Design and evaluation of a real-time fuel-optimal control system for series hybrid electric vehicles

Reza Sharif Razavian*, Amir Taghavipour, Nasser L. Azad and John McPhee

Department of Systems Design Engineering,
University of Waterloo,
200 University Ave. W., Waterloo, Ontario, N2L 3G1 Canada
E-mail: rsharifr@uwaterloo.ca
E-mail: athaghavi@uwaterloo.ca
E-mail: nlashgar@uwaterloo.ca
E-mail: mcphee@uwaterloo.ca
*Corresponding author

Abstract: We propose a real-time optimal controller that will reduce fuel consumption in a series hybrid electric vehicle (HEV). This real-time drive cycle-independent controller is designed using a control-oriented model and Pontryagin's minimum principle for an off-line optimisation problem, and is shown to be optimal in real-time applications. Like other proposed controllers in the literature, this controller still requires some information about future driving conditions, but the amount of information is reduced. Although the controller design procedure explained here is based on a series HEV with NiMH battery as the electric energy storage, the same procedure can be used to find the supervisory controller for a series HEV with an ultra-capacitor. To evaluate the performance of the model-based controller, it is coupled to a high-fidelity series HEV model that includes physics-based component models and low-level controllers. The simulation results show that the simplified control-oriented model is accurate enough in predicting real vehicle behaviour, and final fuel consumption can be reduced using the model-based controller. Such a reduction in HEVs fuel consumption will significantly contribute to nationwide fuel saving.

Keywords: optimal control; model-based control; series hybrid electric vehicle; high-fidelity simulation; low-level controller.

Biographical notes: Reza Sharif Razavian received his BSc in Mechanical Engineering from Sharif University of Technology, Tehran, Iran. He is currently an MASc candidate in the Department of Systems Design Engineering at the University of Waterloo, and his area of research is in modelling and control of hybrid electric vehicles.

Amir Taghavipour received his BSc and MSc in Mechanical Engineering from Sharif University of Technology, Tehran, Iran, in 2007 and 2009. He is currently a PhD student at University of Waterloo, Department of Systems Design Engineering. His research interests are automotive powertrain control, hybrid vehicle power management strategy design and optimisation.

Nasser L. Azad is an Assistant Professor in Systems Design Engineering at University of Waterloo. His primary research interests lie in modelling, estimation and control of complex dynamic mechanical and multi-domain physical systems, with special emphasis on advanced modelling and model reduction methods, sensitivity analysis techniques, non-linear and optimal control, with applications to advanced vehicle systems, such as modern automotive powertrains and vehicle dynamics control systems.

John McPhee is a Professor in Systems Design Engineering at the University of Waterloo, Canada, and the NSERC/Toyota/Maplesoft Industrial Research Chair in Mathematics-based Modelling and Design. His main area of research is multi-body system dynamics, with principal application to the analysis and design of vehicles, mechatronic devices, biomechanical systems, and sports equipment. He is a Fellow of the American Society of Mechanical Engineers and the Canadian Academy of Engineering. He was the Executive Director of the Waterloo Centre for Automotive Research until 2009, after which he spent a sabbatical year at the Toyota Technical Center in Ann Arbor, Michigan.

1 Introduction

Electric vehicles (EVs) show strong potential to be the major form of transportation in the future, but the automotive industry is facing difficulties in developing EVs. High price and limited drive range are the major drawbacks of these vehicles. Improving battery and fuel cell technologies will overcome these drawbacks. For now, as a short-term solution for the problems of fuel consumption and emissions by conventional vehicles, hybrid electric vehicles (HEVs) are among the best candidates. They still rely on fossil fuel and internal combustion engines, but for the following reasons, they can help in reducing fuel consumption and emissions:

1 it is possible to use smaller (and more efficient) engines due to the assistance of electrical propulsion systems

2 HEV electrical storage can be charged using an external source (in plug-in HEVs), therefore reducing fuel consumption

3 the presence of an electrical path in the powertrain allows the vehicle to capture part of its kinetic energy during braking

4 extra degrees of freedom in the powertrain allow the engine to work at higher efficiency points.

The last two points are especially important as there should be an accurate plan (the so called *supervisory controller* in the literature) to determine the proper amount of power to be generated by either of the two onboard sources. The plan should command each component in such a way that the fuel consumption and/or emission is minimised when the driver command is met and physical constraints of the system are not violated. In the early stages of the development of HEVs, rule-based supervisory controllers were used; these plans, although simple to implement, do not necessarily result in optimal behaviour, and are difficult to tune. Studies show that a 3% reduction in HEVs fuel consumption will help to save 6.5 million gallons of gas annually in the USA (Gonder, 2008). Therefore,

model-based controllers have become more popular in recent years (Sciarretta and Guzzella, 2007). Application of model-based controllers is widely studied in the literature, and numerous methods have been presented to find the optimal supervisory controller.

Among these model-based controllers are brute-force numerical methods that give the global optimal solution to the problem. Dynamic programming (Serrao et al., 2011; Ngo et al., 2010; Musardo et al., 2005; Lin et al., 2003; Boyali and Guvenc, 2010), particle swarm optimisation (PSO) (Lin et al., 2010), and genetic algorithm (GA) (Piccolo et al., 2001) are examples of such methods. These methods are not readily applicable for control applications because of their non-causal nature that requires a priori knowledge of the driver command for the whole mission. They are also computationally expensive and are optimal only for the drive cycle upon which they are developed. Thus, more applicable solutions such as stochastic dynamic programming (SDP) (Li et al., 2007; Lin et al., 2004), game theory (GT) (Chin and Jafari, 2010; Dextreit et al., 2008; Soltis and Chen, 2003), and model predictive control (MPC) (Borhan et al., 2010; Taghavipour et al., in press; Sampathnarayanan et al., 2009) are used for real-time control. With these methods, causal near-optimal controllers are developed that can be used in real time. They still require some information about future driving condition, but can compensate for different conditions and provide sub-optimal yet satisfactory results (Taghavipour et al., in press). These methods are still computationally costly and require special approaches to reduce the computational time. An example of such an approach is the combination of analytical and numerical methods (Ngo et al., 2010).

Pontryagin's minimum principle (PMP), as a branch of optimal control theory, has been proven useful in this application (Serrao et al., 2011; Ngo et al., 2010; Musardo et al., 2005; Lin et al., 2003, 2004, 2010; Boyali and Guvenc, 2010; Piccolo et al., 2001; Li et al., 2007; Chin and Jafari, 2010; Dextreit et al., 2008; Soltis and Chen, 2003; Borhan et al., 2010; Taghavipour et al., in press; Sampathnarayanan et al., 2009; Kim et al., 2011a, 2011b; Cipollone and Sciarretta, 2006; Serrao and Rizzoni, 2008). It reduces the integral optimisation over the whole drive cycle to an instantaneous minimisation of the Hamiltonian (Geering, 2007), which in turn is reduced to the tuning of only a few parameters (Kim et al., 2011a, 2011b; Cipollone and Sciarretta, 2006; Serrao and Rizzoni, 2008; Stockar et al., 2010). In one degree-of-freedom HEVs (series and parallel architectures for example), only one parameter needs to be tuned based on the driving condition.

The equivalence consumption minimisation strategy (ECMS) is another promising method (Serrao et al., 2011; Musardo et al., 2005; Ambuhl and Guzzella, 2009; Serrao et al., 2009; Sciarretta et al., 2004). In ECMS, the battery power is interpreted as an imaginary fuel consumption rate. The battery power is converted to fuel consumption using an equivalence factor, S. Then the optimal control, u^*, is chosen so that the total fuel consumption rate in (1) is minimised.

$$u^* = \arg\min\left\{\dot{m} + SP_{battery}\right\} \tag{1}$$

In this relation, $\dot{m}$ and $SP_{battery}$ are the actual and the imaginary fuel consumption rates, respectively. It has been shown that if the value of the equivalence factor is chosen properly, the outcome of the ECMS is optimal (Serrao et al., 2011; Serrao et al., 2009). In these cases, the total fuel consumption in (1) is similar to the Hamiltonian introduced by

the PMP, and the equivalence factor is tightly related to the costates (Lagrange multipliers). Thus, the optimal behaviour of the controller is based on finding an optimal equivalence factor.

The main contributions of this paper include a simple but mathematically optimal feedback controller as the supervisory controller, as well as a way to tune the controller parameters based on certain information about future driving conditions. Although the latter does not remove the necessity for information about future driving conditions, it reduces the amount of information required, and still gives near-optimal results. Like all other model-based controllers, this controller is based on a simplified control-oriented model; hence, the validation process is of great importance. In this work, the performance of the designed controller is evaluated by applying it to a high-fidelity physics-based HEV model (Dao et al., 2010, 2011) developed in the MapleSim environment.

The organisation of this paper is as follows. In Section 2, the simple model upon which the controller is built is discussed. Next, Section 3 defines the optimisation problem and presents the solution. Sections 4 and 5 provide details of the real-time optimal controller and the tuning method. Section 6 generalises the solution to series HEVs with an ultra-capacitor (UC). The discussion about the high-fidelity model and low-level controllers is presented in Sections 7 and 8. Results and discussions conclude the paper.

2   Control-oriented model

The mathematical representation of the hybrid powertrain is the core of the model-based controller design. It is essential that this *control-oriented* model be simple enough so that the computation time remains within real-time requirements. At the same time, this model should be able to represent the vehicle accurately enough to capture important characteristics of the powertrain.

In this work, a series HEV is studied (Figure 1). For designing the controller, a backward quasi-static model for the powertrain is used to calculate the required power based on the vehicle's velocity. This power is then used as the input to the optimisation problem.

Figure 1   Schematic of a series HEV
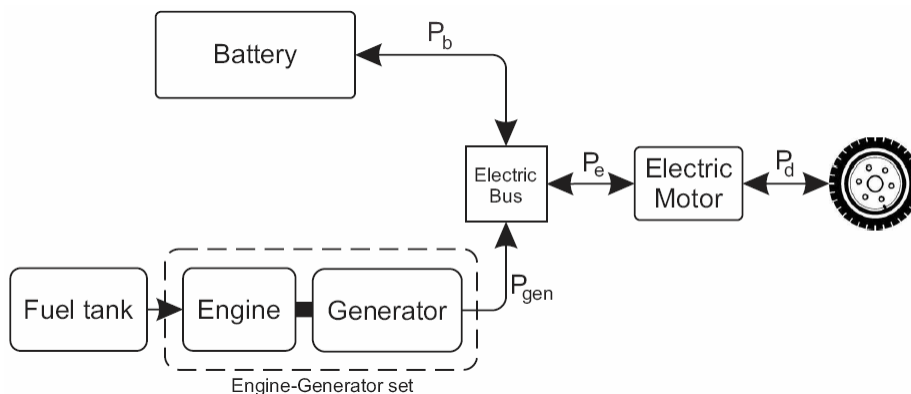
The longitudinal vehicle dynamics is modelled as (2).

$$m_v a_x = f_T - \left(f_D + f_R + m_v g \sin(\gamma) \right) \tag{2}$$

To find the power demand, (3) is used with f_D and f_R, defined in (4) and (5), respectively.

$$P_d = v_x f_T = v_x \left(m_v a_x + f_D + f_R + m_v g \sin(\gamma) \right) \tag{3}$$

$$f_D = \frac{1}{2} \rho v_x^2 A C_D \tag{4}$$

$$f_R = m_v g \cos(\gamma) f_{rr} \tag{5}$$

In the above equations, m_v is the vehicle mass; v_x and a_x are the longitudinal velocity and acceleration respectively; f_T is the traction force, resulting from the torque on the wheels; f_D is the aerodynamic drag force; f_R is the equivalent rolling resistance of all wheels, f_{rr} is the rolling resistance coefficient, and the term $m_v g \sin(\gamma)$ is the resistive force due to the slope of the road, γ. ρ, A, and C_D are air mass density, vehicle frontal area, and drag coefficient, respectively. Numerical values for all the parameters used in this study are presented in Table 1.

Table 1   Parameters used in the simulations

Parameter	Value	Parameter	Value
m_v	1,600 kg	$P_{gen_{max}}$	50 kW
α	4.16e-5 g/s/W	Q	23.18×10^3 C
β	0.007 g	R	0.3 Ω
f_{rr}	0.01	V_{oc}	212.6 V
ρ	1.15 kg/m^3	SoC_{max}	0.7
A	2.31 m^2	SoC_{min}	0.5
C_d	0.32	SoC_{ref}	0.6
$P_{b_{max}}$	27 kW	η_m	0.96
$P_{b_{min}}$	–40 kW	g	9.8 m/s^2

To model the hybrid powertrain, quasi-static models of each component are used (Serrao and Rizzoni, 2008). The following sub-sections present the model of each component.

2.1 Nickel metal hydride battery

As a simple realisation for control purposes, a circuit model can be used for the battery modelling (Figure 2). In HEV applications, the battery works in a narrow window of state of charge (SoC), typically between 50% to 70%. Therefore, the change in the battery voltage (V_{oc}) is negligible, and V_{oc} can be considered constant. Figure 3 shows the simulation results for an accurate chemistry-based NiMH battery model (Dao et al., 2010), which justifies the assumption of constant V_{oc} for this simple model.

Figure 2 Battery circuit model
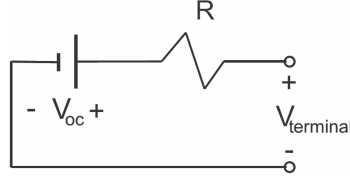
Figure 3 Simulation results for the open circuit voltage versus the SoC in a NiMH battery
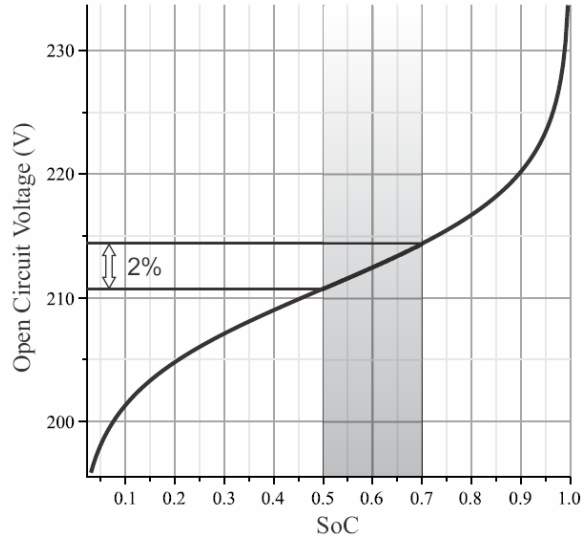
Source: Dao and McPhee (2011)

Considering Q as the battery capacity and i as the current passing through it, the derivative of SoC can be written as:

$$\dot{SoC} = -\frac{i}{Q} \tag{6}$$

The negative sign in (6) states that a positive current discharges the battery, and a negative current charges it. For the simple model of Figure 2 the battery terminal power, P_b, is found using (7).

$$P_b = iV_{oc} - Ri^2 \tag{7}$$

In the above relation, V_{oc} is the battery open circuit voltage, and R is the total of internal and terminal resistances of the battery.

By substituting i from (7) into (6), the time derivative of the SoC becomes:

$$\dot{SoC} = \frac{-V_{oc} + \sqrt{V_{oc}^2 - 4RP_b}}{2RQ} \tag{8}$$

In this HEV model, the only state is the battery SoC. The control parameter is chosen to be the battery power, thus:

$$x \triangleq SoC \tag{9a}$$

$$u \triangleq P_b \tag{9b}$$

$$\dot{x} = \frac{-V_{oc} + \sqrt{V_{oc}^2 - 4Ru}}{2RQ} \tag{9c}$$

2.2   Engine-generator

One of the major advantages of the series HEV architecture is that the engine is not mechanically connected to the driveline. Instead, it is coupled to a generator, allowing the engine speed to be chosen arbitrarily so that the engine works in the minimum brake specific fuel consumption (BSFC) point for every output power. If this minimum fuel consumption rate is plotted versus the generator output power, the outcome is a linear relation. Figure 4 shows the simulation results conducted on a mean-value engine model (Saeedi, 2010) coupled to a permanent magnet DC generator. In such conditions, the fuel consumption rate can be approximated as

$$\dot{m} = \alpha P_{gen} + \beta \tag{10}$$

with α and β being constants.

Figure 4   Simulation results for the minimal fuel consumption rate versus generator power
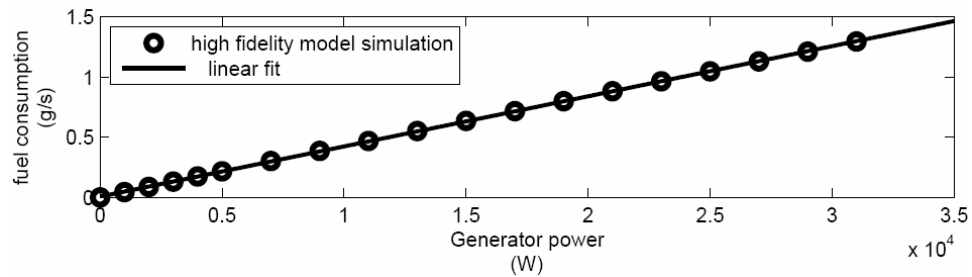
2.3   Electric motor

One or more electric machines are responsible to drive the wheels. These machines can be modelled as power transducers that convert the electrical power to mechanical power and vice versa. Losses in the driveline and the motors can be modelled with a single efficiency using (11). It is also assumed that only a fraction of the kinetic energy (50% in this study) is restored during regenerative braking. It should be noted that this assumption does not have any effect on the general behaviour of the vehicle.

$$\begin{cases} P_e = P_d \eta_m^{-1} & P_d > 0 \\ P_e = \frac{1}{2}(P_d \eta_m) & P_d < 0 \end{cases} \tag{11}$$

In this equation, η_m is the total efficiency of the driveline from the electric motor to the wheels. Comparisons made between this simple model and a high-fidelity HEV model showed that a constant value of η can be accurate enough for control purposes.

2.4 Electrical bus

The electric bus consists of the power electronic drivers for the motor and the generator. Neglecting losses, the balance of the energy in the electric bus can be written as:

$$P_{gen} + P_b = P_e \tag{12}$$

In this relation, the positive values indicate that the power is flowing from the powertrain toward the wheels, and the negative sign shows that the power is reversed. It is obvious that the generator power cannot be negative.

Although this model is very simple, with only one state, it will be shown that this model is capable of providing enough accuracy for the purpose of the model-based control.

3 Optimal control problem

The goal of the supervisory controller in this study is to minimise fuel consumption. Therefore, a cost function of the form (13) is considered.

$$J = \int_0^{t_f} \dot{m}dt \tag{13}$$

By combining (7), (9), (10), (11), and (12) with (13), the cost function can be written as:

$$J = \int_0^{t_f} \left[\alpha \left(P_e - u \right) + \beta \right] dt \tag{14}$$

The physical constraints of this powertrain are

$$x_{\min} < x < x_{\max} \tag{15a}$$

$$P_{b_{\min}} < P_b < P_{b_{\max}} \tag{15b}$$

$$0 < P_{gen} < P_{gen_{\max}} \tag{15c}$$

It is possible to write (15b) and (15c) as a single constraint on the control by combining them with (12) and (7). Therefore, the constraints on the control would be

$$u_{\min} = \max \left\{ P_{b_{\min}}, P_e - P_{gen_{\max}} \right\} \tag{16a}$$

$$u_{\max} = \min \left\{ P_{b_{\max}}, P_e, \frac{V_{oc}^2}{4R} \right\} \tag{16b}$$

$$u_{\min} < u < u_{\max} \tag{16c}$$

The optimal control problem can now be defined as follows: *Find the optimal control, u, such that the cost function* (14) *is minimised while the constraints* (17) *are satisfied.*

$$\dot{x} = \frac{-V_{oc} + \sqrt{V_{oc}^2 - 4Ru}}{2RQ} \tag{17a}$$

$$x(0) = x(t_f) = x_{ref} \text{ (charge sustaining condition)} \tag{17b}$$

$$x \in \mathcal{X}, \quad \mathcal{X} = [x_{min}, x_{max}] \tag{17c}$$

$$u \in \mathcal{U}, \quad \mathcal{U} = [u_{min}, u_{max}] \tag{17d}$$

This is a deterministic optimisation problem, since it is assumed that the input (electrical power demand P_e) is known for all $t \in [0, t_f]$. To solve this problem, the Hamiltonian is defined according to (18).

$$H = [\alpha(P_e - u) + \beta] + \lambda \left(\frac{-V_{oc} + \sqrt{V_{oc}^2 - 4Ru}}{2RQ} \right) \tag{18}$$

In this definition, λ is the Lagrange multiplier or the *costate*, with its dynamics defined by (19).

$$\dot{\lambda} = -\frac{\partial H}{\partial x} \tag{19}$$

Since none of the battery parameters (V_{oc}, R, and Q) is assumed to be a function of the state, the costate derivative is zero, and the costate holds its initial value to the end of the mission.

$$\dot{\lambda} = 0 \tag{20}$$

PMP converts the integral minimisation of (14) to an instantaneous minimisation of the Hamiltonian, and states that the optimal control is the one that satisfies (21).

$$u^* = \arg\min_{u \in \mathcal{U}} \{H\} \tag{21}$$

The solution to this problem is the solution of a two-point-boundary-value (TPBV) problem, as the initial and final state values are specified but the initial (and constant) costate value is unknown. The shooting method is a relatively fast and simple method for this TPBV problem, in which the unknown initial conditions are *guessed* and the differential equations are integrated to the final time. If the final values found are close enough to the specified values, the solution has been reached. Otherwise, the initial guess is changed and the whole process is repeated until the final criterion is met within the desired tolerance.

For this method, a discrete-time version of dynamic equations is considered as in (22).

$$x[k+1] = x[k] + \frac{-V_{oc} + \sqrt{V_{oc}^2 - 4Ru[k]}}{2RQ} \tag{22}$$

The costate, λ, is the value to be guessed in this problem. At each step of integration, and with known values of $x[k]$ and λ, the control range $[u_{min}, u_{max}]$ is identified. In this range, the value of u that minimises the Hamiltonian is chosen as the optimal control value, $u^*[k]$. Then with this value, (22) is integrated (forward Euler integrator) to the next step. This solution continues to the final time, when $x(t_f)$ is found. If $x(t_f) \neq x_{ref}$, the guess for the costate is modified and the whole process is repeated until $x(t_f)$ is close enough to x_{ref}.

Solving for the correct value of the costate is therefore a lower level optimisation problem: find the costate, λ, in such a way that $(x(t_f) - x_{ref})^2$ is minimised. This problem can easily be solved using available software packages such as the optimisation toolbox in Matlab.

Since at each time the control value is chosen from the range $[u_{min}, u_{max}]$, the control constraint is essentially satisfied. However, considering the state inequality constraint (17c) is a more challenging process. This constraint can be written as the inequality $G(x, t) < 0$, with G defined as:

$$G = (x - x_{min})(x - x_{max})$$
(23)

Whenever this constraint is active (i.e., $G \geq 0$), the necessary conditions in the PMP must be slightly altered. In such a case, the optimal control value is identified as:

$$u^* = \arg\min_{u \in \mathcal{U}} \{H + \mu \dot{G}\}, \quad \mathcal{U} = \left\{ [u_{min}, u_{max}] \middle| \dot{G} = 0 \right\}$$
(24)

where μ is a positive number, and $\dot{G}$ is the time derivative of G:

$$\dot{G} = \dot{x}(x - x_{max}) + \dot{x}(x - x_{min}) = \dot{x}(2x - x_{min} - x_{max})$$
(25)

$$\dot{G} = 0 \Rightarrow \dot{x} = 0 \Rightarrow u = 0 \Rightarrow \mathcal{U} = \{0\}$$
(26)

Therefore, the optimal control, u^*, is zero in active constraint regions. The details and proof of this method are available in Geering (2007, Section 2.5).

This method can be explained heuristically; whenever the SoC (x) reaches its boundaries, the battery cannot be charged (or discharged), and the admissible control is zero to prevent it from being over-charged (over-discharged).

4 Optimal feedback control

At every time step, the value of the control in $[u_{min}, u_{max}]$ that minimises the Hamiltonian is chosen as the optimal value. Since at each time step the state, the costate, and the power $[x, \lambda$ and P_e in (18)] have certain values, the Hamiltonian takes a convex form in terms of the control, u.

$$\frac{\partial H}{\partial u} = -\alpha + \frac{\lambda}{2RQ}\left[-\frac{2R}{\sqrt{V_{oc}^2 - 4Ru}} \right]$$
(27)

$$\frac{\partial^2 H}{\partial u^2} = -\frac{2\lambda R}{Q}\frac{1}{(V_{oc}^2 - 4Ru)^{\frac{3}{2}}} > 0$$
(28)

$$\frac{\partial H}{\partial u} = 0 \Rightarrow \bar{u} \triangleq \frac{1}{4R}\left(V_{oc}^2 - \frac{\lambda^2}{\alpha^2 Q^2} \right) \tag{29}$$

In (29), it can be seen that the Hamiltonian has only one extremum at $\bar{u}$. For stable shooting method solutions, the costate has to be negative, thus, the second derivative in (28) will be positive, resulting in a convex function. It is worth noting that by using the shooting method, the global optimality of the solution is guaranteed. That is because there is only one solution candidate, and if the choice of the costate satisfies the final boundary conditions, the solution is unique, thus globally optimal.

Figure 5   Hamiltonian versus control, u

(a)   (b)   (c)
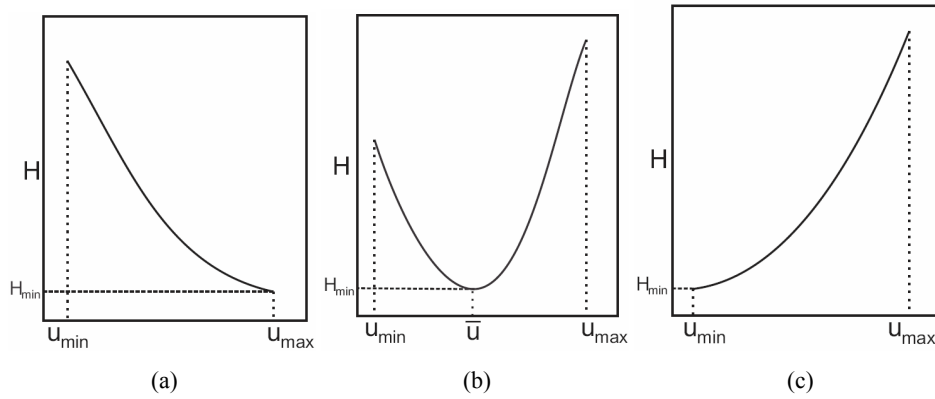
The quadratic form of the Hamiltonian also implies that the minimum of H happens either at a boundary value of u (namely u_{min} or u_{max}), or when $\dfrac{\partial H}{\partial u}$ is zero, (29). These conditions are shown in Figure 5. According to (30), $\bar{u}$ has a constant value throughout a mission.

$$\frac{d\bar{u}}{dt} = -\frac{1}{2R\alpha^2 Q^2}(2\lambda\dot{\lambda}) = 0 \tag{30}$$

In fact, $\bar{u}$ is the governing parameter in this problem, and can be found by (29), using only battery parameters and the costate value. Therefore, a simple yet optimal feedback controller can be defined as:

$$u^* = \begin{cases} u_{max} & u_{max} < \bar{u} \\ \bar{u} & u_{min} < \bar{u} < u_{max} \\ u_{min} & \bar{u} < u_{min} \end{cases} \tag{31}$$

with u_{min}, u_{max} and $\bar{u}$ defined in (16a), (16b) and (29), respectively. The inputs to this controller are the state (as feedback) and P_e, and the controller determines the optimal battery power.

The mechanism of the optimal control can be simply explained. When the electric power demand is negative (during braking) and the SoC is within the admissible range,

the battery absorbs all the available power. When the power demand is positive, but less than a certain value $(\bar{u})$, the battery provides all the required power. These two conditions correspond to the first case in (31), and are shown in Figure 6(a). When the power demand is more than $\bar{u}$, only a portion of the required power is delivered by the battery, and the rest is provided by the generator. This condition corresponds to the second case of (31), and is shown in Figure 6(b). Finally, if the power demand is too high (when $u_{\min} > \bar{u}$ or equivalently $P_e > P_{gen,\max} + \bar{u}$), the battery provides more power than $\bar{u}$ to drive the vehicle. This corresponds to the last case in (31), and is shown in Figure 6(c). When the SoC is less than its minimum allowable value, the battery will not provide any power, and when SoC is more than its maximum allowable value, it will not absorb electrical power.

Figure 6 The mechanism of the optimal supervisory controller
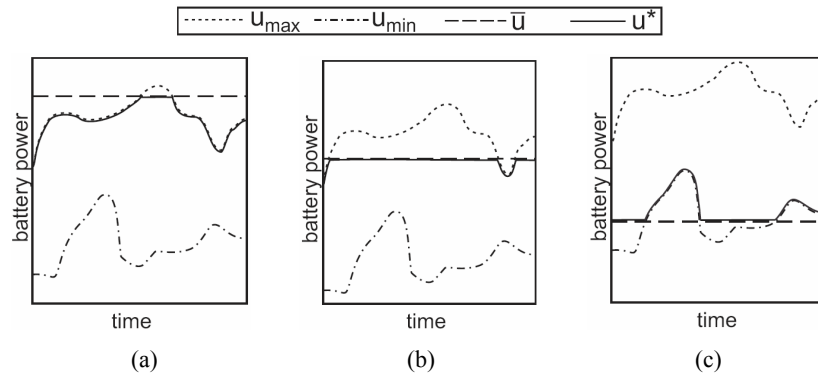
As was mentioned earlier, the most important parameter in this controller is the costate (which in turn determines $\bar{u}$). In the next section, a simple method is presented to find the proper value of the costate and $\bar{u}$.

5 Costate estimation method

For optimal behaviour, it is necessary to have the future driving condition. Without such information, only sub-optimal behaviour is achievable (Sciarretta and Guzzella, 2007). In this study, it was observed that it is not necessary to consider the whole drive cycle. Instead, if only the driving condition until the next stop [*stop-to-stop* (STS) *cycle*] is known, it is possible to obtain a solution that is almost as optimal as the solution found considering the whole drive cycle. An example is presented in Figure 7, with the FTP75 drive cycle shown in the top plot, and the SoC shown in the bottom plot for two different control strategies. To obtain these results, the PMP was solved once for the full drive cycle, and once for successive STS cycles when the final state was required to be x_{ref} at the end of each STS cycle. A comparison of the resultant fuel consumption for various drive cycles is presented in Table 2, which shows negligible difference in fuel consumption between full drive cycle and STS cycle optimisation.

Figure 7   State trajectory when considering the full drive cycle, and when considering STS cycles
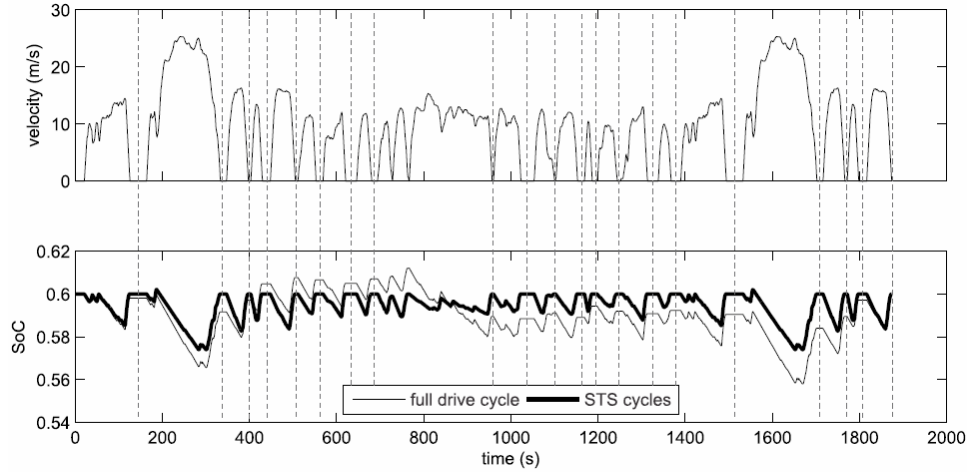
Table 2   Fuel consumption for the two methods

	FTP75	HUDDS	EUDC	NYCC	LA92
Full drive cycle (gr/cycle)	274.8	154.4	257.5	31.69	308.5
STS cycles (gr/cycle)	275.1	154.6	257.6	31.89	308.8
Increase in fuel consumption	0.10%	0.12%	0.04%	0.63%	0.10%

The costate value is the only parameter that should be tuned for these STS cycles. Since the optimal control mechanism is independent of the driving condition, it is only the charge sustenance that should be considered in tuning the costate.

The mechanism mentioned earlier follows one important concept: it tries to capture as much negative energy as possible, thus elevating the SoC. To discharge the battery to its initial charge level, the controller decides that a *certain amount* of power has to be provided by the battery during acceleration and cruising. This certain amount is $\bar{u}$ found in (29).

It should be noted that in both solutions shown in Figure 7, the mechanism of the optimal controller is the same, as is the amount of regenerative braking absorbed by the battery. The only cause for the change in fuel consumption is the small change in battery losses. When the whole drive cycle is considered, the optimal control adjusts $\bar{u}$ so that the battery losses are minimised over the whole mission. But when the STS cycles are considered separately, the controller changes $\bar{u}$ for each cycle, thereby increasing the charge/discharge rate of the battery, the battery loss, and finally, the total fuel consumption. It was observed that by changing battery parameters (so that the battery efficiency is decreased), the difference in fuel consumption between the two methods becomes more noticeable. However, with reasonable battery parameters, the difference is small, as in Table 2.

The objective is now to estimate the costate (or equivalently $\bar{u}$) for each of the STS cycles to satisfy charge sustenance. A typical plot of electric power demand, P_e, and the corresponding battery power, u, is shown in Figure 8. To have the SoC at the same level

at the beginning and end of the cycle, the total change in the battery energy (the integral of the shaded area) must be zero, i.e.:

$$E_n + E_p = 0 \tag{32}$$

where E_n represents the total negative energy absorbed by the battery, and E_p is the total energy delivered to the powertrain by the battery.

Figure 8 A typical plot of electric power
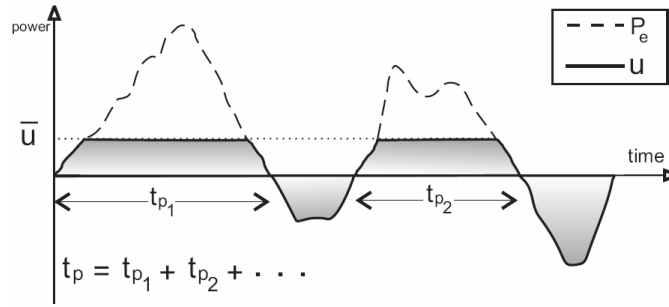
As was observed in simulations of different standard drive cycles, $\bar{u}$ has a relatively small value (usually less than 5 kW). In most of the times when the power demand is positive, it is greater than $\bar{u}$; thus the optimal battery power is equal to $\bar{u}$ in a great portion of the drive time. Therefore, it is reasonable to assume the battery power is equal to $\bar{u}$, which leads to

$$E_p = \int_{t_p} u\,dt \simeq t_p \bar{u} \tag{33}$$

In (33), t_p is the time when power demand is positive. Combining (33) and (32) gives

$$\bar{u} = -\frac{E_n}{t_p} \tag{34}$$

If the relations (29) and (34) are considered together, the value of the costate that satisfies charge sustenance requirements, λ^*, can be found as in (35).

$$-4R\bar{u} + V_{oc}^2 = \frac{\lambda^{*2}}{a^2 Q^2} \Rightarrow \lambda^* = -aQ\sqrt{V_{oc}^2 + 4R\left(\frac{E_n}{t_p}\right)} \tag{35}$$

Relations (34) and (35) relate the optimal values of λ and $\bar{u}$ to only two parameters of the drive cycle: the total negative energy available and the time when positive power is required. During the simulations, it was observed that the optimal value of $\bar{u}$ is independent of the order of events. For example, it is not necessary to know when the driver is going to push the brake pedal; it is only important to know how much kinetic energy is going to be transferred to electrical energy before the next stop. This behaviour can be justified by (34), which is only a function of total energy and time.

In equation (32), it is assumed that the final SoC should come back to its *initial* level. If (due to any kind of error) the initial SoC has a value different from the desired SoC_{ref},

then the feedback controller tries to bring it back to the initial value, not SoC_{ref}. To compensate for this error, the difference in the battery energy should be considered as in

$$E_n + E_p = E_{res} \tag{36}$$

with E_{res} being the amount of energy that the battery should absorb or release. E_{res} can be estimated using (39) by assuming constant V_{oc}.

$$E_{res} = Q \int_{SoC_0}^{SoC_{ref}} \left[dSoC \, V_{oc}(SoC) \right] \tag{37}$$

$$E_{res} \simeq QV_{oc} \int_{SoC_0}^{SOC_{ref}} dSoC \tag{38}$$

$$E_{res} \simeq QV_{oc} \Delta SoC \tag{39}$$

Therefore, $\bar{u}$ and λ can more robustly be approximated using:

$$\bar{u} = -\frac{E_n - E_{res}}{t_p} \tag{40}$$

$$\lambda^* = -aQ \sqrt{V_{oc}^2 - \frac{4R}{t_p} \left(E_n + Q\left(x_0 - x_{ref} \right) V_{oc} \right)} \tag{41}$$

As mentioned in Section 1, the ECMS can be optimised using the PMP approach. In this case, the equivalence factor is tightly related to the costate. Therefore, it is possible to find the optimal value of the equivalence factor at each instant using the costate found by the method presented in this paper.

Although the costate estimation method still requires certain information about future driving conditions, it is a less demanding problem than finding the exact speed profile. It is possible to estimate cruise times using ITS and GPS systems; the available negative energy is related to vehicle kinetic energy during braking, which can also be estimated using the longitudinal vehicle dynamics (Barth et al., 2003).

6   Comments regarding use of ultra-capacitor in HEVs

An ultra-capacitor (UC) can be a good option for an HEV electrical energy storage system, as it has higher efficiency and power-density, and much longer lifetime compared to NiMH and Li-ion batteries. In contrast, an UC's relatively low energy-density, and internal energy dissipation make it inappropriate for EV and PHEV applications (Burke, 2007).

UCs can be modelled using RC circuits [Figure 9(a)], and the number of the RC branches determines the accuracy of the model (Shi and Crow, 2008). Since increasing the number of the branches increases the number of the states of the system, a simple RC circuit is suitable for the control-oriented model [Figure 9(b)]. This model can still capture enough details about the UC behaviour. For such a model, the relationship between the capacitance voltage, V_C, and the current, i, can be written as

$$\dot{V}_C = -\frac{i}{C} \tag{42}$$

with C being the capacitance. Considering the capacitance voltage as the state of the system and the current as the control, the state equation is found to be according to (43).
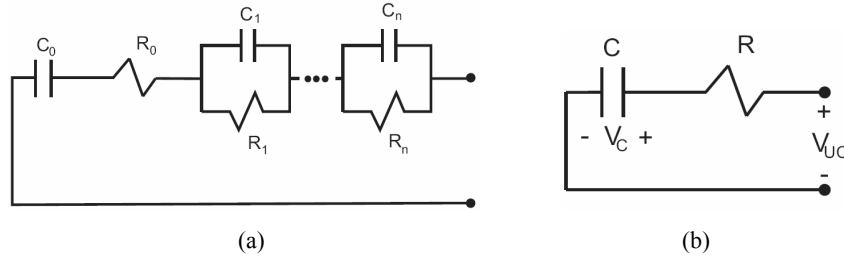
$$\dot{x} = -\frac{u}{C} \tag{43}$$

Equation (44) defines the UC output power, P_{UC}, in terms of the state and the control.

$$P_{UC} = ux - Ru^2 \tag{44}$$

In the above relation, R is the UC equivalent resistance.

Figure 9 UC circuit model

(a)   (b)

Based on this modified HEV model, the new cost function and Hamiltonian can be written as in (45) and (46), respectively (Razavian et al., 2012).

$$J = \int_0^{t_f} \left[\alpha \left(P_e - xu + Ru^2 \right) + \beta \right] dt \tag{45}$$

$$H = \left[\alpha \left(P_e - xu + Ru^2 \right) + \beta \right] + \lambda \left(-\frac{u}{C} \right) \tag{46}$$

The costate is no longer constant, and is governed by the dynamic equation:

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = \alpha u \tag{47}$$

When using an UC, similar to the case that a NiMH battery was used, the Hamiltonian takes a convex form (in the UC case, it is quadratic). Therefore, the arguments regarding uniqueness and global optimality of the solution are still valid. Moreover, $\bar{u}$ (the control value at which the Hamiltonian has a zero derivative) is a constant value:

$$\frac{d\bar{u}}{dt} = \frac{1}{2R}\frac{dx}{dt} + \frac{1}{2R\alpha C}\frac{d\lambda}{dt} \tag{48}$$

$$= \frac{1}{2R}\left(-\frac{u}{C} \right) + \frac{1}{2R\alpha C}(\alpha u) = 0 \tag{49}$$

Therefore, the optimal supervisory controller has the same logic, except the fact that the governing parameter, $\bar{u}$, is the UC *current*, not the power. The optimal UC current is always the maximum possible current, u_{max}, except when it is higher than $\bar{u}$, in which case the optimal UC current is $\bar{u}$. Finally, the same method can be used to estimate the costate initial value and $\bar{u}$, based on the same information (i.e., cruise time and regenerative energy).

7   High-fidelity model

The controller and the methods presented in the previous sections were based upon a very simple HEV model. Because of extensive simplification, this control-oriented model may fail to accurately represent the actual behaviour of the vehicle. To evaluate the performance of the designed controller, the first step is model-in-the-loop (MIL) simulation. In MIL simulations, a high fidelity model is used to emulate vehicular behaviour.

In this work, the series HEV is modelled in the MapleSim environment (Dao et al., 2010). MapleSim allows a causal modelling, and does symbolic calculations to reduce run times. The MapleSim model is presented in Figure 10. As can be seen, the model consists of different components, including the engine, the generator, the battery, the electric motor, and a multi-body vehicle.

Figure 10 High-fidelity model for a series HEV in the MapleSim environment (see online version for colours)
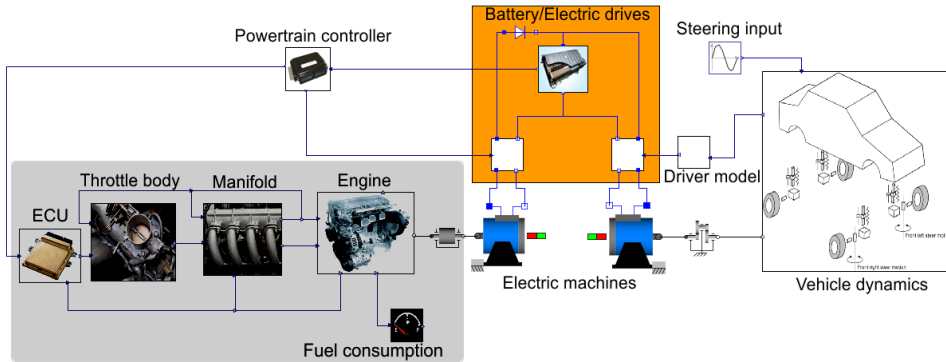
7.1   Engine-generator set model

This model uses a mean-value engine model (Saeedi, 2010), which is accurate enough for this application and also is simple enough to allow fast simulations. The engine model, as can be seen in Figure 10, consists of four components: the ECU, throttle, manifold, and engine body. The ECU block is responsible for adjusting the throttle angle to make the engine deliver the desired amount of torque. In the throttle and manifold blocks, the pressure and mass flow rate of the air/fuel mixture are calculated:

$$\dot{P}_m = \frac{R_g T_m}{V_m}\left(\dot{m}_{thr} - \dot{m}_e\right) \tag{50}$$

where P_m is the pressure of the air/fuel mixture, T_m and V_m are the temperature and volume of the intake manifold, respectively, R_g is the gas constant, and $\dot{m}_{thr}$ and $\dot{m}_e$ are throttle mass flow rate and manifold out flow respectively.

Having the mass flow rate and pressure calculated, the engine body block calculates the thermal efficiency and indicated power. Finally, the engine output power is found according to:

$$P_{net} = P_{ind} - P_{loss} - P_{load} \tag{51}$$

with P_{ind}, P_{loss}, and P_{load} being the indicated power, lost power, and internal load power, respectively.

The engine flywheel is coupled to a permanent magnet DC generator that produces electricity to charge the battery. The effects of power electronics in the generator electric drive are neglected in this model. It is assumed that the electric drive is a DC-DC converter with an efficiency of 100%. The schematic of the power converter is shown in Figure 11. In this system, the power drawn from (or delivered to) terminal 2 is equal to the amount of power at terminal 1. The PID controller ensures that this power is independent of variations in terminal voltage due to changes in generator conditions.

Figure 11 Schematic of the DC/DC converter
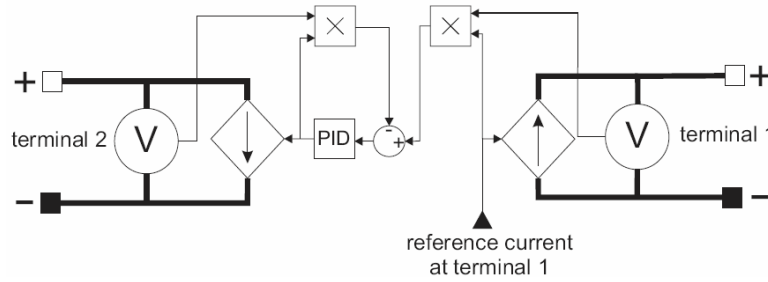
7.2 Battery model

One of the most important components in a hybrid powertrain is the electrical storage. In the model developed here, a chemistry-based NiMH battery (Dao and McPhee, 2011) is used to further enhance the accuracy of the simulations. In this model, cell terminal voltage is calculated using:

$$v_{cell} = (\phi_c + \eta_c) - (\phi_a + \eta_a) + i_{cell} R_{int} \tag{52}$$

where v_{cell} is the terminal voltage, and ϕ_i and η_i are the electromotive force and over-voltage, respectively, which are found based on chemical reactions and ion concentrations in the anode (a) and cathode (c). R_{int} is the ohmic internal resistance, and i_{cell} is the cell current found using:

$$i_{cell} = A_i a_i l_i j_i \tag{53}$$

where A_i, a_i and l_i are the geometry parameters. j_i is defined in (54), with F and T being the Faraday constant and temperature, respectively. $i_{0,i}$ is also calculated based on the ion concentration at electrodes.

$$j_i = i_{0,i} \left(e^{\frac{0.5F}{RT}\eta_i} - e^{\frac{-0.5F}{RT}\eta_i} \right) \tag{54}$$

In (53) and (54), subscript *i* should be replaced with *c* for cathode and *a* for anode.

7.3   Vehicle dynamics

For the electric motor, which is responsible for driving the wheels, a permanent magnet DC machine is used. Similar to the generator, the effects of power electronics in the motor electric drive are neglected. The output shaft of this motor is connected to the wheels via a constant gear ratio.

The vehicle dynamics is simulated by a 14 degree-of-freedom (DOF) multi-body model with Fiala tyres and aerodynamic drag force. The degrees of freedom include six DOFs for the body, four DOFs for vertical displacement of the sprung mass, and four DOFs for rotation of the wheels. This model is sufficient to predict the handling and braking behaviors of a vehicle. The aerodynamic drag force is simulated using an external load acting on the vehicle's centre of mass, and is calculated using (4) and the vehicle's longitudinal velocity.

To couple the designed supervisory controller to this high-fidelity model, different components of the model are converted to the Matlab/Simulink environment as *S*-functions, and are connected together by Simulink signal links. The converted model in Simulink is no longer an acausal model, since different powertrain components are connected together by causal Simulink signal links. The next section presents the overview of the control loop and the way the supervisory controller interacts with other components.

8   Low-level controllers

The schematic of the converted model can be seen in Figure 12. The vehicle dynamics and the electric motor components from MapleSim are converted together as the new vehicle dynamics block in Figure 12. The driver model is a simple PID controller that adjusts the motor current to ensure that the vehicle follows the desired speed profile. The outputs of the vehicle dynamics block are the electric power required to follow the speed profile, P_e, and the vehicle speed, which is monitored and used as a feedback for the PID controller. This electric power (consumed by the electric motor or generated during braking) is used as an input to the supervisory controller. It is also used as a signal to determine the charge/discharge rate of the battery.

8.1   Engine-generator set controllers

The supervisory controller uses the future traffic information as an input to calculate the optimal battery power, and in this study, it is assumed that such information is available in advance for the current STS cycle. Based on the traffic information, instantaneous electric power demand, P_e, and SoC, the supervisory controller can determine the optimal battery power using the logic presented in previous sections. Then the controller uses (55) to calculate the amount of power that the engine-generator set should produce. As long as the low-level controllers for the engine-generator set provide appropriate tracking of the set points, tracking of the optimal battery power is guaranteed:

$$P_{gen}^* = P_e - P_b^* \tag{55}$$

Based on the reference generator power calculated by the supervisory controller, a pair consisting of the torque and speed is identified to give the maximum efficiency of the engine-generator set in that output power. The curves shown in Figure 13 are found by simulating the mean-value engine model, and are used to find the optimal speed and torque based on the generator power. Because of the engine's non-linear behaviour, a sliding mode controller is used to control the engine torque by adjusting the throttle angle.

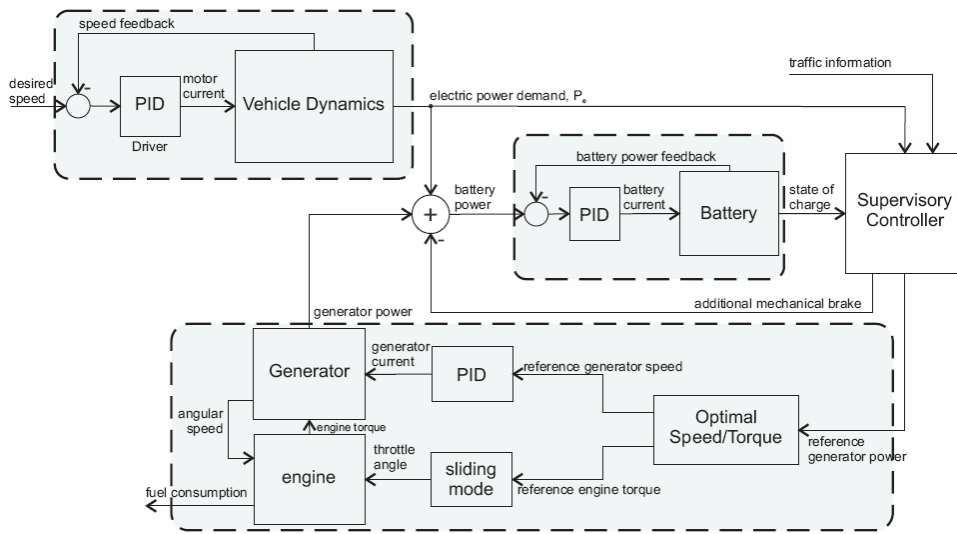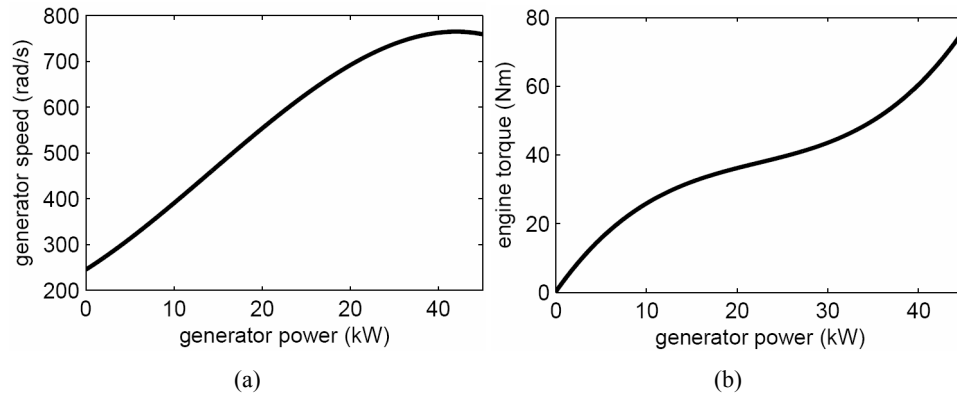**Figure 12** Schematic of the causal HEV model used in the control loop

Figure 13 Optimal engine torque and generator speed versus generator output power

(a)   (b)

Sliding mode control has proven to be a reliable method for engine torque management in practical cases. This is a model-based method, and is capable of handling the model uncertainties. The main control input for the engine is throttle angle. Other inputs, like air fuel ratio and ignition timing, highly affect the transient behaviour of the engine.

According to legislation to have the best fuel economy and emission for the engine, we are not allowed to change these two parameters away from their optimum values for a long time (Lamberson, 2003). Therefore, throttle angle is generally more reliable and dominating engine input to change the steady state response in this case.

Engine torque management can be done in two ways. The first approach is using an engine torque sensor to measure crankshaft torque for feedback control. This measurement includes the effects of the engine combustion torque, friction torque, pumping torque, and all accessory loads. In this derivation, spark timing and air fuel ratio are used to control the engine torque. Using these two variables as the inputs, forces them away from the optimal ranges, as mentioned above. Thus, torque control is not a legitimate option for the torque management strategy, but it would reduce uncertainties especially in case of engine aging.

The second approach is to measure and control manifold pressure. Engine combustion torque is a function of cylinder air flow which is a function of the manifold pressure. Assuming constant air to fuel ratio and ignition timing, the control goal can be changed to make the manifold pressure follow the desired value. If the throttle is used to control manifold pressure, the effect of ignition timing and air to fuel ratio on combustion torque does affect the throttle control. Thus, the disadvantage of this approach is a larger amount of calibration required to get a proper conversion from desired torque to desired manifold pressure for all engine operating conditions. But, use of this strategy will not require a torque sensor (Lamberson, 2003).

Here we use the manifold pressure control approach and ignore the effect of transient torque change for the engine set points so we keep the spark timing and air fuel ratio unchanged throughout the simulation.

To accomplish this goal, we need a simple model of the engine to be used for designing the sliding mode controller. A simplified model of the engine is a single input single output. The manifold pressure, P_m, is the only state variable and engine indicated torque is the output:

$$\dot{P}_m = \underbrace{-\frac{\eta_v N_{cyl} V_d \omega}{60 N_{eng} V_m}}_{A_{eng}} P_m + \underbrace{\frac{R_g T_m}{V_m}\left(C_D \times MA \times PRI\right)}_{B_{eng}} A_{th} \tag{56a}$$

$$T_{ind} = \underbrace{-\frac{\phi \eta_i \eta_v N_{cyl} V_d H_f}{60 N_{eng} R_g T_m}}_{C_{eng}} P_m \tag{56b}$$

where A_{th} is the throttle area, the control input. Throttle area can be found from *th* throttle angle, θ, and geometry (*d* and *D* are diameter values for input and output vents and θ_0 is the angle when throttle is totally closed):

$$A_{th} = -\frac{d.D}{2}\sqrt{1-\left(\frac{d}{D}\right)^2} + \frac{d.D}{2}\sqrt{1-\left(\frac{d\cos\theta_0}{D\cos\theta}\right)^2}$$
$$+ \frac{D^2}{2}\sin^{-1}\left(\sqrt{1-\left(\frac{d}{D}\right)^2}\right) - \frac{D^2}{2}\cdot\frac{\cos\theta}{\cos\theta_0}\sin^{-1}\left(\sqrt{1-\left(\frac{d\cos\theta_0}{D\cos\theta}\right)^2}\right) \tag{57}$$

In (56), η_v is the volumetric efficiency which is a function of manifold pressure and engine speed, ω. N_{cyl} is the number of cylinders (four in this case). V_d and V_m are the engine displacement and air manifold volume respectively. N_{eng} is 2 for a four-stroke engine. R_g is the gas constant, and T_m is the manifold temperature (considered constant for simplicity). C_D is the throttle discharge coefficient.

Also, $MA = P_0 \big/ \sqrt{R_g T_0}$ where P_0 and T_0 are atmosphere pressure and temperature, respectively, and *PRI* is a non-dimensional value to consider subsonic and supersonic air flow (γ is air heat capacity ratio):

$$PRI = \begin{cases} \left(\dfrac{P_m}{P_0}\right)^{\frac{1}{\gamma}} \sqrt{\dfrac{2\gamma}{\gamma-1}\left(1-\left(\dfrac{P_m}{P_0}\right)^{\frac{\gamma-1}{\gamma}}\right)} & \text{for } \dfrac{P_m}{P_0} > \left(\dfrac{2}{\gamma+1}\right)^{\frac{\gamma-1}{\gamma}} \\ \sqrt{\gamma\left(\dfrac{2}{\gamma+1}\right)^{\frac{\gamma-1}{\gamma+1}}} & \text{for } \dfrac{P_m}{P_0} \le \left(\dfrac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \end{cases} \tag{58}$$

Lastly, η_i, ϕ and H_f are engine thermal efficiency (approximately a function of engine speed and manifold pressure), stoichiometry fuel-air ratio and gasoline heat of combustion, respectively. Readers are referred to Saeedi (2010) for numeric values of the parameters used in this engine model.

According to Slotine and Li (1991), we can define a sliding surface, *S*:

$$S = P_m - P_{m,desired} \Rightarrow \dot{S} = \dot{P}_m - \dot{P}_{m,desired} \tag{59}$$

where $P_{m,desired}$ is the desired manifold pressure which leads to the desired engine indicated torque. Therefore, $\dot{S}$ can be found as in (60)

$$\dot{S} = A_{eng} P_m + B_{eng} A_{th} - \frac{\dot{T}_{ind,desired}}{C_{eng}} \tag{60}$$

where A_{eng}, B_{eng}, and C_{eng} are defined in (56).

In order to satisfy reachability condition, we use the signum (*sgn*) function:

$$S\dot{S} < 0 \Rightarrow \dot{S} = -\eta sgn(S) \tag{61}$$

Now all we have to do is to tune η according to the model uncertainty and operating conditions. Then the control input will be:

$$A_{th} = \frac{1}{B_{eng}}\left(-\eta sgn\left(P_m - P_{m,desired}\right) - A_{eng} P_m + \frac{\dot{T}_{ind,desired}}{C_{eng}}\right) \tag{62}$$

On the generator side, a manually tuned PID controller ensures that the generator speed follows the reference value by adjusting the generator current. These two controllers provide power tracking, and at the same time, move the engine operating point to the minimum BSFC point.

8.2   Battery controller

The algebraic sum of the power generated by the engine-generator set and the power consumed (or generated) by the electric motor is used to charge/discharge the battery. A PID controller is responsible for making the battery follow the reference power by adjusting the battery current. The NiMH battery model calculates the SoC, and the SoC is sent back to the supervisory controller as a feedback.

During braking, the battery absorbs part of the kinetic energy, and the SoC increases. Although not likely with the model-based supervisory controller, the SoC may reach its upper limit during braking. In such cases, the supervisory controller turns off the regenerative braking, and to compensate for that, it increases the mechanical braking effort. In the model, the extra mechanical braking power is subtracted from the electric power demand to consider such loss of regenerative braking.

9   Results

In this study, all simulations are conducted in the Matlab environment. The optimal control problem is solved using the PMP for the FTP75 drive cycle, and control and state trajectories are shown in Figure 14. Note that the optimal control value is the same as u_{max} but is limited to a constant value $(\bar{u})$.

Figure 14   The optimal SoC and control trajectories for the FTP75 drive cycle found using the PMP solution
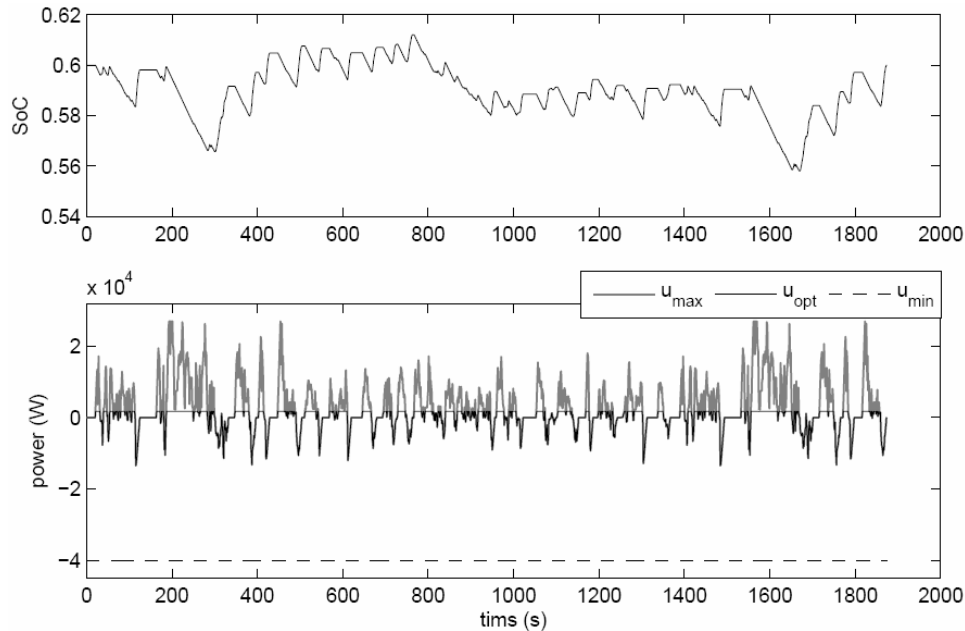

For 68 STS cycles listed in Table 3, the optimal $\bar{u}$ is found, and the correlation between this parameter and the drive cycle parameters, relation (34), is presented in Figure 15. It

is interesting to observe that the best linear approximation matches the relation (34) very well.

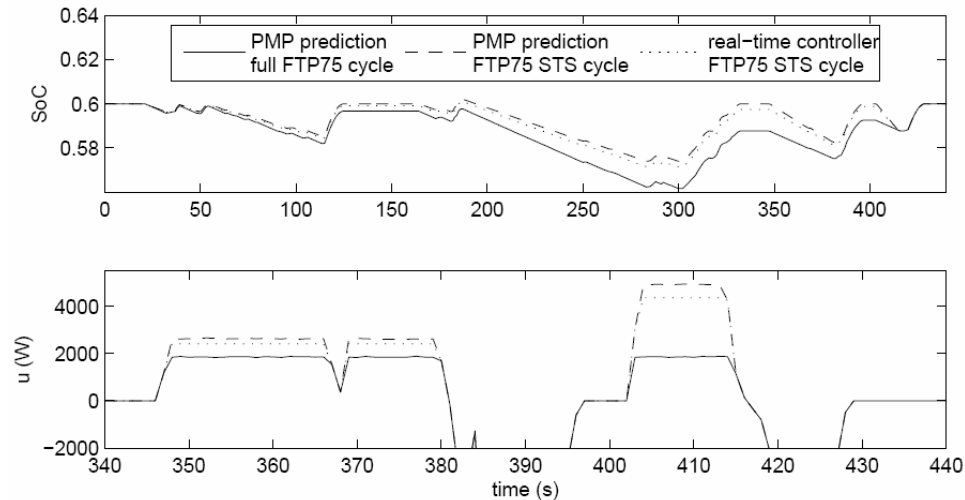**Table 3** List of the standard drive cycles used

Drive cycle	Number of STS cycles	Drive cycle	Number of STS cycles
UDDS	17	NYCC	10
US06	5	LA92	16
EUC	5	SC03	5
HUDDS	4	JN1015	3
HWFET	1	IM240	2

Figure 15 Linear fit for $\bar{u}$
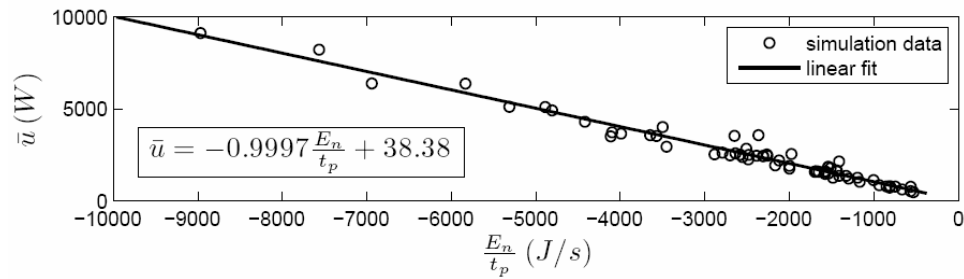
In Figure 16, the state and the control trajectories obtained by three different methods are shown for a portion of the FTP75 drive cycle. The methods are the PMP solution for the full drive cycle, the PMP solution for successive STS cycles, and the solution of the feedback controller with estimated $\bar{u}$. Although many simplifications are made to estimate $\bar{u}$, the results are very close to the solutions found using PMP.

Figure 16 SoC and control trajectories for the three different solutions for FTP75

The fuel consumption for different standard drive cycles is presented in Table 4. The first row of the table presents the optimal fuel consumption for the entire drive cycle, and the second row presents the fuel consumption resulted from the feedback controller with estimated $\bar{u}$ for successive STS cycles. Due to errors in $\bar{u}$ estimation, the final SoC is not exactly the same as the reference value (the reference value is 0.6). This variation in final SoC contributes to increase or decrease in total fuel consumption. To compensate for this error, the fuel consumption is corrected according to (63d).

$$\dot{m} = \alpha P_{gen} + \beta \tag{63a}$$

$$\int \dot{m}\,dt = \int \left(\alpha P_{gen} + \beta\right) dt \tag{63b}$$

$$\Delta m = \alpha E_{res} + \beta t \simeq \alpha E_{res} \text{ (assuming short time)} \tag{63c}$$

$$\Delta m = \alpha \left(QV_{oc}\Delta SoC\right) \tag{63d}$$

To evaluate the performance of the optimal controller, it is applied to the high fidelity series HEV model. The input to this simulation is the desired speed profile. The controller follows the logic in (31), and for the calculation of $\bar{u}$, the linear fit in Figure 15 is used. In the present work, it is assumed that the necessary information is available to the controller to estimate $\bar{u}$. The information includes the cruise time (t_p) and available negative energy (E_n), as is required in the calculation of $\bar{u}$ in Figure 15.

Table 4   Comparison of fuel consumption for different drive cycles

	FTP75	NYCC	EUC	HUDDS	HWFET
PMP solution for the full drive cycle (g)	274.27	31.520	257.243	154.150	263.272
Feedback controller for the STS cycles (g)	274.68	32.379	256.080	154.368	261.757
Final SoC	0.6005	0.6034	0.5937	0.6004	0.5925
Corrected fuel consumption (g)	274.577	31.681	257.374	154.286	263.297
Increase in fuel consumption with respect to optimal value	0.11%	0.5%	0.05%	0.09%	0.01%

To compare the behaviours of the control-oriented model and the high-fidelity model, simulations shown in Figure 17 are conducted. The input to both models is the desired speed profile, and output is the SoC. The comparison of the two SoC trajectories for the first 440s of the FTP75 drive cycle is shown in Figure 18. It can be seen that the controller can predict the actual vehicle behaviour very well based on the simple control-oriented model, in spite of the extensive number of simplifications made in its design process.

To see the optimality of the control strategy, the performance of this controller was compared with that of a PID controller. The idea behind this PID controller is to preserve the health of the battery as long as possible by minimising variation of the SoC. The PID controller keeps the SoC near the reference value by controlling the engine-generator power. The PID controller is tuned in such a way that the battery is charged in an appropriate time, and at the same time, is allowed to absorb all the regenerative braking without reaching the charge limit. Therefore, besides preserving the battery health, such a

PID controller is a good candidate for series HEV supervisory controllers in terms of optimality.

Figure 17 Simplified schematic of the simulations to validate controller-oriented model
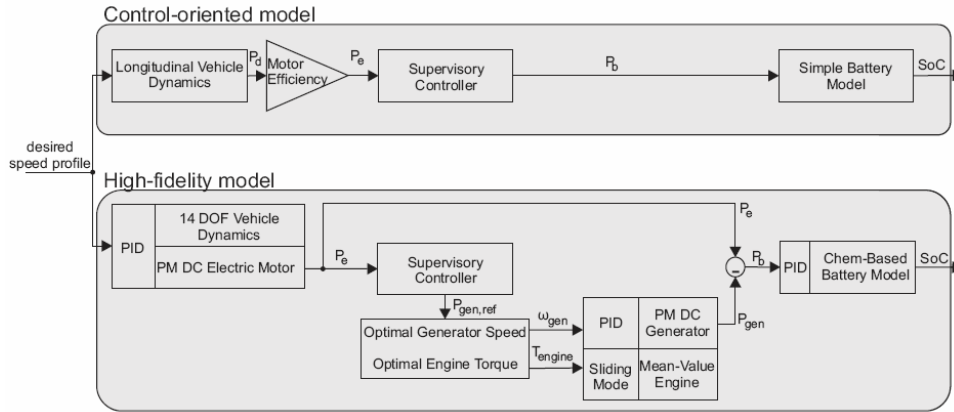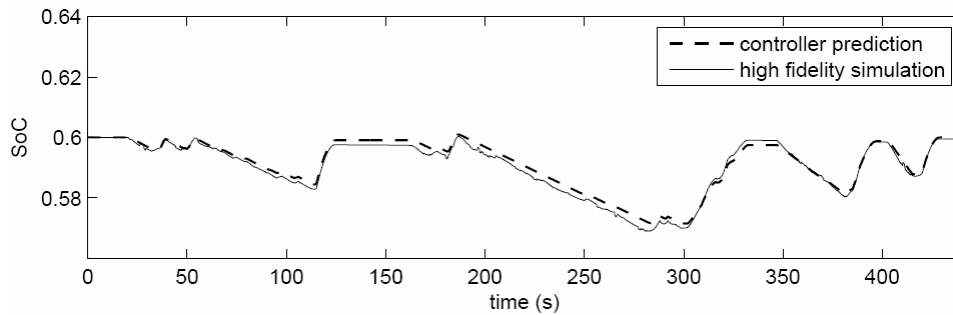

Figure 18 SoC trajectory for the first 440s of FTP75 drive cycle

As can be seen in Figure 19, the final SoC with the PID controller is higher than the reference value, because the battery captures the regenerative braking energy at the end of the drive cycle, when the SoC is close to the reference value. This increased SoC causes a noticeable increase in fuel consumption, and hinders comparison between this controller and the optimal controller (which is charge sustaining). To better compare the fuel consumptions, five successive FTP75 cycles were used to approximate infinite driving pattern. The results show that the optimal controller gives a fuel consumption of 1,504.2 grams, whereas the PID controller resulted in a total fuel consumption of 1,607.3 grams. Thus, fuel consumption is reduced by 6.4% with our model based controller.

Another way to compare the fuel consumptions is to change $\bar{u}$ so that the final SoC resulting from the optimal controller is equal to the final SoC resulting from PID controller. For the FTP75 drive cycle, this comparison shows that the optimal controller has a fuel consumption of 304 grams versus the 324 of the PID controller. In this case, the optimal controller shows a 6.5% reduction in fuel consumption compared to the PID controller.

Figure 19 Comparison of SoC trajectories for the FTP75 drive cycle for the two controllers
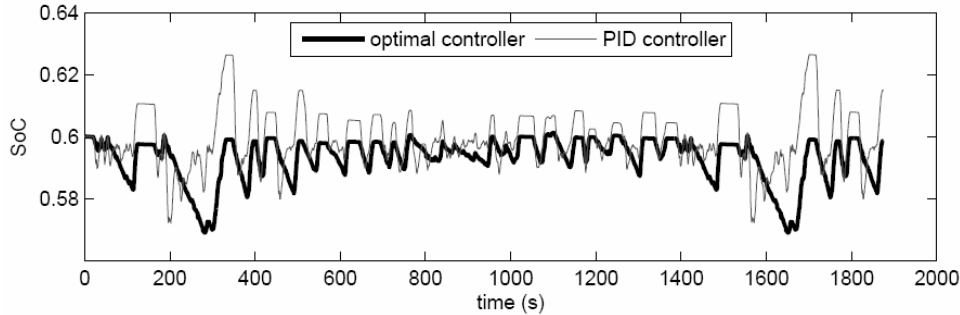
10 Conclusions

In this work, the problem of minimising fuel consumption in a series HEV is studied. Because of several advantages of model-based controllers, this type of controller is chosen for study. Based on a simple mathematical model (the controller-oriented model), an optimisation problem is defined. PMP was chosen as the optimal control method with known driver inputs. After off-line optimisation, it was observed that there is a simple relationship between the optimal solution and the power demand. This relation is found to be optimal, and a simple feedback controller is designed based on this relationship.

The literature mentions that future driving conditions have to be known in advance in order to obtain optimal results. In this work, a method is presented to tune the controller parameters, but the full drive cycle in not required. If both the time when positive power is required (during acceleration and cruising) and the amount of regenerative braking until the next stop are known, it is possible to achieve a result which is just slightly inferior to the global optimal solution.

Finally, the performance of the controller has been evaluated by a high-fidelity physics-based model, developed in MapleSim. Low-level controllers are used to ensure that the powertrain components follow the supervisory controller setpoints. The simulations show that the controller can predict real vehicle behaviour with acceptable accuracy. Moreover, this controller performs better than a well-tuned PID controller which tries to keep the SoC near the reference value to extend the battery life. This reduction in fuel consumption has a considerable effect in saving fuel in a global scope.

Acknowledgements

References

Ambuhl, D. and Guzzella, L. (2009) 'Predictive reference signal generator for hybrid electric vehicles', *IEEE Transactions on Vehicular Technology*, November, Vol. 58, No. 9, pp.4730–4740.

Barth, M., Scora, G. and Younglove, T. (2003) 'Intelligent off-board management of vehicle operating parameters', *Proceedings of the 2003 IEEE Intelligent Transportation Systems*, October, Vol. 1, pp.352–357.

Borhan, H., Zhang, C., Vahidi, A., Phillips, A., Kuang, M. and Di Cairano, S. (2010) 'Nonlinear model predictive control for power-split hybrid electric vehicles', *2010 49th IEEE Conference on Decision and Control (CDC)*, December, pp.4890–4895.

Boyali, A. and Guvenc, L. (2010) 'Real-time controller design for a parallel hybrid electric vehicle using neuro-dynamic programming method', *2010 IEEE International Conference on Systems Man and Cybernetics (SMC)*, October, pp.4318–4324.

Burke, A. (2007) 'Batteries and ultracapacitors for electric, hybrid, and fuel cell vehicles', *Proceedings of the IEEE*, April, Vol. 95, No. 4, pp.806–820.

Chin, H. and Jafari, A. (2010) 'Design of power controller for hybrid vehicle', *42nd Southeastern Symposium on System Theory (SSST), 2010*, March, pp.165–170.

Cipollone, R. and Sciarretta, A. (2006) 'Analysis of the potential performance of a combined hybrid vehicle with optimal supervisory control', *2006 IEEE Computer Aided Control System Design*, *2006 IEEE International Conference on Control Applications*, *2006 IEEE International Symposium on Intelligent Control*, October, pp.2802–2807.

Dao, T-S. and McPhee, J. (2011) 'Dynamic modeling of electrochemical systems using linear graph theory', *Journal of Power Sources*, Vol. 196, No. 23, pp.10,442–10,454.

Dao, T-S., Seaman, A. and McPhee, J. (2010) 'Mathematics-based modeling of a series-hybrid electric vehicle', *5th Asian Conference on Multibody Dynamics*, August.

Dao, T-S., Seaman, A., McPhee, J. and Shigematsu, K. (2011) 'Symbolic modeling of a series-hybrid electric vehicle for real-time simulations', *JSAE Annual Congress*, May.

Dextreit, C., Assadian, F., Kolmanovsky, I.V., Mahtani, J. and Burnham, K. (2008) 'Hybrid electric vehicle energy management using gametheory', SAE technical paper.

Geering, H.P. (2007) *Optimal Control with Engineering Applications*, Springer, Berlin Heidelberg, New York.

Gonder, J.D. (2008) 'Route-based control of hybrid electric vehicles', *SAE 2008 World Congress*.

Kim, N., Cha, S. and Peng, H. (2011a) 'Optimal control of hybrid electric vehicles based on Pontryagin's minimum principle', *IEEE Transactions on Control Systems Technology*, Vol. 19, No. 5, pp.1279–1278.

Kim, N., Cha, S.W. and Peng, H. (2011b) 'Optimal equivalent fuel consumption for hybrid electric vehicles', *IEEE Transactions on Control Systems Technology*, No. 99, pp.1–9.

Lamberson, D.M. (2003) 'Torque management of gasoline engines', Master's thesis, University of California at Berkeley.

Li, W., Xu, G., Tong, H. and Xu, Y. (2007) 'Design of optimal, robust energy management strategy for a parallel HEV', *IEEE International Conference on Robotics and Biomimetics, 2007 (ROBIO 2007)*, December, pp.1894–1899.

Lin, C-C., Peng, H. and Grizzle, J. (2004) 'A stochastic control strategy for hybrid electric vehicles', *Proceedings of the 2004 American Control Conference*, 30 June to 2 July 2004, Vol. 5, pp.4710–4715.

Lin, C-C., Peng, H., Grizzle, J. and Kang, J-M. (2003) 'Power management strategy for a parallel hybrid electric truck', *IEEE Transactions on Control Systems Technology*, November, Vol. 11, No. 6, pp.839–849.

Lin, X., Banvait, H., Anwar, S. and Chen, Y. (2010) 'Optimal energy management for a plug-in hybrid electric vehicle: real-time controller', *American Control Conference (ACC)*, 30 June to 2 July 2010, pp.5037–5042.

Musardo, C., Rizzoni, G. and Staccia, B. (2005) 'A-ECMS: an adaptive algorithm for hybrid electric vehicle energy management', *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC'05)*, December, pp.1816–1823.

Ngo, D.V., Hofman, T., Steinbuch, M. and Serrarens, A. (2010) 'An optimal control-based algorithm for hybrid electric vehicle using preview route information', *American Control Conference (ACC)*, 30 June to 2 July 2010, pp.5818–5823.

Piccolo, A., Ippolito, L., zo Galdi, V. and Vaccaro, A. (2001) 'Optimisation of energy flow management in hybrid electric vehicles via genetic algorithms', *Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Vol. 1, pp.434–439.

Razavian, R.S., Azad, N.L. and McPhee, J. (2012) 'On real-time optimal control of a series hybrid electric vehicle with an ultra-capacitors', *American Control Conference (ACC)*.

Saeedi, M. (2010) 'A mean value internal combustion engine model in Maplesim', Master's thesis, Department of Mechanical and Mechatronics Engineering, University of Waterloo.

Sampathnarayanan, B., Serrao, L., Onori, S., Rizzoni, G. and Yurkovich, S. (2009) 'Model predictive control as an energy management strategy for hybrid electric vehicles', *ASME Conference Proceedings*, No. 48937, pp.249–256.

Sciarretta, A. and Guzzella, L. (2007) 'Control of hybrid electric vehicles', *Control Systems*, April, Vol. 27, No. 2, pp.60–70, IEEE.

Sciarretta, A., Back, M. and Guzzella, L. (2004) 'Optimal control of parallel hybrid electric vehicles', *IEEE Transactions on Control Systems Technology*, May, Vol. 12, No. 3, pp.352–363.

Serrao, L. and Rizzoni, G. (2008) 'Optimal control of power split for a hybrid electric refuse vehicle', *American Control Conference*, June, pp.4498–4503.

Serrao, L., Onori, S. and Rizzoni, G. (2009) 'ECMS as a realization of Pontryagin's minimum principle for HEV control', *American Control Conference, 2009, ACC'09*, June, pp.3964–3969.

Serrao, L., Onori, S. and Rizzoni, G. (2011) 'A comparative analysis of energy management strategies for hybrid electric vehicles', *Journal of Dynamic Systems, Measurement, and Control*, Vol. 133, No. 3, p.031012.

Shi, L. and Crow, M. (2008) 'Comparison of ultracapacitor electric circuit models', *2008 IEEE Power and Energy Society General Meeting – Conversion and Delivery of Electrical Energy in the 21st Century*, July, pp.1–6.

Slotine, J. and Li, W. (1991) *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, New Jersey.

Soltis, A. and Chen, X. (2003) 'A new control strategy for hybrid electric vehicles', *Proceedings of the 2003 American Control Conference*, Vol. 2, Nos. 4–6, pp.1398–1403.

Stockar, S., Marano, V., Rizzoni, G. and Guzzella, L. (2010) 'Optimal control for plug-in hybrid electric vehicle applications', *American Control Conference (ACC)*, 30 June to 2 July 2010, pp.5024–5030.

Taghavipour, A., Azad, N.L. and McPhee, J. (in press) 'An optimal power management strategy for power split plug-in hybrid electric vehicles', *International Journal of Vehicle Design*.