A control benchmark on the energy management of a plug-in hybrid electric vehicle

A. Sciarretta [a,*], L. Serrao [c,1], P.C. Dewangan [a,e], P. Tona [a], E.N.D. Bergshoeff [h,5], C. Bordons [i], L. Charmpa [e,5], Ph. Elbert [d], L. Eriksson [f], T. Hofman [h], M. Hubacher [h], P. Isenegger [h], F. Lacandia [g,6], A. Laveau [e,4], H. Li [e,3], D. Marcos [i], T. Nüesch [d], S. Onori [g,7], P. Pisu [b], J. Rios [b], E. Silvas [h], M. Sivertsson [f], L. Tribioli [g,2], A.-J. van der Hoeven [h], M. Wu [e,3]

[a] IFP Energies nouvelles, France
[b] Clemson University, United States
[c] Dana Corporation, Italy
[d] ETH Zurich, Switzerland
[e] IFP School, France
[f] Linköping University, Sweden
[g] Ohio State University, United States
[h] TU Eindhoven, The Netherlands
[i] University of Sevilla, Spain

ARTICLE INFO

ABSTRACT

A benchmark control problem was developed for a special session of the IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling (E-COSM 12), held in Rueil-Malmaison, France, in October 2012. The online energy management of a plug-in hybrid-electric vehicle was to be developed by the benchmark participants. The simulator, provided by the benchmark organizers, implements a model of the GM Voltec powertrain. Each solution was evaluated according to several metrics, comprising of energy and fuel economy on two driving profiles unknown to the participants, acceleration and braking performance, computational performance. The nine solutions received are analyzed in terms of the control technique adopted (heuristic rule-based energy management vs. equivalent consumption minimization strategies, ECMS), battery discharge strategy (charge depleting–charge sustaining vs. blended mode), ECMS implementation (vector-based vs. map-based), ways to improve the implementation and improve the computational performance. The solution having achieved the best combined score is compared with a global optimal solution calculated offline using the Pontryagin's minimum principle-derived optimization tool HOT.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Energy management of hybrid electric vehicles (HEV) is nowadays a more-than-ten-years-old field of research in control engineering (Baumann, Rizzoni, & Washington, 1998; Brahma, Guezennec, & Rizzoni, 2000; Hofman, Ebbesen, & Guzzella, 2012; Kleimaier & Schroder, 2002; Koot et al., 2005; Lin, Kang, Grizzle, & Peng, 2001; Paganelli et al., 2000; Salman, Schouten, & Kheir, 2000; Sciarretta, Back, & Guzzella, 2004; Sciarretta & Guzzella, 2007). Indeed, energy management is a control task since it consists in determining the setpoints (mostly, torque) to the various power converters (internal combustion engine, electric machines with their power electronics, mechanical transmission devices, electrical power converters, etc.) that constitute the HEV powertrain. These setpoints are chosen by the energy management strategy (EMS) in order to fulfil the driver's request and at the same time exploit the remaining degrees of freedom to obtain the most suitable powertrain behaviour. "Optimal" EMS that have been disclosed in these years are aimed at minimizing an objective function that typically represents the overall fuel consumption, but might include pollutant emissions, battery life degradation, under several constraints concerning battery charge,

---

drivability, etc. In particular, charge-sustaining or autonomous HEV implies that the battery State Of Charge (SOC) at the end of a vehicle mission is required to be as close as possible to its initial value. A mathematical formulation of such a control problem has been posed in terms of optimal control (Ambühl et al., 2007; Hofman, Steinbuch, Serrarens, & van Druten, 2008; Kim, Cha, & Peng, 2011; Serrao, Onori, & Rizzoni, 2009; van Berkel, Hofman, Vroemen, & Steinbuch, 2012) and numerous practical implementations for various architectures such as parallel (Lin et al., 2001; Musardo & Rizzoni, 2005; Pisu & Rizzoni, 2007; Salman et al., 2000; Sciarretta et al., 2004; Sivertsson, Sundström, & Eriksson, 2011), series (Anatone, Cipollone, Donati, & Sciarretta, 2005; Pisu & Rizzoni, 2005), and combined HEV (Borhan & Vahidi, 2010; Cipollone & Sciarretta, 2006; Hofman et al., 2008; Liu & Peng, 2006) have been presented.

The class of plug-in HEV (PHEV), where the battery can be recharged from an external source (grid) also, has attracted less research than charge-sustaining HEV, although pioneering papers have already treated this topic in terms of optimal control and presented simulation or experimental results (Larsson, Johannesson, & Egardt, 2010; O'Kneefe & Markel, 2006; Stockar, Marano, & Canova, 2011; Serrao et al., 2013; Tulpule, Marano, & Rizzoni, 2009). The specific difficulty in this class of EMS is to generate an optimal discharge of the battery. Indeed it is known that a simple CD–CS strategy, i.e., a fully electrical operation (charge depleting, CD) followed by a Charge-Sustaining (CS) operation from when the battery is discharged onwards, although attractive as it allows presenting the HEV as an "electric vehicle", is far from being optimal from a fuel economy standpoint. Therefore, progressive battery discharge ("blended-mode") operation is expected to be the output of an optimal EMS.

While several EMS have been generally presented in the scientific literature, a way to compare them is obviously not generally available, since studied systems and driving conditions vary from case to case. Clearly, the ability to make direct comparisons between systems, employing these algorithms, would be highly beneficial for the scientific community to verify common claims concerning both performance (optimality) and implementability (flexibility or reusability, easiness of calibration and implementation, etc.) of EMS and focus future efforts in the most promising directions. Such comparison tools have been deployed for other control applications (Spencer, Dyke, & Deoskar, 1998) and consist of benchmark control problems that are typically solved using simulation models replacing real systems. As a second step, functional solutions might be benchmarked on physical systems as well. Recently, the Japanese automotive societies JSAE and SICE have jointly proposed a benchmark HEV control problem (Yasui, 2012) based on a simulator of a combined hybrid (Prius-like) vehicle and driver and aimed at challenging academic researchers.

This paper presents a benchmark PHEV control problem and analyzes a set of solutions. The benchmark was developed for a special session of the IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling (E-COSM '12), held in Rueil-Malmaison, France, in October 2012. The participation of nine teams presenting their own solution demonstrated the interest in such initiative. All teams were provided with a fully functional simulator of a PHEV, and were to implement an EMS to optimize a set of criteria. The simulator (see Section 2) is of the quasi-static type and accounts for longitudinal vehicle dynamics and battery SOC dynamics, while the engine and electric machines are modeled using stationary maps. Solutions were to be submitted in the form of a Simulink block with a specific format (inputs/outputs/solver). The evaluation of the strategies was done on the basis of the fuel and energy consumption for two realistic driving cycles that were unknown to the participants, as well as acceleration performance and controller runtime performance (details in

Section 3). In the cycle tests, the battery is completely charged at the beginning of the cycle and can be depleted at the end of the cycle. The participants were able to make use of some approximated information about the cycle, namely the total distance and average speed, which could be easily retrieved from a GPS device. Given the focus of the benchmark problem, this information was included in the simulator as perfectly known, albeit in practice it is affected by measurement uncertainties. A special jury, presided by the holder of the IFP School – Fondation Tuck Chair on Hybrid vehicle and energy management, defined the two test cycles and guaranteed the correct evaluation of the solutions to be benchmarked. The nine solutions evaluated are presented in Section 4, while Section 5 discusses the results obtained. The software developed for this benchmark will be made available on the web site www.ecosm12.org.

2. Simulator

Although a detailed description of vehicle propulsion systems would require the modeling of several dynamic phenomena, it has long been recognized (Guzzella & Sciarretta, 2013, Chap. 2) that for the purpose of fuel economy estimation, quasi-static models, i.e., based on efficiency maps measured under stationary operation of the various components, suffice to a large extent. For such a reason, quasi-static models are largely used to design and pre-assess energy management strategies of HEV, as per the literature cited within the paper. Of course, the mutual relationship between the EMS and typical transient maneuvers would not be represented by such models, but if the main focus is on the fuel economy, they can still reasonably serve to compare the global performance of different EMS. These are also the reasons why the present benchmark PHEV control problem is based on a quasi-static simulator.

The simulator provided implements a model of Chevrolet Volt, validated with published GM data, which are well reflected in the simulation results (Falières et al., 2011; Grebe & Nite, 2011; Miller, Holmes, Conlon, & Savagian, 2011; Parrish, Elankumaran, Gandhi, & Nance, 2011). The simulator implements three main blocks (Fig. 1): (1) driving cycle, which computes the torque demand based on the specified driving cycle, and also outputs the preview information (nominal distance and average speed); (2) control strategy (EMS), which was to be filled with the benchmark solutions respecting given input and output ports; (3) vehicle and powertrain model, which contains the quasi-static model of the powertrain and vehicle dynamics.

The participants had access to the content of the driving cycle and the vehicle model block, but they were not to be modified. Only their respective outputs could be used for developing the EMS, and only the controller block was to be submitted at the end.

2.1. Powertrain model: GM Voltec

The powertrain architecture powering the Chevrolet Volt consists of a power-split, planetary-based system, named Voltec and shown in Fig. 2. Three clutches (C1, C2, C3) allow connecting or disconnecting the internal combustion engine (ICE), the generator (GEN) and the main traction machine (MOT). Both electric machines can actually work in both motoring and generating mode, and for both of them the sign convention is that positive torque and positive electric power indicate motoring operation.

The powertrain can operate in the following modes (Falières et al., 2011; Grebe & Nite, 2011; Parrish et al., 2011):

1. *One-motor EV (C1 locked, C2 open, C3 open, engine off)*: MOT alone propels the vehicle, powered by the battery. The planetary gear
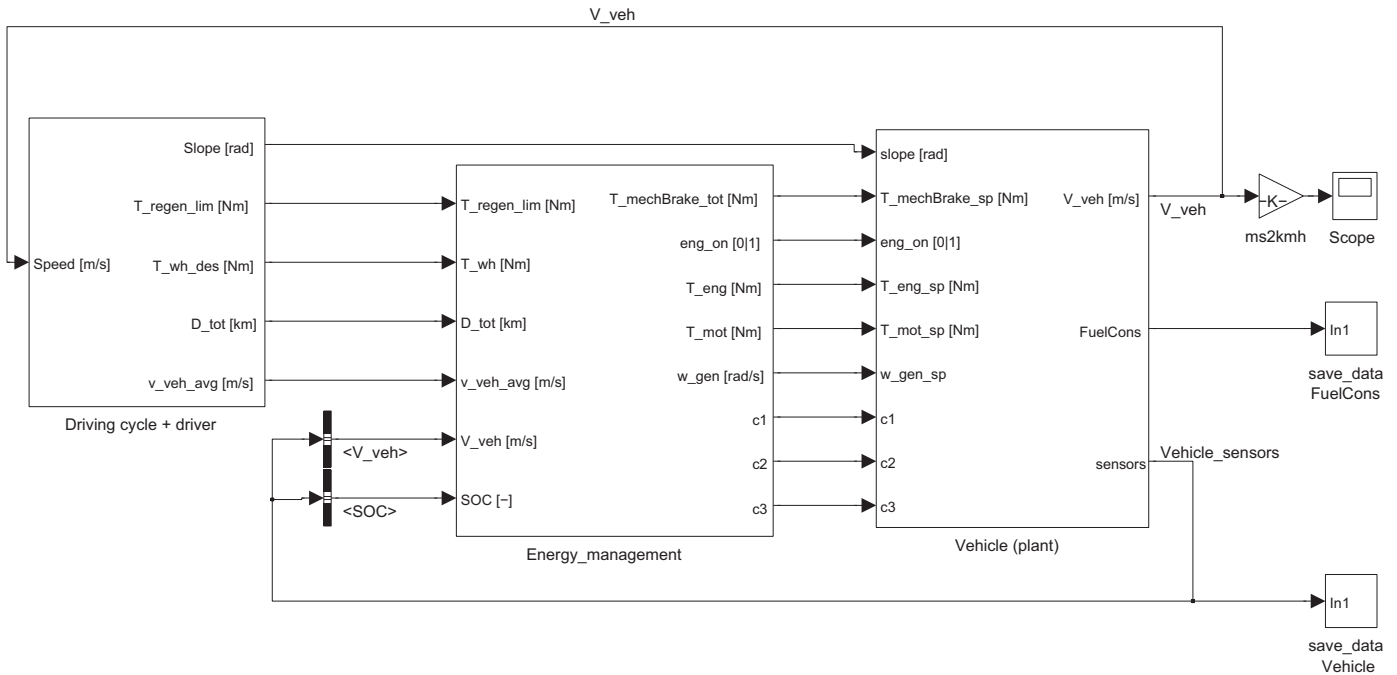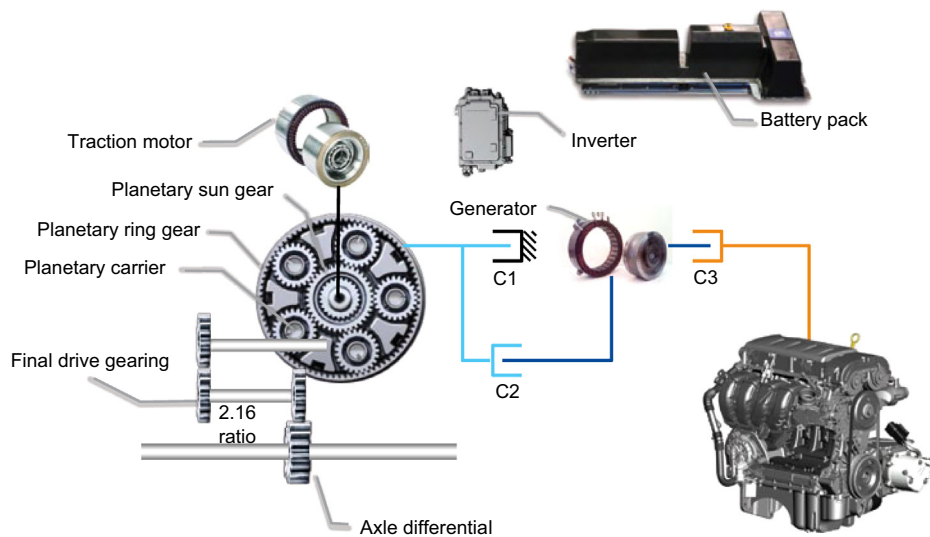
Fig. 1. Benchmark simulator layout.

Fig. 2. Kinematic architecture of GM's Voltec (Grebe & Nite, 2011).

set introduces a fixed reduction between the machine MOT and rest of the driveline (final ratio and differential).

2. *Two-motor EV (C1 open, C2 locked, C3 open, engine off).* In this case, the machine GEN acts on the planetary ring through C2 and thus it changes the gear ratio between MOT and the powertrain output. This mode is useful to reduce MOT speed at high vehicle speed, thus increasing overall powertrain efficiency by combining the use of both electric machines.

3. *Range-extender mode (C1 locked, C2 open, C3 locked, engine on).* This is a traditional series-HEV mode: the engine and generator are connected and produce electric power; MOT alone propels the wheels.

4. *Power-split mode (C1 open, C2 locked, C3 locked, engine on).* In this mode, the three machines are all connected together with a variable speed ratio that depends on the generator speed. The mode allows transmitting mechanical power directly from the

engine to the wheels, thus resulting in overall higher efficiency than a pure series mode.

As a whole, the vehicle and powertrain model takes as inputs the outputs of the EMS, i.e., engine torque setpoint T_e (`T_eng_sp` in Fig. 1), motor torque setpoint T_m (`T_mot_sp`), generator speed setpoint ω_g (`w_gen_sp`), brake torque T_{br} (`T_mechBrake_sp`), engine on/off signal (`eng_on`), and clutch commands C1 to C3 (`c1,c2,c3`). The outputs are vehicle speed v (`V_veh`), fuel consumption $\overset{*}{m}_f$ (`FuelCons`), and the battery SOC ξ (`SOC`), to be used within the EMS.

2.1.1. Vehicle model

The vehicle submodel takes the wheel torque T_{wh} from the powertrain and the brake torque T_{br} command as inputs and yields

the vehicle speed v and the wheel speed ω_{wh} as output. The submodel implements vehicle's longitudinal dynamics

$$\dot{v} = \frac{r_{wh}}{J_v}[T_{wh} + T_{br} - m_v r_{wh} g \sin \alpha - r_{wh}(c_0 + c_1 v + c_2 v^2)] \quad (1)$$

where α is the road slope and J_v is the vehicle moment of inertia.

The relevant vehicle parameters have been extracted from Grebe and Nite (2011) or estimated from similar vehicles and are listed in Table 1. All simulations are carried out by considering 97 kg in addition to the vehicle mass (75 kg driver and 35 l, i.e., 22 kg full tank of fuel).

2.1.2. Transmission model

The transmission consists of a differential gear ($R_d = 2.16$), a planetary gear set (PGS), and a node (unit ratio). The generator is connected to the ring (r), the motor is connected to the sun (s) and the transmission output is the satellite carrier (c) of the PGS. The node connects the engine and the generator.

The forward transmission submodel takes the generator and wheel speeds, ω_g and ω_{wh}, as well as the engine and motor torques, T_e and T_m as inputs, and yields the engine and motor speeds, ω_e and ω_m, as well as the generator and wheel torques, T_g and T_{wh}, as outputs.

The kinematic relation between the speeds of three elements of the planetargy gear set is

$$\rho \cdot \omega_r + \omega_s = \omega_c \cdot (\rho + 1), \quad (2)$$

where ρ is the ratio between the number of teeth of the ring and the sun gear: $\rho = N_r/N_s = 83/37 = 2.24$. The torque relations imposed by the planetary gear set are

$$\frac{T_r}{\rho} = \frac{T_c}{\rho + 1} = T_s \quad (3)$$

The connection between PGS elements and power converters depends on the mode in which the powertrain is operated. See Table 2 for mode-dependent correspondences.

The simulator is implemented using these relations, including the gear efficiencies but neglecting the dynamics of the machines and the inertia of the gears.

Table 1
Vehicle parameters.

Curb weight m_v	1812 kg
Road law coefficient c_0	105.95 N
Road law coefficient c_1	0.01 N s m^{-1}
Road law coefficient c_2	0.4340 N s^2 m^{-2}
Wheel radius r_{wh}	33 cm
Vehicle inertia J_v	207 kg m^2
Wheelbase	2685 mm
Center of gravity height	550 mm
Front/rear static weight distribution	49%/51%

Table 2

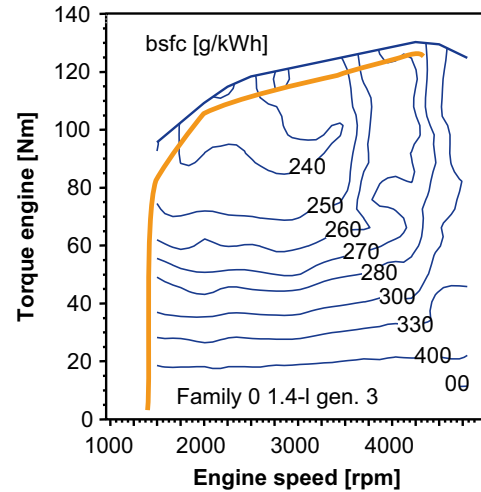| Mode 1 | $\omega_r = 0, \quad \omega_e = 0$ |
	$T_g = 0$
Mode 2	$\omega_r = \omega_g, \quad \omega_e = 0$
	$T_g = \rho \cdot T_m$
Mode 3	$\omega_r = 0, \quad \omega_e = \omega_g$
	$T_g = -T_e$
Mode 4	$\omega_r = \omega_g, \quad \omega_e = \omega_g$
	$T_g = \rho \cdot T_m - T_e$
All modes	$\omega_m = \omega_s, \quad \omega_c = R_d \cdot \omega_{wh}$
	$T_m = T_s, \quad T_{wh} = R_d \cdot T_c$

Fig. 3. Engine fuel map (Grebe & Nite, 2011).

2.1.3. Engine model

The engine submodel takes ω_e and T_e as inputs and yields the fuel consumption $\dot{m}_f^*$. Since a quasi-static modelling approach is used, the engine is represented by its fuel maps, shown in Fig. 3. Torque and speed limits are enforced. Special treatments are used to represent cranking and idle phases. When the engine is switched on, the model generates a predefined negative torque (-25 Nm) for a short period (1 s), after which the torque generated matches the setpoint.

2.1.4. Motor and generator model

The motor submodel takes the motor torque T_m and speed ω_m as inputs and yields the electric power P_m as output. Similarly for the generator submodel T_g and ω_g are the inputs and P_g is the output. The electric machines are represented by their efficiency maps, which are shown in Fig. 4. Torque and power limits are enforced. The generator model also enforces a maximum rate of change of ω_g (200 rad/s^2) to represent machine inertia.

2.1.5. Battery model

The battery submodel takes the overall electrical power $P_b = P_g + P_m$ and the current SOC as input and yields the SOC at next time step as well as the inner or electrochemical battery power P_{ech}. The model is based on a simple equivalent circuit composed of a voltage source V_{oc} and a resistance R_b in series, both functions of the SOC (I_b is the battery current),

$$P_{ech} = V_{oc} I_b, \quad I_b = \frac{V_{oc}}{2R_b} - \sqrt{\frac{V_{oc}^2 - 4R_b P_b}{4R_b^2}} \quad (4)$$

$$\dot{\xi} = -\frac{I_b}{Q_b} \quad (5)$$

where Q_b is the charge capacity. Power, current, and voltage limits are enforced. Moreover, SOC operation within a prescribed window is enforced.

In the case of the Volt, only the basic battery parameters, collected in Table 3, are published. For the lack of specific data, the internal resistance and the open circuit voltage characteristic of a single Li-ion cell are assumed to be the same as the experimental data in Do (2010).
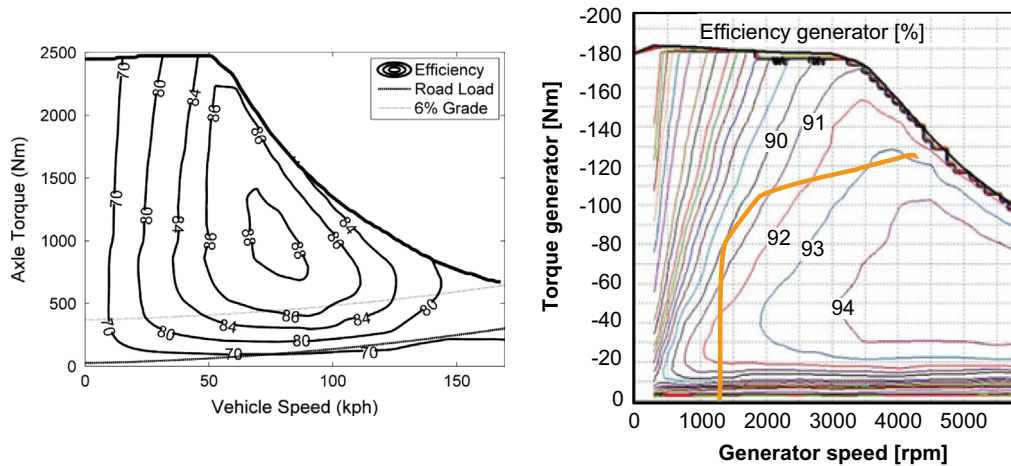
Fig. 4. Combined motor-transmission efficiency map (top) (Miller et al., 2011) and generator efficiency map (bottom) (Grebe & Nite, 2011).

Table 3
Battery parameters.

Parameter	Value	Source
Total energy capacity	16 kWh	Grebe and Nite (2011)
Total nominal voltage	360 V	Grebe and Nite (2011)
SOC range	65%	Grebe and Nite (2011)
Number of cells in series	96	Parrish et al. (2011)
Number of strings in parallel	3	Parrish et al. (2011)
Peak current	400 A	Parrish et al. (2011)
Peak power (charge)	110 kW	
Peak power (discharge)	−60 kW	

Table 4
Scoring metrics.

	Metric	Weight (%)
Performance (30%)	Acceleration 0–100 km/h (s)	7.5
	Acceleration 70–120 km/h (s)	7.5
	Acceleration 0–1000 m on 4% slope (s)	7.5
	Braking distance from 1000 km/h (m)	7.5
Energy and economy (50%)	Total energy use (fuel+electricity) (MJ)	15
	Fuel consumption (MJ)	20
	Tailpipe emissions (not modeled)	0
	Well-to-wheel CO_2 emissions (kg)	15
Computational performance (20%)	Processor use (s)	10
	Memory use (kB)	10

2.2. Driver model

The driver model is responsible for yielding the powertrain torque request T_d (T_wh_des in Fig. 1) to the EMS as a function of the vehicle speed v (V_veh) measured from the vehicle model output. For the two cycle tests, the model is a cascade of (i) a feedforward calculation of the torque request as a function of the desired speed v, and (ii) a PI regulator as a function of the difference between cycle speed and v. The term (i) is calculated as

$$T_d = \begin{cases} T_{d,tot}, & T_{d,tot} \geq 0, \\ (1-\sigma)T_{d,tot} - \frac{T_{d,tot}H_{CG}}{m_v g r_{wh} B_w}, & T_{d,tot} < 0 \end{cases} \quad (6)$$

$$T_{d,tot} = \frac{\dot{v} J_v}{r_{wh}} + r_{wh}(c_0 + c_1 v + c_2 v^2 + m_v g \sin \alpha) \quad (7)$$

where the wheel torque request $T_{d,tot}$ is obtained from the inversion of (1), H_{CG} is the height of the center of gravity, B_w the wheelbase, and σ the static weight distribution (see Table 1). The proportional and integral gains of the term (ii) are tuned as $K_P = 500$ N s, $K_I = 1$ N.

For acceleration tests, only the PI regulator is used, with a reference speed set to a very high value in order to enforce full torque demand. The test end is detected as a function of the speed or the distance covered, according to the particular test. Conversely, the braking test is split into two parts, with a first part to reach 100 km/h through a PI regulator, then a second part with constant negative torque request.

2.3. EMS

The EMS block to be filled by benchmark participants takes as inputs the driver torque demand T_d output by the driver's model

(T_wh in Fig. 1) and the vehicle speed v (V_veh) and the SOC ξ (SOC) yielded by the vehicle and powertrain model. The expected outputs of the EMS model are the engine and motor torque setpoints, the generator speed setpoint, the clutch commands, the engine on/off state, and the mechanical brake torque, all sent to the vehicle and powertrain model.

3. Benchmarking

3.1. Scoring metrics

Each solution has been evaluated according to several metrics, listed in Table 4. The overall score is obtained by weighting each metric by the factor shown in the table. The weights are intended to be a compromise between energy efficiency (50%) and expectations of manufacturers (20%) and end users (30%). Since they intervene only in the final evaluation stage of the benchmark, they can be easily modified in future versions of the benchmark to reflect a different focus. Although a correlation is expected between energy efficiency sub-metrics, they are accounted separately in order to emphasize the effects of residual battery energy and the CO_2 factor. The actual scoring is obtained by normalizing the result obtained in each metric with respect to the average value for that metric calculated over all valid solutions. Note that all scores are best when minimized.

3.1.1. Performance

Four performance tests are enforced in the driver's model, see Section 2.2. Elapsed simulation time or distance covered until the simulation stop are monitored and used as metrics.

Table 5
Data for CO_2 emissions.

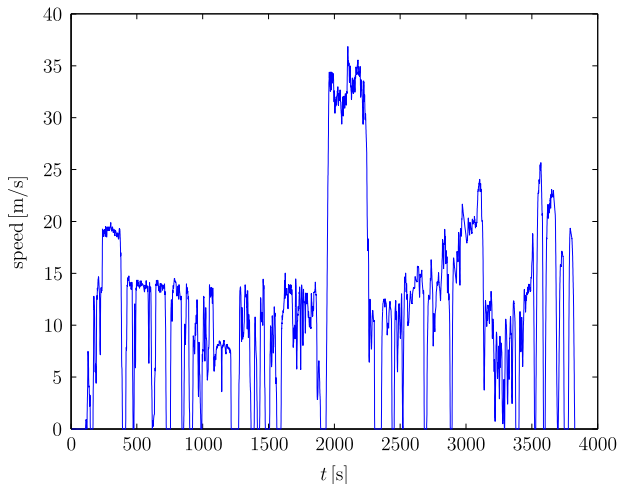|	
Gasoline well-to-tank emissions (TNO Report, 2006)	12.5 gCO_2/MJ of fuel
Gasoline combustion (TNO Report, 2006)	73.4 g CO_2/MJ of fuel
Electricity production (emissions for electricity and heat production (International Energy Agency report, 2011), average 2007–2009)	Europe average: 94.7 g CO_2/MJ France: 24.7 g CO_2/MJ US: 147.5 g CO_2/MJ China: 207.8 g CO_2/MJ World average: 140 g CO_2/MJ

Fig. 5. Aachen cycle.

3.1.2. Energy and economy

These metrics are the sum of the results in two different driving cycles, based on real-world data, whose length (see Section 3.2) exceeds the all-electric range of the vehicle. In each cycle, the initial SOC must match the value given in the data file provided with the simulator, while the final SOC must be at least equal to a target final value. CO_2 emissions were calculated from energy use results as the total emissions from fuel combustion, fuel production (well-to-tank), and electricity production. For electricity, Europe average emissions were to be taken into account (data for France, US, and China were also provided for information). The coefficients in Table 5 were used to compute the CO_2 emissions. Tailpipe emissions were not considered for simplicity (unavailable reliable modeling) and due to their weak dependency on the EMS for warm engine conditions.

3.1.3. Computational performance

Processor use and memory use were monitored on the computer running the simulation. The former was evaluated using MATLAB functions `tic` and `toc` before and after each test, the latter using MATLAB function `sldiagnostics`. Only for fuel economy tests, the actual results are averaged over four runs, in order to reduce the effects of other possible instances running on the test computer. Counting of floating point operations could be used to make these metrics less dependent on the particular machine. However, albeit an option for future versions of the benchmark, this solution has not implemented due to the lack of a simple flop counting function in the most recent versions of MATLAB.

3.1.4. Solution validity

At all times, all component limitations detailed in the data file (max/min power, torque, state of charge, speed etc.) were to be
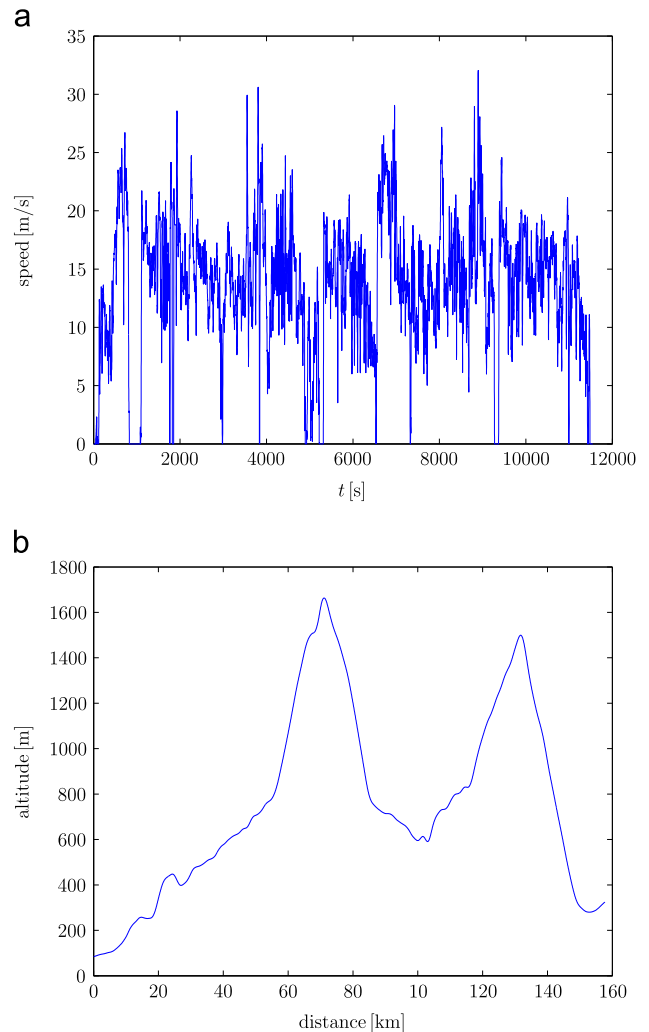

Fig. 6. Arco to Merano cycle: (a) speed profile and (b) altitude profile.

respected in all components; failure to do so during any simulation invalidated the corresponding results.

3.2. Driving cycles

Driving cycles were defined by the benchmark jury as "surprise" cycles, not communicated in advance to the participants. The two real-life cycles selected are shown in Figs. 5 and 6. The first cycle is actually the combination of two trips recorded in the German city of Aachen, namely, an urban trip spanning 16.2 km and a mixed-drive trip spanning 28.4 km. Elevation has been set to zero for the whole cycle. In order to allow full battery depletion if the HEV would be operated in EV modes, the combined Aachen cycle is repeated three times.

Table 6
Cycle statistics.

	Aachen urban[a]	Aachen mixed[a]	Arco–Merano
Length (km)	16.2	28.4	157.7
Max speed (km/h)	71.6	132.6	115.4
Average speed (km/h)	30.3	53.8	49.2
Duration (s)	1926	1871	11549
Max acceleration (m/s^2)	3.9	4.7	5.7
Max deceleration (m/s^2)	3.0	4.6	7.4
No. of vehicle stops	14	11	15
Total stop time (s)	430	185.5	671.5
Total stop time (%)	22	10	6
Altitude variation (m)	0	0	1579

[a] The actual test cycle consists of three repetitions of Aachen urban cascaded by Aachen mixed.

The second cycle is a trip recorded between Arco and Merano, Italy (in the Alps), spanning 157.7 km and including severe altitude variations that are visible in Fig. 6.

A summary of cycle statistics is listed in Table 6.

4. Proposed solutions

4.1. Introduction

The benchmark control problem can be summarized as to find a control policy $u(t)$ for $t=0,\ldots,T$ where T is any test duration, in such a way to minimize the score defined in Section 3. For each single metrics, this is equivalent to minimize a cost function $J(u)$, where the control vector $u \triangleq \{T_e, T_m, \omega_g, C_1, C_2, C_3, B_e, T_{br}\}$ defines the powertrain mode and the operating point of each component. In the example of the total energy use metrics

$$J = LHV \cdot \int_0^T \overset{*}{m}_f(u,w) \, dt + \int_0^T P_{ech}(u,w) \, dt \qquad (8)$$

where the disturbance vector $w(t) \triangleq \{v(t), T_d(t)\}$ comprises the exogenous inputs (not known in advance) and LHV is the lower heating value of the fuel. The minimization (8) is subject to the SOC constraint, $\xi(T) \geq \xi_{min}$, and other constraints. Formulated as such, the benchmark control problem is formally equivalent to a constrained optimal control problem.

Data published by GM (Grebe & Nite, 2011) show that, in the Volt embodied strategy, mode selection depends primarily on SOC status. For charge-depleting (CD) operation (high SOC), modes 1 or 2 are selected according to the vehicle speed and the required torque, see Fig. 7. When the SOC becomes low, charge-sustaining operation (CS) is activated, which consists of three modes 1, 3, and 4, again selected as a function of vehicles speed and desired torque. This SOC-regulating policy depends on the fact that the Volt is classified as a range-extended electric vehicle. In the benchmark control problem, participants were allowed to activate a "blended-mode" (BM) strategy to regulate the SOC, i.e., to start the engine before the SOC reaches a lower-end value. Of course, battery-recharge (BR) operation aimed at restoring the battery charge to a certain value was possible as well.

Nine control solutions were submitted to the benchmark problem. Table 7 summarizes the approaches implemented in the proposed solutions. The column "EMS technique" classifies the entries according to the type of control design. Two approaches are represented, namely, based on rules (heuristic) or based on optimal control theory and particularly on the Pontryagin Minimum Principle (PMP). Additionally, the table classifies the nine solution according to the discharge strategy, the mode selection strategy, and the operating point selection strategy.

All the used techniques are further illustrated below. More details on two solutions are provided in the ECOSM'12 papers (Marcos & Bordons, 2012; Sivertsson, 2012).

4.2. Heuristic controllers

Heuristic EMS are based on intuitive rules and correlations involving various vehicular variables. One guiding principle is to use the engine when its efficiency can be relatively high, while in less favorable conditions the electric modes should be given preference and the engine should be turned off. Moreover, when the engine is on, it should be operated in the highest possible efficiency regions (i.e., at high loads). Two common approaches to implement these intuitive principles are the map-based and the rule-based approach. In the map-based approach, the output setpoints are stored in multi-dimensional maps whose entries are measured quantities describing the state of the powertrain. In the rule-based approach, the EMS is either coded as a set of logic rules or implemented as a finite state machine (FSM).

In most solutions (S2, S4, S6, and S9), a two-level architecture is implemented. A high-level supervisor (see Fig. 8a) first selects the powertrain mode and the engine power setpoint P_e. The former output is either mapped (S2) as a function of v, T_t, and ξ and enhanced with exception rules, or calculated by a finite-state machine (S4, S9) with transition rules as a function of the same vehicular quantities. Engine power is calculated from rules as a function of the same variables and of the distance D_t, either in order to implement a CD–CS SOC-regulation strategy (S2) or a blended-mode strategy (S4) by minimizing a performance criterion.

The cascaded low-level controller evaluates the engine, motor, and generator setpoints, as well as the clutch commands, according to the mode selected and the engine power. While for mode 1 there is no degree of freedom, other modes require additional choices. Optimal operating point map of the engine is used to calculate T_e and ω_e as a function of P_e, while the other setpoints are used as such to fulfill transmission kinematics/dynamics. In S9 (a mixed-approach strategy) this task is performed using ECMS (see Section 4.3) for mode 4.

An alternative control architecture (used in S6, Fig. 8b) is such that the high-level supervisor selects CD operation or BR operation as a function of ξ, T_t, v and the information on the distance-to-go. In the case of CD operation, the low-level controller selects mode 1 or mode 2 by minimizing the electric consumption and determines electric setpoints. In the case of BR operation, the low-level controller operates the engine either to maximize the recharge efficiency or the recharge power and selects accordingly the setpoints using predefined maps.

A simpler approach (used in S5) switches between electric modes and engine-on modes as a function of the difference between ξ and a reference SOC profile based on the distance-to-go. If on, the engine is operated at its best-efficiency point and mode 4 is actuated, with exceptions possibly shifting the engine operating point or enforcing mode 3.

4.3. ECMS

The second approach used in the proposed solution is that based on optimal control. Since early 2000s it has been shown that the control problem of minimizing the fuel consumption of a HEV over a driving mission, while letting its SOC tend to a target value ξ_{ref} can be formulated as an optimal control problem similar to (8). Offline solutions to such a problem, that is, solutions that make use of the prior knowledge of the driving profile can be calculated using dynamic programming or directly solving the Euler–Lagrange formulation of optimal control. In the latter case, the optimal control
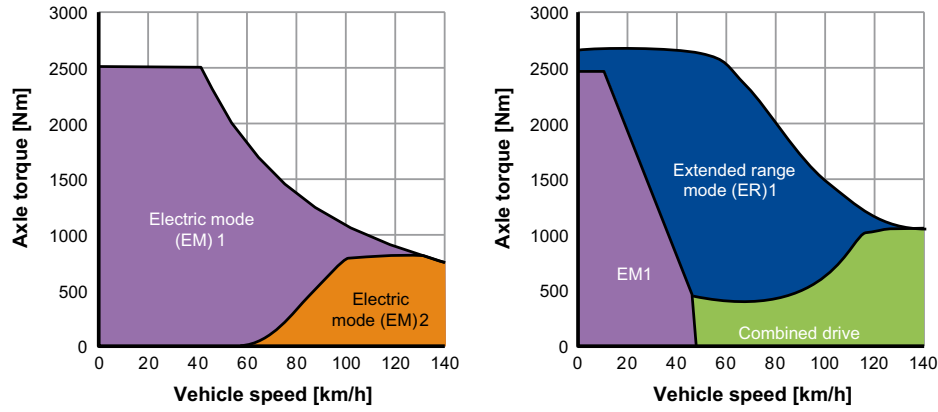
Fig. 7. Operating modes of GM's Voltec as a function of vehicle speed and desired torque (Grebe & Nite, 2011).

Table 7
List of solutions to the PHEV control benchmark.

Solution	EMS technique	Discharge strategy	Mode selection	Operating point selection
S1	ECMS	BM[a,c]	Map based	Map based
S2	Heuristic	CD–CS	Map based[b]	Map based
S3	ECMS	BM[c]	Vector based	Vector based
S4	Heuristic	BM	Rule based	Rule based
S5	Heuristic	BM	Rule based	Rule based
S6	Heuristic	BM	Rule based	Map based
S7	ECMS	BM[c]	Vector based	Vector based
S8	ECMS	BM[c]	Vector based	Vector based
S9	Mixed	?	Rule based	Rule based

[a] With 10% driving distance underestimation.
[b] Enhanced with exception rules.
[c] Adaptive ECMS with varying SOC reference.

policy is found as

$$u^o(t) = \arg \min_u H(u, w(t), \xi(t)) = \arg \min_u LHV \cdot \overset{*}{m}_f(u, w(t))$$
$$+ \lambda(t) \frac{d\xi(t)}{dt} \qquad (9)$$

under the state and costate dynamics

$$\frac{d\xi(t)}{dt} = -\frac{I_b(u, w, \xi)}{Q_b}, \quad \frac{d\lambda(t)}{dt} = -\frac{\partial H(u, w, \xi)}{\partial \xi}. \qquad (10)$$

Under the approximation $\partial V_{oc}/\partial \xi \approx 0$, $\partial R_b/\partial \xi \approx 0$, λ is a constant and the strategy (9) reduces to offline technique commonly known as Equivalent Consumption Minimization Startegy (ECMS)

$$u^o(t) = \arg \min_u LHV \cdot \overset{*}{m}_f(u, w(t)) + s^o P_{ech}(u, w(t)) \qquad (11)$$

with the equivalence factor defined as $s^o = -\lambda/(V_{oc}Q_b)$. The value s^o fulfills the global constraint on SOC, i.e., $\xi(T) = \xi_{ref}$.

4.3.1. Costate adaptation

While in offline ECMS s^o can be calculated using iterative root-finding algorithms, in online application it has to be continuously adapted and thus a variable $s(t)$ replaces s^o in (11). The most common approach for such adaptation, which proves to be directly related to Hamilton–Jacobi–Bellman optimal feedback control (Ambühl et al., 2007), is a feedback on current SOC:

$$s(t) = s_0 - k_p(\xi(t) - \xi_{ref}) - \ldots, \qquad (12)$$

where the ellipsis indicates possible integral or nonlinear terms.

All solutions (S1, S3, S7, S8, and S9 when ECMS is used) include in the right-hand side of (12) a term proportional to the integral of the error $\xi(t) - \xi_{ref}$. However, S8 uses a modified coding where s is

initialized to a value s_0, updated according to a space-depending version of (12) when a prefixed sampling distance is covered, and reset to s_0 at each time the reference SOC is tracked back (Onori, Serrao, & Rizzoni, 2010). A nonlinear term in the form of a tangent-like function of the error is added to the proportional-integral terms in S1 (Sivertsson, 2012). The feedforward term s_0 is variously calculated as a function of the average speed and the distance (S1, S8), the average speed only (S3), or it is constant (S7).

While ξ_{ref} clearly equals the initial value of SOC, $\xi(0)$, for CS operation, plug-in hybrids allow for discharging the battery and thus a different definition for ξ_{ref} must be used. Recent literature has addressed this problem and the most common strategy is to use a varying, i.e., decreasing $\xi_{ref}(t)$ aimed at implementing a BM operation. The decrease of ξ_{ref} should represent the advance in the mission and toward its end, $\xi_{ref}(T)$ should tend to ξ_{min}. All five solutions (S1, S3, S7, S8, and S9) adopt a linear variation of ξ_{ref} with the distance covered $D(t)$, namely,

$$\xi_{ref}(t) = \xi(0) - k_\xi \frac{D(t)}{D_{tot}}, \quad k_\xi = \xi(0) - \xi_{min}. \qquad (13)$$

This approach clearly require the prior estimation of the distance to be covered D_{tot}.

4.3.2. ECMS implementation

The vectorial implementation of strategy (11), used in S3, S6, and S8, requires discretizing the field of admissible values $u(t) \in \{u_q(t)\}$, $q = 1, \ldots, N$ at any time step. Thus,

$u^o(t) = u_{q^o(t)}(t)$, where

$q^o(t) = \arg \min_q H_q(t)$, and

$$H_q(t) = LHV \cdot \overset{*}{m}_f(u_q(t), w(t)) + s(t) \cdot P_{ech}(u_q(t), w(t)). \qquad (14)$$

The vector size N and the nature of u vary from solution to solution. For each q, u_q is obtained by imposing a set of predefined values to independent components of u (degrees of freedom, DoF) and calculating the dependent components through kinematic and other physical constraints. The powertrain mode is clearly a DoF itself, thus $N = \sum_{i=1}^4 N_i$. In mode 1 there is no additional degree of freedom and thus $N_1 = 1$. In mode 2 there is one speed DoF (either ω_m or ω_g). In mode 3 the DoF is engine power P_e since engine speed and torque are obtained by maximizing the engine–generator efficiency for a given power. In mode 4 there are one speed and one torque independent DoFs. As an example, S8 uses $N_{\{2,3,4\}} = 21$.

From (14), it is clear that $q^o(t)$ and thus $u^o(t)$ depend only on $w(t)$ and $s(t)$. Since $w(t)$ is a two-dimensional vector (speed and torque at the wheels), the energy management output is completely determined by three variables. While the dependency of the
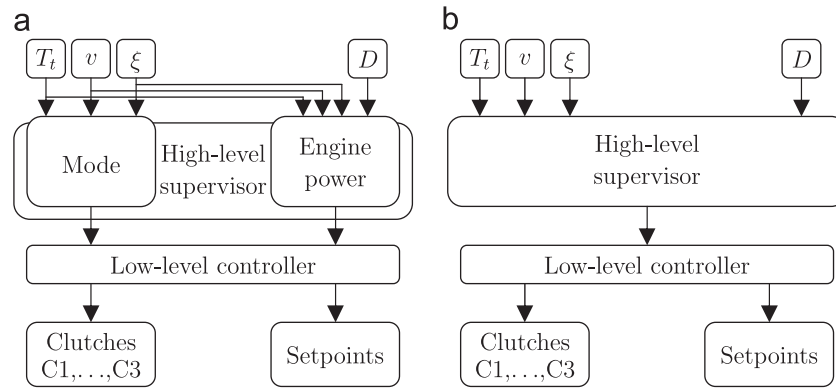
Fig. 8. General heuristic strategy flowchart (a) and variant (b).

Hamiltonians $H_q(t)$ on SOC is neglected in the ECMS, $s(t)$ does depend on $\xi(t)$, as well as on the distance $D(t)$. This fact suggests decoupling the evaluation of the energy management output in a two-step process, where first $s(t)$ is evaluated using (12) and subsequently $u^o(t)$ is evaluated as

$$u^o(t) = F_U(w(t), s(t)) \tag{15}$$

where $F_U(w, s)$ is a look-up table where the offline-calculated results of (14) are stored. The advantage of this map-based implementation, used in S1, in terms of computing time with respect to the vectorial implementation, i.e., the online solution of (14) is evident. Moreover, the map-based implementation can be generated offline using a much finer discretization (i.e., a much higher N) than its vectorial counterpart. S1 uses a total of 7 maps (Sivertsson, 2012).

4.4. Implementation issues

Since CPU use and memory use are among the scoring metrics, all solutions pay a particular attention in trying to reduce them.

Concerning the heuristic solutions, implementation of rules reduces memory use, while the use of maps reduces CPU time consumption. Other techniques adopted include:

- simplification of look-up tables (S2),
- disabling of inactive Simulink subsystems, triggered subfunctions (S1, S2, S6),
- only native Simulink blocks without maps (S5),
- reduction of complexity for modes 2 and 3 (S9).

Concerning ECMS-based solutions, vector-based implementation reduces memory use while map-based implementation reduces CPU time consumption. Other techniques adopted include:

- iterative sparsification of maps (S1),
- use of Matlab S-functions (S3),
- setpoint candidate reduction (S3),
- use of vector merging instead of multi-dimensional matrices for setpoint candidates (S7).

Enforcing component limits is a delicate issue that has a strong impact on the solution validity (see Section 3.1). Two main approaches have been followed in the proposed solutions:

1. A priori limitation (S3, S6, S7), where map inputs or ECMS degrees of freedom are preliminarily saturated to corresponding limits: in this approach actual limits might take into account also limits induced by the concurrent operation of other components; for example, the engine torque could be constrained by

the generator limits at the corresponding speed and such a method often introduces iterations, algebraic loops that deserve a special treatment (S7).
2. A posteriori limitation (S2, S5), where map outputs or ECMS candidates are checked and possibly saturated to their respective limits.

A combined approach, with a priori and a posteriori check, is used in S1, S8, and S9.

5. Results

5.1. Proposed solutions

Table 8 summarizes the main scores obtained by running the proposed solutions.

A comparison between all valid solutions in terms of SOC trajectory for cycle "Arco–Merano" is shown in Fig. 9. The figure clearly shows that best performing strategies, i.e., S1 and S7 (see respective scores in Table 8), are able to drive the SOC toward the minimum value before the last downhill in a smooth fashion. In contrast, solutions S3 and S5 suffer from SOC oscillations around the best-performing trajectory, solutions S2 and S4 tend to discharge the battery too fast, while S6 and S8 are characterized by heavy SOC deviations with respect to the best-performing trajectory. In the following section, the best-performing strategy S1 is detailed and compared with the optimal solution of Problem 8 calculated using an offline PMP algorithm.

Two among the three best performing solutions (S1 and S3) have adopted ECMS, alongside with two other solutions (S7, which is among the best performing strategies in the fuel economy tests, and S8). It is apparent that the performance of ECMS strongly depends on the implementation of (12). All four solutions have adopted (13) with possible SOC margins. As a consequence, the $\xi_{ref}(t)$ signal is decreasing almost linearly with the increase of time and distance covered, as shown in Fig. 10.

Table 8 shows a weak correlation between fuel economy results and memory use, processing time results. Discrepancies are particularly visible for solutions S7 and S5. The latter is the simplest solution, so it performs rather well both in terms of CPU time and memory use. On the other hand, S7 has the second-best aggregate score in the fuel economy tests but it is penalized in terms of CPU time by the use of long ECMS-candidate vectors and in memory use by the use of large maps. Generally speaking, ECMS-based strategies require Hamiltonian minimization and thus need a larger CPU time than heuristic strategies; however S1 that is map-based rather than vector-based performs better than S3, S7, and S8 in terms of CPU time. As for memory use, there is no clear trend among ECMS and

Table 8
Benchmark results. Fuel A: cycle Arco–Merano; Fuel B: cycle Aachen.

Solution	Fuel A L/h km	Fuel B L/h km	Memory kB	Proc. time 100=Avg.	Score
S1	2.86	1.83	153	16	0.836
S2	3.18	2.27	81	14	0.850
S3	3.56	1.86	24	84	0.877
S4	3.39	2.57	159	18	0.934
S5	3.66[a,c]	4.60[a,c]	36	13	0.993
S6	3.66	4.60	186	18	1.086
S7	3.05	1.88	298	197	1.119
S8	6.06[a,c]	4.99[a,c]	353	442	1.873
S9[c]	NA[b]	8.93	95	361	NA

[a] Test disqualified due to constraints not met, results set to maximum valid if lower.
[b] Test invalid, cycle not entirely driven.
[c] Entry or test not considered for average calculation.
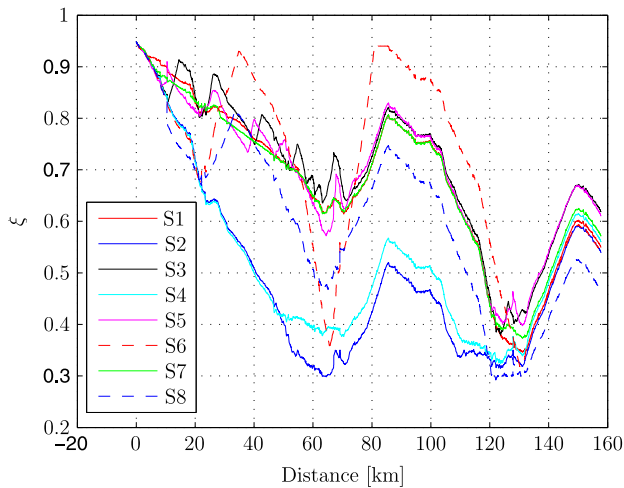
Fig. 9. Cycle Arco–Merano, SOC for all valid strategies.
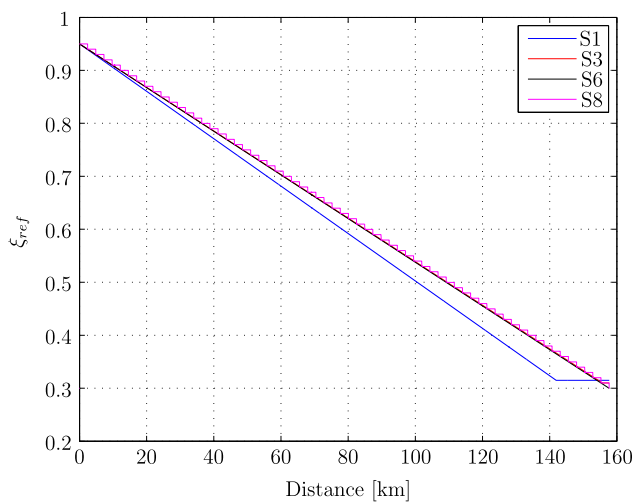
Fig. 10. Cycle Arco–Merano, reference SOC for all four ECMS strategies.

heuristic strategies, since both require maps and look-up tables either to represent the various components' efficiency, or to store the pre-calculated setpoints or rules.

The final scores shown in Table 8 were calculated using the European CO_2 emission factor, as explained in Section 3.1. If other CO_2 factors were used, the scores would change only slightly and the ranking of the proposed solutions would remain the same.

A different focus in weighting the scoring metrics would obviously lead to a different ranking. For instance, an enhanced focus on computational performance (raised from 20% to 40%) would let S2 have a better overall score than S1. On the other hand, an enhanced focus on energy efficiency (raised from 50% to 75%) would confirm S1 as the best solution. Since performance metrics are similar for all solutions, enhancing their weight (e.g., rasing it from 30% to 65%) would not change the ranking at all. However, setting different weights would have probably led to differently focused solutions, so that the previous discussion must be taken very cautiously.

5.2. Globally optimal solution

To assess the optimality of the best-performing strategy S1, it has been compared with the outcome of an offline optimization tool (HOT, Chasse & Sciarretta, 2011) that has prior knowledge of the driving cycle. HOT is based on PMP and finds iteratively the initial values of the costates. Model parameters are the same as in the Voltec simulator. The optimization criterion set is fuel consumption. Although having the capability of running cold-start cycles and performing two- or three-state optimization with engine and aftertreatment temperatures as additional state variables (Merz, Sciarretta, Dabadie, & Serrao, 2012), for this test the standard functionality with SOC as the single state has been used. The constant value of the equivalence factor s_0 is found with a root-finding algorithm based on the SOC at the end of the test cycle, $\xi(T)$. The target value for the SOC is set to $\xi_t = 30\%$: positive deviations $\xi(T) - \xi_t$ make s_0 decrease in the next iteration, while negative deviations make s_0 increase.

Fig. 11 compares the optimal profiles of SOC and fuel consumption calculated by HOT for test cycle "Aachen" with the outcome of S1. The two profiles are very close to each other. Wider differences are visible in Fig. 12 that refers to the test cycle "Arco–Merano". However, in this case too, S1 is able to make the SOC approach the minimum value (30%) before the final downhill, at the end of which both S1 and HOT predict a SOC of about 50%. Consequently, the fuel consumption predicted by S1 is very close to the minimum value calculated by HOT (2.76 l/h km for test cycle "Arco–Merano", 1.77 l/h km for test cycle "Aachen"), see Table 8.

Figs. 13 and 14 show the distribution of engine operating points during the two test cycles. Both S1 and HOT tend to operate the engine around the best efficiency region, although S1, being an online strategy, is characterized by many more distinct operating points (transient maneuvers, SOC regulation, etc.) than HOT. Figs. 15 and 16 compare the two distributions of powertrain
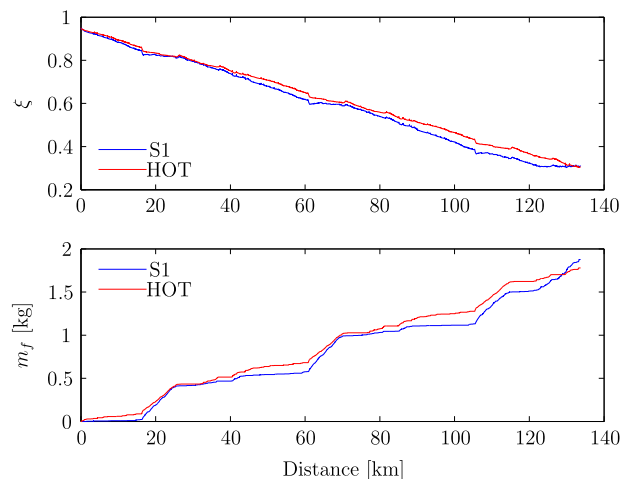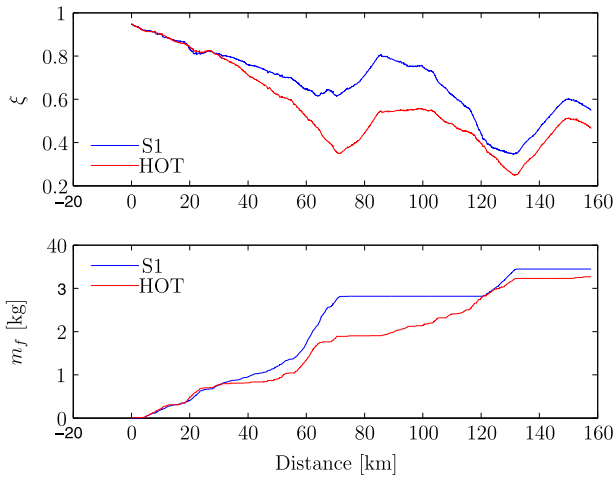

Fig. 11. Aachen cycle, SOC and fuel consumption.
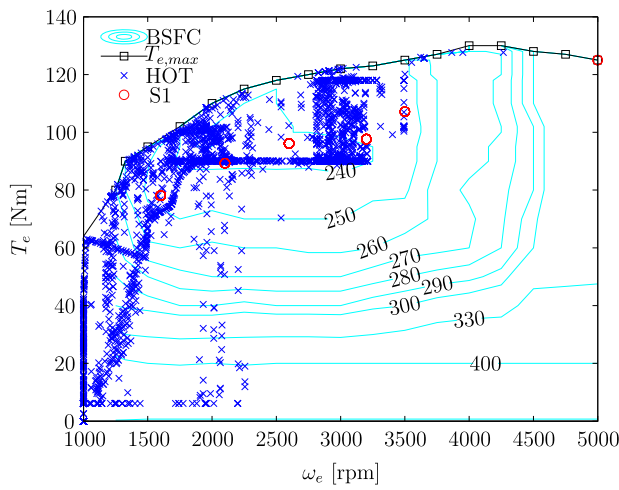
Fig. 12. Arco–Merano cycle, SOC and fuel consumption.
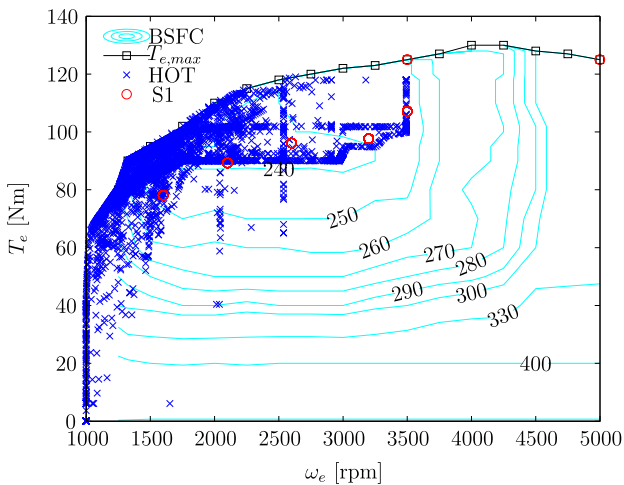

Fig. 13. Aachen cycle, engine operating points.

Fig. 14. Arco–Merano cycle, engine operating points.

Fig. 15. Aachen cycle, powertrain modes, HOT (top) and S1.

Fig. 16. Arco–Merano cycle, powertrain modes, HOT (top) and S1.

6. Conclusions

The development of benchmark control problems is an engineering practice that can help assessing different methods and techniques under the same circumstances (same system, same operating conditions, same exogenous inputs, etc.). The benchmark control problem illustrated in the paper, concerning the energy management of a plug-in hybrid-electric car, has fulfilled this goal since it has allowed:

1. a comparison in terms of various performance criteria: while the dispersion among different solutions in terms of vehicle performance is rather low, fuel economy is more sensitive to the energy management strategy adopted, particularly for real-life driving profiles with severe altitude variations; even larger differences can arise in terms CPU time and memory use;

2. a comparison between heuristic and optimal control-based techniques: generally speaking the latter outperform the rule-based approach under the same unprevisible circumstances, particularly in terms of fuel economy; however, heuristic algorithms can also achieve good performance, yet they require a higher amount of tuning and they are less robust to system variations;

3. a comparison between different implementations of optimal-control-based techniques (ECMS): map-based ECMS induce

modes as a function of vehicle speed and torque. In both strategies, mode 1 is selected at low speeds, mode 2 for medium speeds and low torques, mode 3 at high torques, and mode 4 for medium to high speeds. The S1 use of the mode 3 is wider than in HOT, certainly in the attempt of perform a blended-mode strategy as a function of SOC. The behavior in the two cycles is similar for both strategies, except for the different speed and torque range.
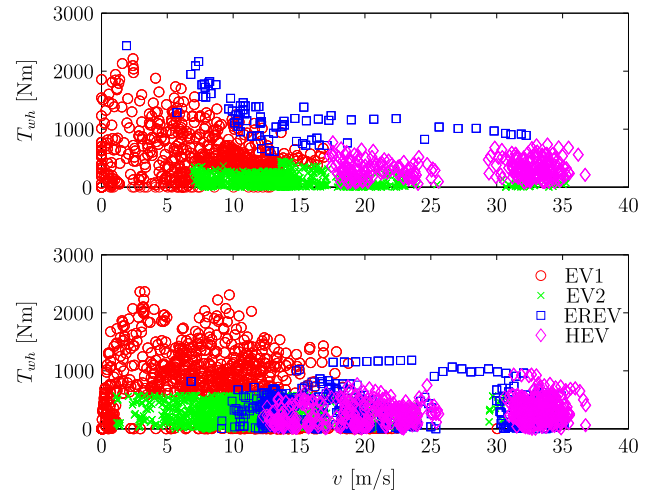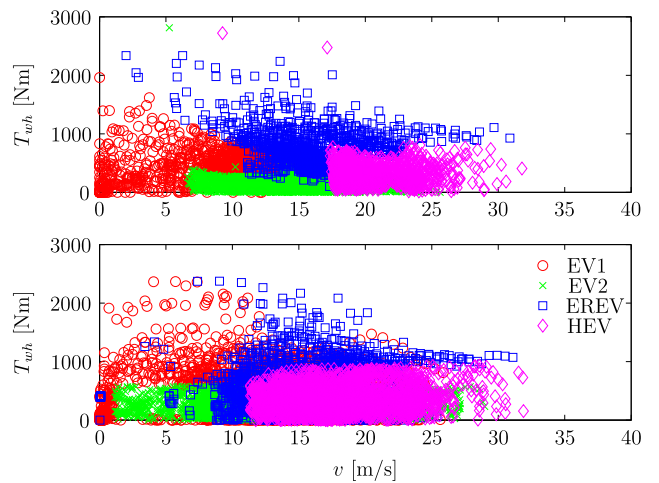
lower computing efforts but higher memory use, while the opposite is true for vector-based ECMS;

4. a comparison between online solutions and a globally optimal solution calculated offline with a prior knowledge of the driving profile: if properly designed, online energy managers can be very close to global optimum in terms of fuel economy.

The software developed for this benchmark will be made available on the web site www.ecosm12.org for further use by control engineering students and professionals that desire to test their own energy management solutions. A parametric PMP algorithm, HOT, will be also accessible to calculate a global optimum reference. Such platform could be enriched in the future with generators of random but realistic driving profiles and the possibility of modifying the structure and the components of the powertrain in the online-running model representing the actual system.

Acknowledgements

References

Ambühl, D., Sciarretta, A., Onder, C., Guzzella, L., Sterzing, S., Mann, K., et al. (2007). A causal operation strategy for hybrid electric vehicles based on optimal control theory. In *Proceedings of the 4th Braunschweig symposium on hybrid vehicles and energy management.*

Anatone, M., Cipollone, R., Donati, A., & Sciarretta, A. (2005). *Control-oriented modeling and fuel optimal control of a series hybrid bus.* SAE Paper 2005-01-1163.

Baumann, B., Rizzoni, G., & Washington, G. (1998). *Intelligent control of hybrid vehicles using neural networks and fuzzy logic.* SAE Paper 981061.

Borhan, H. A., & Vahidi, A. (2010). Model predictive control of a power-split hybrid electric vehicle with combined battery and ultracapacitor energy storage. In *Proceedings of the American control conference* (pp. 5031–5036).

Brahma, A., Guezennec, Y., & Rizzoni, G. (2000). Optimal energy management in series hybrid electric vehicles. In *Proceedings of the American control conference* (pp. 60–64).

Chasse, A., & Sciarretta, A. (2011). Supervisory control of hybrid powertrains: *An experimental benchmark of offline optimization and online energy management. Control Engineering Practice, 19*(11), 1253–1265.

Cipollone, R., & Sciarretta, A. (2006). Analysis of the potential performance of a combined hybrid vehicle with optimal supervisory control. In *Proceedings of the IEEE international conference on control applications.*

Do, D. V. (2010). *Diagnostic de batteries lithium-ion dans des applications embarquées* (Ph.D. thesis). Université de Technologie de Compiègne.

Falières, Q., Grasset, O., Roblet, K., Xu, Y., Noiret, C., Serrao, L., et al. (2011). A contradictory analysis of GM voltec powertrain. In *Proceedings of the European electric vehicle congress.*

Grebe, U. D., & Nite, L. T. (2011). VOLTEC – the propulsion system for chevrolet volt and opel ampera. *ATZ Autotechnology, 5,* 4.

Guzzella, L., & Sciarretta, A. (2013). *Vehicle propulsion systems, introduction to modeling and optimization.* Springer pp. 13–45.

Hofman, T., Steinbuch, M., Serrarens, A., & van Druten, R. (2008). Rule-based equivalent consumption minimization strategies for hybrid vehicles. In *Proceedings of the 17th IFAC World Congress.*

Hofman, T., Ebbesen, S., & Guzzella, L. (2012). Topology optimization for hybrid electric vehicles with automated transmissions. *IEEE Transactions on Vehicular Technology, 61*(6), 2442–2451.

International Energy Agency report (2011). CO$_2$ emissions from fuel combustion: highlights.

Kim, N. W., Cha, S. K., & Peng, H. (2011). Optimal control of hybrid electric vehicles based on Pontryagin's minimum principle. *IEEE Transactions On Control Systems Technology, 19*(5), 1279–1287.

Kleimaier, A., & Schroder, D. (2002). An approach for the online optimized control of a hybrid powertrain. In *Proceedings of the 7th international workshop on advanced motion control.*

Koot, M., Kessels, J., de Jager, B., Heemels, W., van den Bosch, P., & Steinbuch, M. (2005). Energy management strategies for vehicular electric power systems. *IEEE Transactions on Vehicle Technology, 54*(3), 1504–1509.

Larsson V., Johannesson, L., & Egardt, B. (2010). Impact of trip length uncertainty on optimal discharging strategies for PHEVs. In *Proceedings of the IFAC symposium on advances in automotive control.*

Lin, C. C., Kang, J. M., Grizzle, J. W., & Peng, H. (2001). Energy management strategy for a parallel hybrid electric truck. In *Proceedings of the American control conference* (pp. 2878–2883).

Liu, J. M., & Peng, H. (2006). Control optimization for a power-split hybrid vehicle. In *Proceedings of the American control conference.*

Marcos, D., & Bordons, C. (2012). Power management of a plug-in hybrid electric vehicle based on cycle energy estimation. In *Proceedings of the 2012 IFAC workshop on engine and powertrain control, simulation, and modeling.*

Merz, F., Sciarretta, A., Dabadie, J.-C., & Serrao, L. (2012). On the optimal thermal management of hybrid-electric vehicles with heat recovery systems. *Oil and Gas Science and Technology, 67*(4), 610–612.

Miller, M. A., Holmes, A. G., Conlon, B. M., & Savagian, P. J. (2011). *The GM voltec 4ET50 multi-mode electric transaxle.* SAE Paper 2011-01-0887.

Musardo, C., & Rizzoni, G. (2005). A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management. In *Proceedings of the IEEE conference on decision and control and the European control conference* (pp. 1816–1823).

O'Kneefe, M., & Markel, T. (2006). Dynamic programming applied to investigate energy management strategies for a plug-in HEV. In *Proceedings of the 22nd international battery, hybrid and fuel cell electric vehicle symposium and exposition.*

Onori, S., Serrao, L., & Rizzoni, G. (2010). Adaptive equivalent consumption minimization strategy for HEVs. In *Proceedings of the ASME dynamic systems and control conference.*

Paganelli, G., Guerra, T. M., Delprat, S., Santin, J. J., Combes, E., & Delhom, M. (2000). Simulation and assessment of power control strategies for a parallel hybrid car. *Journal of Automobile Engineering, 214,* 705–718.

Parrish, R., Elankumaran, K., Gandhi, M., Nance, B., et al. (2011). *Voltec battery design and manufacturing.* SAE Paper 2011-01-1360.

Pisu, P., & Rizzoni, G. (2005). A supervisory control strategy for series hybrid electric vehicles with two energy storage systems. In *Proceedings of the IEEE conference on vehicle power and propulsion.*

Pisu, P., & Rizzoni, G. (2007). A comparative study of supervisory control strategies for hybrid electric vehicles. *IEEE Transactions On Control Systems Technology, 15*(3), 506–518.

Salman, M., Schouten, N. J., & Kheir, N. A. (2000). Control strategies for parallel hybrid vehicles. In *Proceedings of the American control conference* (pp. 524–528).

Sciarretta, A., & Guzzella, L. (2007). Control of hybrid electric vehicles. *IEEE Control Systems Magazine, 27*(2), 60–70.

Sciarretta, A., Back, M., & Guzzella, L. (2004). Optimal control of parallel hybrid electric vehicles. *IEEE Transactions on Control Systems Technology, 12*(3), 352–363.

Serrao, L., Onori, S., & Rizzoni, G. (2009). ECMS as a realization of Pontryagin's minimum principle for HEV control. In *Proceedings of the American control conference* (pp. 3964–3969).

Serrao, L., Sciarretta, A., Grondin, O., Chasse, A., Creff, Y., & diDomenico, D. (2013). Open issues in supervisory control of hybrid electric vehicles: A unified approach using optimal control methods. *Oil and Gas Science and Technology, 68*(1), 23–33.

Sivertsson, M. (2012). Adaptive control using map-based ECMS for a PHEV. In *Proceedings of the 2012 IFAC workshop on engine and powertrain control, simulation, and modeling.*

Sivertsson, M., Sundström, C., & Eriksson, L. (2011). Adaptive control of a hybrid powertrain with map-based ECMS. In *Proceedings of the IFAC World Congress.*

Spencer, B. F., Dyke, S. J., & Deoskar, H. S. (1998). Benchmark problems in structural control. Part I. Active mass driver system. *Earthquake Engineering and Structural Dynamics, 27*(11), 1127–1139.

Stockar, S., Marano, V., & Canova, M. (2011). Energy-optimal control of plug-in hybrid electric vehicles for real-world driving cycles. *IEEE Transactions On Vehicular Technology, 60*(7), 2949–2962.

TNO Report (2006), Review and analysis of the reduction potential and costs of technological and other measures to reduce CO$_2$ emissions from passenger cars.

Tulpule, P., Marano, V., & Rizzoni, G. (2009). Effects of different PHEV control strategies on vehicle performance. In *Proceedings of the American control conference.*

van Berkel, K., Hofman, T., Vroemen, B., & Steinbuch, M. (2012). Optimal control of a mechanical-hybrid powertrain. *IEEE Transactions on Vehicular Technology, 61*(2), 485–497.

Yasui, Y. (2012). JSAE-SICE benchmark problem 2: Fuel consumption optimization of commuter vehicle using hybrid powertrain. In *Proceedings of World Congress on intelligent control and automation* (pp. 606–611).