same over the predictive horizon.

(2) The velocity demand is obtained by the Markov-chain model eq. (33) and the torque demand is calculated by the driver model which can be described by

$$m\dot{V} = \frac{T_{dmd}}{r_w} - \frac{1}{2}\rho C_d A_f V^2 - \mu mg\cos(\theta) - mg\sin(\theta), \qquad (35)$$

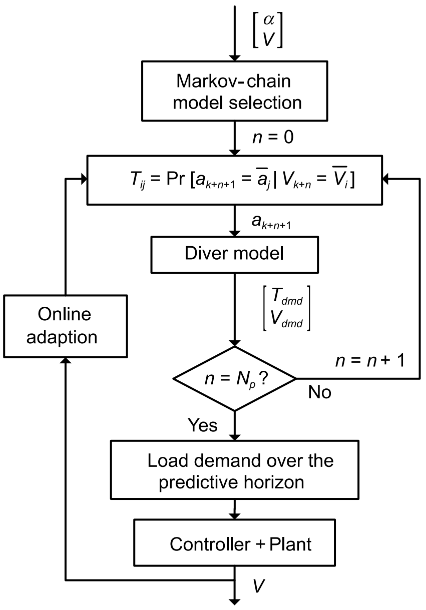
$\begin{bmatrix}\alpha \\ V\end{bmatrix}$
Markov-chain model selection
$n = 0$
$T_{ij} = \Pr\,[a_{k+n+1} = \bar{a}_j \,|\, V_{k+n} = \bar{V}_i]$
a_{k+n+1}
Diver model
$\begin{bmatrix}T_{dmd} \\ V_{dmd}\end{bmatrix}$
Online adaption
$n = N_p$?
$n = n + 1$
No
Yes
Load demand over the predictive horizon
Controller + Plant
V

Figure 8 Flowchart of the Markov-chain stochastic prediction.

where m is the vehicle mass; r_w is the radius of wheels; A_f is the frontal area of the vehicle; ρ is the density of air; μ is the friction coefficient; C_d is the drag coefficient; g is the acceleration due to gravity, and θ is the gradient of the road.

(3) The load demands over the whole predictive horizon are obtained and used in the controller.

(4) The actual vehicle velocity is measured and the Markov-chain transition probability matrices are updated online with Algorithm 1.

4 Results and discussion

In this section, the effectiveness of the proposed EMS for a dual-mode HEV is validated by simulations and test-bed experiments. MATLAB™/Simulink software and Sim power systems modules are used to model the mechanical and electrical elements of the HEV for closed-loop simulations: a test-bed is established (Figure 9), with which to validate the proposed EMS in real-world implementation.

The hybrid powertrain, comprising a 160 kW diesel engine (limited to 120 kW) and two 60 kW MGs, is established at the test-bed and modelled in a 2500 kg SUV for simulation. The battery is a lithium-iron-phosphate chemistry based battery with 76 Ah capacity and 360 V nominal voltage (100 series with two parallel connected cells). The test-bed dynamometer is connected over a torque-sensing shaft to apply the load. The proposed EMS is implemented on a RapidECU (an on-board prototype electrical control unit) platform, which is also used for test-bed measurements. A CAN interface is used to exchange relevant information from the low-level test-bed control system. Other key parameters of the vehicle are listed in Table 1.

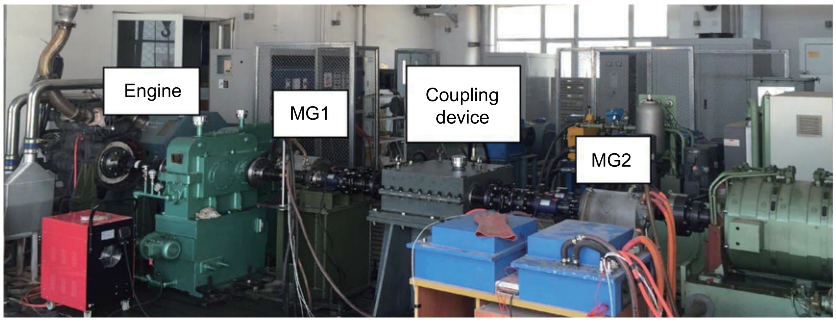

Figure 9 (Color online) Experimental test-bed.

Table 2 summarises the user-defined controller parameters: since the main purposes of the EMS are to minimise fuel consumption and maintain an ideal battery SOC, the penalties w_s and w_m are significantly higher than the others in eq. (30). Also since the MGs' dynamic response is much faster than that of the engine, to achieve good coordinating control performance, the output penalty $\boldsymbol{Q}_s$ is chosen so as to be higher than the control penalty $\boldsymbol{R}_s$.

4.1 Simulation results

Simulations are conducted to evaluate the performance of the proposed EMS and compare it with other methods. As shown in Figure 10, the proposed EMS is tested in a typical real-world driving cycle. The mode switch process is assumed to be instantaneous and the mode switch strategy is based on vehicle velocity and driver pedal position. It can be seen that battery SOC is maintained close to its reference value over the cycle while the driving requirements are also satisfied.

Table 1 Key parameters of the vehicle

Parameter	Value	Description
r_w	0.388 m	radius of wheels
A_f	3.24 m^2	frontal area of the vehicle
μ	0.015	friction coefficient
C_d	0.5	drag coefficient
i_f	4.24	gear ratio of final drive
i_q	0.56	gear ratio of front drive
J_e	0.25 kg m^2	engine rotational inertia
J_p	16.7 kg m^2	powertrain's total inertia
k_1	2.13	PG1's inherent parameter
k_2	2.13	PG2's inherent parameter
k_3	2.33	PG3's inherent parameter

Table 2 Used user-defined controller parameters

Parameter	Nominal value	Description
SOC$_0$	0.65	initial SOC
SOC$_r$	0.65	SOC reference
N_p	5	supervisory prediction horizon
w_s	20000	SOC penalty
w_m	0.01	fuel consumption penalty
w_t	10^{-4}	engine torque rate penalty
w_w	10^{-6}	engine speed rate penalty
w_h	100	Final SOC penalty
N_s	10	coordinating prediction horizon
$\boldsymbol{Q}_s$	diag([5 10 10])	output penalty
$\boldsymbol{R}_s$	diag([2 1 1])	control penalty

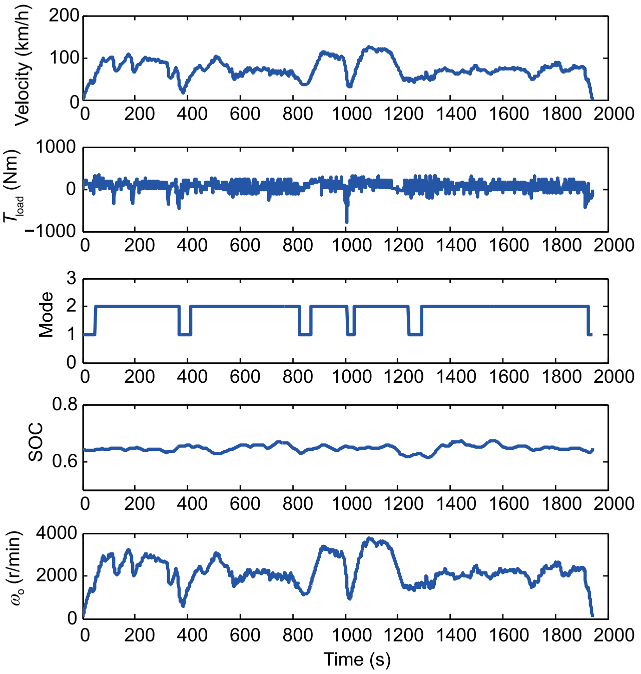

Figure 10 (Color online) Simulation results of vehicle velocity, load disturbance, mode, SOC, and output speed with the proposed EMS.

As shown in Figure 11, to make the engine work in a more efficient way, the speed and torque of the engine are regulated by the MGs to a certain extent. With the help of the MGs, the fluctuations in engine speed are small; however, the engine torque still varies significantly according to the driving condition. This is because the requested driving power is too large for the MGs to compensate for the engine state.

To demonstrate the proposed Markov-chain-MPC-based EMS (MMPC) performance quantitatively, simulations over different cycles with different strategies were undertaken: the proposed EMS was compared with a frozen MPC-based strategy (FMPC) where the load demand is assumed constant over the supervisory prediction horizon, a rule-based strategy

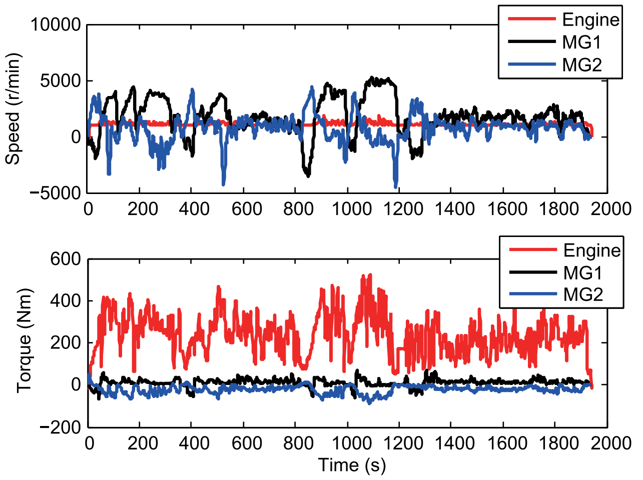

Figure 11 (Color online) Simulation results of speeds and torques of the engine, MG1, and MG2 with the proposed EMS.

which is taken as the benchmark, and a DP-based strategy which is adopted to verify the approximate global optimal performance of the proposed EMS. The results are given below.

In Figure 12, the SOC trajectories with different strategies are shown: the MMPC and FMPC strategies yield locally optimal solutions for each control horizon which are similar to the global optimisation of DP and differ noticeably from the poor rule-based strategy. In Figure 13, the engine operating points with different strategies are shown: it can be seen that, with MMPC and FMPC, the engine always works efficiently near its optimum fuel economy curve, which is similar to the results from the DP-based strategy and improved upon those achieved with the rule-based strategy.

The results of fuel economy when using MMPC, FMPC, DP, and Rule over different driving cycles are shown in Table 3, in terms of norm of SOC difference between the end and the beginning of the driving cycle (ΔSOC), fuel consumption, equivalent fuel consumption, and equivalent fuel consumption improvement with respect to the rule-based strategy. The equivalent fuel consumption is used for fair comparison, where the ΔSOC needs to be converted into fuel and added to the fuel consumption, as calculated by

$$E_{c,s} = F_{c,s} - \varphi \cdot \Delta\mathrm{SOC}_{c,s}, \tag{36}$$

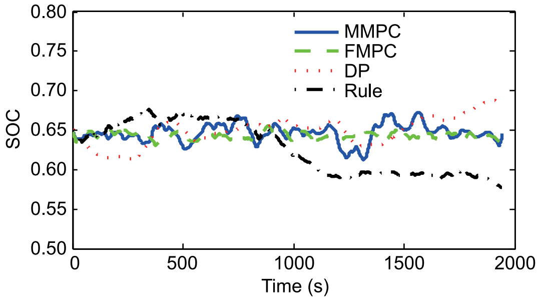

Figure 12 (Color online) Simulation results of SOC with different strategies.

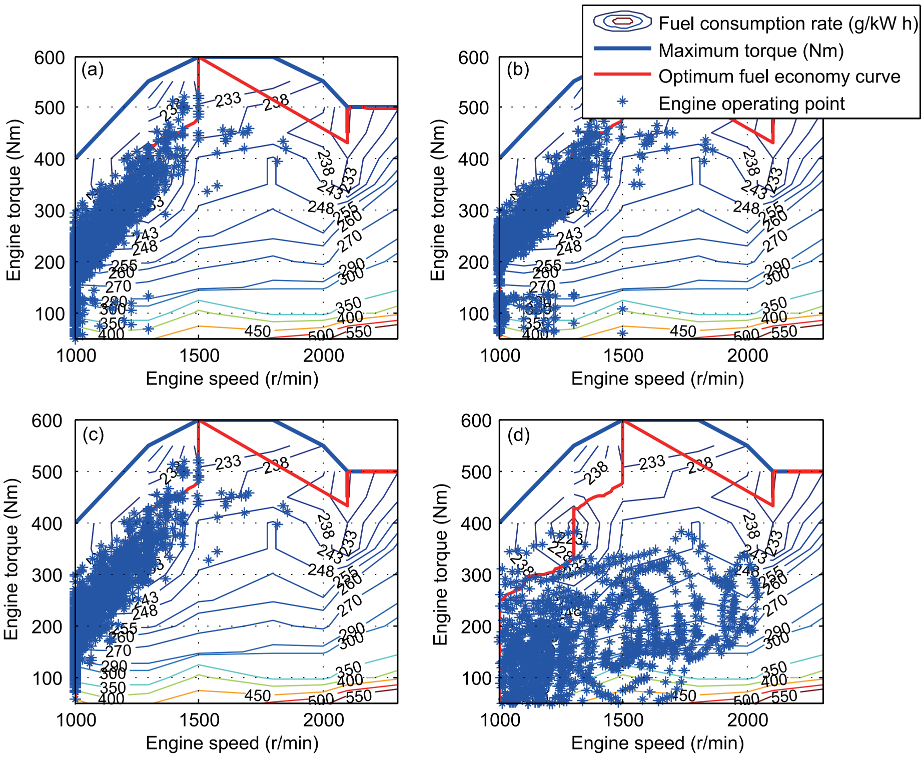

Figure 13 (Color online) Simulation results of engine operating points with different strategies. (a) MMPC; (b) FMPC; (c) DP; (d) Rule.

Table 3 A comparison of fuel economies

	Strategy	ΔSOC	Fuel consumption (L)	Equivalent fuel consumption (L)	Improvement (%)
Real-world driving cycle	MMPC	–0.01	5.15	5.20	10.63
	FMPC	–0.02	5.21	5.32	8.59
	DP	0.04	5.12	4.90	15.81
	Rule	–0.07	5.44	5.82	–
HWFET	MMPC	0.03	3.04	2.88	15.31
	FMPC	0.03	3.17	3.01	11.53
	DP	0.01	2.81	2.76	18.97
	Rule	–0.04	3.18	3.40	–
UDDS	MMPC	0.02	3.75	3.48	16.82
	FMPC	0.05	3.94	3.67	12.25
	DP	0.02	3.36	3.25	22.14
	Rule	0.03	4.34	4.18	–

where $E_{c,s}$, $F_{c,s}$ and $\Delta\mathrm{SOC}_{c,s}$ are the equivalent fuel consumption, the fuel consumption, and the SOC difference for cycle c with strategy s, respectively. In eq. (36), φ is the coefficient that converts battery charge into fuel via their prices. Here, the current price of diesel is 5.54 Yuan/L while the price of electrical energy is 0.9 Yuan/kW h. The results show that MMPC improves fuel economy by nearly 3% with respect to FMPC by taking advantage of future load predictions. The MMPC achieves a 10.63% improvement in fuel economy, compared with the rule-based strategy in real-world driving conditions, and about 15.31% and 16.82% improvement in HWFET and UDDS, respectively. It also can be observed that the results of MMPC are close to the optimal results of DP.

4.2 Experimental results

To validate the real-world implementation of the proposed EMS, test-bed experiments are conducted along unknown real driving cycles. Some of the results with FMPC and MMPC strategies are presented below.

In the first sub-plot of Figure 14, the pedal position trajectory (unknown, a priori) is depicted: the current load, shown in the second sub-plot, is measured by the sensor and transferred to the EMS as a disturbance. Then the engine torque and the engine speed are depicted. It can be seen in the marked region, due to the prediction, the engine speed with MMPC decreased before that with FMPC. Also, the engine torque with MMPC is less dynamic than that with FMPC. This shows that the prediction also has a smoothing influence on the powertrain. In sub-plot five, the output speed is depicted, which shows that the output with MMPC better reflects driver expectations than that with FMPC. The last sub-plot shows the SOC trajectories which are both maintained near their reference values.

In Figure 15, the comparison of the engine operating points with MMPC and FMPC is shown. It can be seen that the distribution of the engine operating points with MMPC is better than that with FMPC.

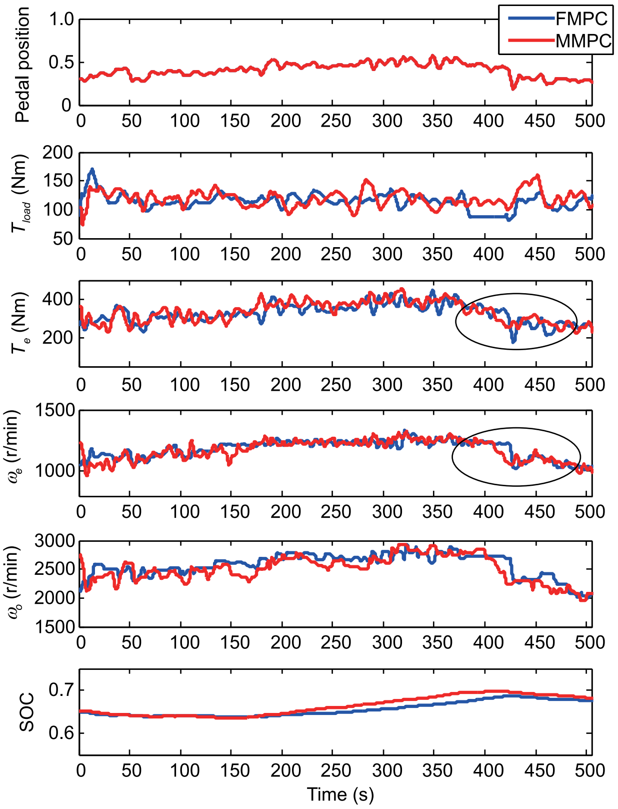

Figure 14 (Color online) Experimental results of pedal position, load disturbance, engine torque, engine speed, output speed, and SOC with FMPC and MMPC.

In Table 4, the comparison of fuel economy with MMPC and FMPC is presented. The results demonstrate the capability of MMPC to adapt to different real-world driving cycles by online adaptation. The equivalent fuel consumption yielded by MMPC is notably improved with respect to that with FMPC.

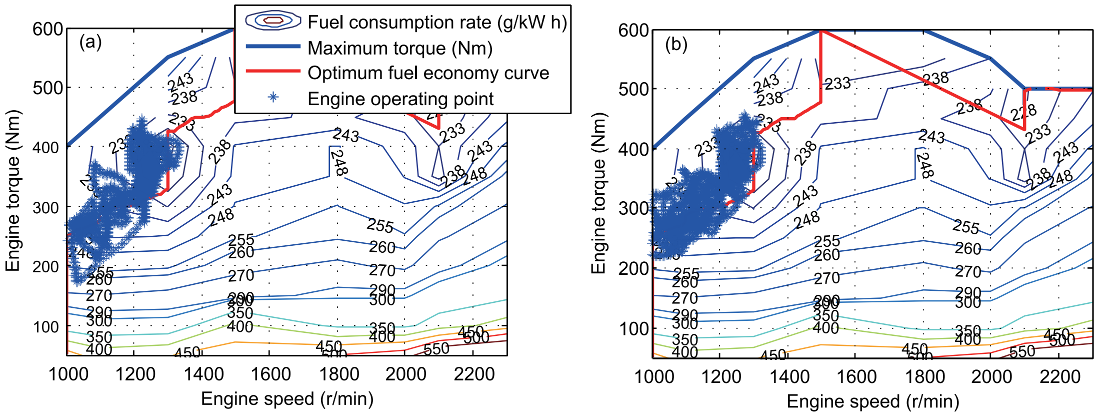

Figure 15 (Color online) Experimental results of engine operating points with FMPC (a) and MMPC (b).

Table 4 A fuel economy comparison

	Strategy	ΔSOC	Fuel consumption (L)	Equivalent fuel consumption (L)	Improvement (%)
Real world driving cycle 1	MMPC	0.03	2.14	1.98	4.11
	FMPC	0.03	2.22	2.06	–
Real world driving cycle 2	MMPC	0.02	2.80	2.69	5.37
	FMPC	0.03	3.00	2.84	–

5 Conclusions

In this paper, a model-predictive-control-based energy management strategy with adaptive Markov-chain prediction is proposed for a dual-mode hybrid electric vehicle. In this approach, future load demand is predicted online on the basis of Markov chains without a priori driving cycle knowledge. The Markov chains are built offline from standard driving cycles and updated online to adapt to new situations. Then two model predictive controllers with different sampling time intervals are designed. The NMPC-based supervisory controller is designed to improve fuel economy and maintain an ideal battery SOC with the predicted future load: the LMPC-based coordinating controller is designed to deal with the fast dynamics thereof. The effectiveness, and real-time capability, of the proposed EMS are validated by simulations and test-bed experiments. A maximum of 16% fuel economy improvement is achieved by use of the proposed EMS compared to that with the use of a rule-based strategy.

This work was supported by the National Natural Science Foundation of China (Grant Nos. 51005017, 51575043 & U1564210).

1 Bayindir K Ç, Gözüküçük M A, Teke A. A comprehensive overview of hybrid electric vehicle: Powertrain configurations, powertrain control techniques and electronic control units. Energy Conv Manage, 2011, 52: 1305–1313

2 Kim J, Kim T, Min B, et al. Mode control strategy for a two-mode hybrid electric vehicle using electrically variable transmission (EVT) and fixed-gear mode. IEEE Trans Veh Tech, 2011, 60: 793–803

3 Qin K J, Wang E H, Zhao H, et al. Development and experimental validation of a novel hybrid powertrain with dual planetary gear sets for transit bus applications. Sci China Tech Sci, 2015, 58: 2085–2096

4 Sciarretta A, Guzzella L. Control of hybrid electric vehicles. IEEE Control Syst Mag, 2007, 27: 60–70

5 Liu J M, Peng H. Modeling and control of a power-split hybrid vehicle. IEEE Trans Contr Syst Tech, 2008, 16: 1242–1251

6 Xiang C, Wang Y, Hu S, et al. A new topology and control strategy for a hybrid battery-ultracapacitor energy storage system. Energies, 2014, 7: 2874–2896

7 Castaings A, Lhomme W, Trigui R, et al. Comparison of energy management strategies of a battery/supercapacitors system for electric vehicle under real-time constraints. Appl Energy, 2016, 163: 190–200

8 Sorrentino M, Rizzo G, Arsie I. Analysis of a rule-based control strategy for on-board energy management of series hybrid vehicles. Control Eng Practice, 2011, 19: 1433–1441

9 Zhang S, Xiong R. Adaptive energy management of a plug-in hybrid electric vehicle based on driving pattern recognition and dynamic programming. Appl Energy, 2015, 155: 68–78

10 Li H Z, Li L, He L, et al. PID plus fuzzy logic method for torque control in traction control system. IntJ Automot Tech, 2012, 13: 441–450

11 Hemi H, Ghouili J, Cheriti A. A real time fuzzy logic power management strategy for a fuel cell vehicle. Energy Conv Manage, 2014, 80: 63–70

12 Sun Y, Chen Z, Yan B J, et al. A learning method for energy optimization of the plug-in hybrid electric bus. Sci China Tech Sci, 2015, 58: 1242–1249

13 Xia C Y, Zhang C. Real-time optimization power-split strategy for hybrid electric vehicles. Sci China Tech Sci, 2016, 59: 814–824

14 Shams-Zahraei M, Kouzani A Z, Kutter S, et al. Integrated thermal and energy management of plug-in hybrid electric vehicles. J Power Sources, 2012, 216: 237–248

15 Zhang Y H, Jiao X H, Li L, et al. A hybrid dynamic programming-rule based algorithm for real-time energy optimization of plug-in hybrid electric bus. Sci China Tech Sci, 2014, 57: 2542–2550

16 Chen Z, Xia B, You C, et al. A novel energy management method for series plug-in hybrid electric vehicles. Appl Energy, 2015, 145: 172–179

17 Chen S Y, Hung Y H, Wu C H, et al. Optimal energy management of a hybrid electric powertrain system using improved particle swarm optimization. Appl Energy, 2015, 160: 132–145

18 Zheng C, Xu G, Xu K, et al. An energy management approach of hybrid vehicles using traffic preview information for energy saving. Energy Conv Manage, 2015, 105: 462–470

19 Xiong W. Series-parallel hybrid vehicle control strategy design and optimization using real-valued genetic algorithm. Chin J Mech Eug, 2009, 22: 862–868

20 Chen Z, Mi C C, Jun Xu C C, et al. Energy management for a power-split plug-in hybrid electric vehicle based on dynamic programming and neural networks. IEEE Trans Veh Tech, 2014, 63: 1567–1580

21 Li L, Yang C, Zhang Y H, et al. Correctional DP-based energy management strategy of plug-in hybrid electric bus for city-bus-route. IEEE Trans Veh Tech, 2015, 64: 2792–2803

22 Chasse A, Sciarretta A. Supervisory control of hybrid powertrains: An experimental benchmark of offline optimization and online energy management. Control Eng Pract, 2011, 19: 1253–1265

23 Unger J, Kozek M, Jakubek S. Nonlinear model predictive energy management controller with load and cycle prediction for non-road HEV. Control Eng Pract, 2015, 36: 120–132

24 Laldin O, Moshirvaziri M, Trescases O. Predictive algorithm for optimizing power flow in hybrid ultracapacitor/battery storage systems for light electric vehicles. IEEE Trans Power Electron, 2013, 28: 3882–3895

25 He L, Shen T, Yu L, et al. A model-predictive-control-based torque demand control approach for parallel hybrid powertrains. IEEE Trans Veh Tech, 2013, 62: 1041–1052

26 Zhang J, Shen T. Real-time fuel economy optimization with nonlinear MPC for PHEVs. IEEE Trans Contr Syst Tech, 2016, 24: 2167–2175

27 Borhan H, Vahidi A, Phillips A M, et al. MPC-based energy management of a power-split hybrid electric vehicle. IEEE Trans Contr Syst Tech, 2012, 20: 593–603

28 Sun C, Hu X, Moura S J, et al. Velocity predictors for predictive energy management in hybrid electric vehicles. IEEE Trans Contr Syst Tech, 2015, 23: 1197–1204

29 Li L, You S, Yang C, et al. Driving-behavior-aware stochastic model predictive control for plug-in hybrid electric buses. Appl Energy, 2016, 162: 868–879

30 Bender F A, Kaszynski M, Sawodny O. Drive cycle prediction and energy management optimization for hybrid hydraulic vehicles. IEEE Trans Veh Tech, 2013, 62: 3581–3592

31 Wang W, Xiang C, Liu H, et al. A model-predictive-control-based power management strategy for a power-split electromechanical transmission. Proc Inst Mech Eng D-J Auto Eng, 2016: 1–15

32 He H, Xiong R, Guo H, et al. Comparison study on the battery models used for the energy management of batteries in electric vehicles. Energy Conv Manage, 2012, 64: 113–121

33 Fekri S, Assadian F. The role and use of robust multivariable control in hybrid electric vehicle energy management-Part II: Application. In: Proceedings of IEEE International Conference on Control Applications. Dubrovnik: IEEE, 2012. 317–322

34 Liu Y H, Li T, Yang Y Y, et al. Estimation of tire-road friction coefficient based on combined APF-IEKF and iteration algorithm. Mech Syst Signal Process, 2017, 88: 25–35

35 Ma B, Liu Y, Gao Y, et al. Estimation of vehicle sideslip angle based on steering torque. Int J Adv Manuf Tech, 2016

36 Brockwell P J, Davis R A. Time Series: Theory and Methods. New York: Springer, 1998