Commuter Route Optimized Energy Management of Hybrid Electric Vehicles

Viktor Larsson, Lars Johannesson Mårdh, *Member, IEEE*, Bo Egardt, *Fellow, IEEE*, and Sten Karlsson

***Abstract*—Optimal energy management of hybrid electric vehicles requires *a priori* information regarding future driving conditions; the acquisition and processing of this information is nevertheless often neglected in academic research. This paper introduces a commuter route optimized energy management system, where the bulk of the computations are performed on a server. The idea is to identify commuter routes from historical driving data, using hierarchical agglomerative clustering, and then precompute an optimal solution to the energy management control problem with dynamic programming; the obtained solution can then be transmitted to the vehicle in the form of a lookup table. To investigate the potential of such a system, a simulation study is performed using a detailed vehicle model implemented in the Autonomie simulation environment for MATLAB/Simulink. The simulation results for a plug-in hybrid electric vehicle indicate that the average fuel consumption along the commuter route(s) can be reduced by 4%–9% and battery usage by 10%–15%.**

***Index Terms*—Clustering algorithms, data mining, dynamic programming, energy management, hybrid electric vehicles, intelligent vehicles.**

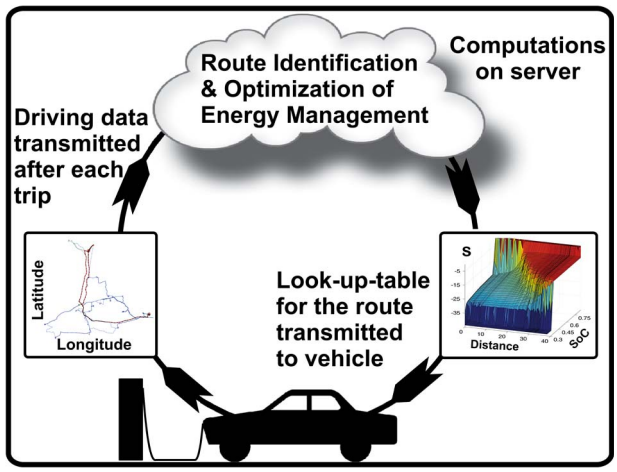


Fig. 1. Route optimized EMS. Driving data are transmitted to a server, where routes are identified and an optimal control strategy is precomputed.

I. Introduction

RISING fuel prices and pollutant emissions and an increasing concern for global warming has initiated a transition toward electric powertrains. One of the main challenges with a hybrid electric powertrain is the design of the energy management system (EMS), which controls the power split between the battery and the engine throughout a driving mission. Consequently, over the past decade, there has been a vast number of studies investigating different EMSs, for both hybrid electric vehicles (HEVs) and plug-in HEVs (PHEVs). The established research consensus is that a near-optimal fuel economy can only be reached if the future driving conditions are known *a priori* [1]–[3]. The most straightforward way to obtain the desired *a priori* information is to connect a GPS-based navigation system, containing speed limits and topography data, to the EMS and let the driver specify the final destination; such a system has been investigated in several studies [4]–[6] and is already available on the consumer market [7]. There are, of course, limitations with this type of system; the driver might not enter a route during everyday usage, and the predicted driving path/conditions might not reflect the actual drive. Furthermore, to solve the energy management control problem onboard the vehicle, the computational demand must be kept within reasonable limits, thereby drastically limiting the possible complexity of the vehicle model.

However, a recent development trend within the automotive industry is to connect the vehicle to the cellular network and the Internet [8], [9]. These systems enable a user to control certain features using his or her cell phone, e.g., check the battery charge level or start preheating/precooling. It is also possible for a user to download detailed driving statistics since trips are logged and the data are transmitted to servers [8]. The introduction of such systems opens up for new possibilities in terms of energy management. If driving data are available on a server, it is possible to use machine-learning algorithms to identify commuter routes and to precompute an optimal control strategy that can be transmitted back to the vehicle, as illustrated in Fig. 1. During operation, either the vehicle can then try to recognize routes autonomously, or the driver can be asked to confirm if any of the frequent routes will be driven.

The two principal building blocks of such a system, i.e., route identification and optimization of the EMS, have mainly been studied separately. Several studies have investigated different machine-learning techniques that can be used to identify routes

Manuscript received April 5, 2013; revised July 16, 2013; accepted November 26, 2013. Date of publication January 22, 2014; date of current version May 30, 2014. The Associate Editor for this paper was A. Amditis.

V. Larsson and B. Egardt are with the Department of Signals and Systems, Chalmers University of Technology, 412 96 Göteborg, Sweden (e-mail: lviktor@chalmers.se; egardt@chalmers.se)

L. Johannesson Mårdh is with the Viktoria Swedish ICT, 417 56 Göteborg, Sweden (e-mail: lars.johannesson@viktoria.se).

S. Karlsson is with the Department of Energy and Environment, Chalmers University of Technology, 412 96 Göteborg, Sweden (e-mail: sten.karlsson@chalmers.se).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2013.2294723

from historical driving data and how to recognize these routes when the vehicle is being operated [10]–[14]. Other studies have examined how to optimize the EMS toward known routes, where the driving conditions are modeled based on historical driving data [4], [14]–[18]. There is, as far as the authors know, only one study that combines route identification with optimization of the EMS [14]; however, in [14], the focus is on reinforcement learning, and a highly simplified vehicle model is considered in a simulation study that is presented only briefly.

The main contributions of this paper are therefore as follows: 1) a framework to combine route identification with optimization of the EMS; 2) a simulation study based on two different logged commuter driving patterns, using a detailed model of a parallel PHEV powertrain; and 3) a comparative study of a discharge strategy computed with dynamic programming (DP) [19] and two simpler discharge strategies (depletion–sustenance and linear discharge). Note that the main objective of the paper is not further development of the methods, but rather to obtain a realistic assessment of the overall benefit of such a system.

The methodology used in the paper is to identify commuter routes from logged GPS driving data, using hierarchical agglomerative clustering, which is an unsupervised learning technique, where the number of clusters, i.e., routes, are not specified in advance; the method has previously been used for route clustering with promising results in [11]. Once the commuter routes are identified, a route representation is derived, which is aimed to model both the driving conditions along the route and the uncertainty of the exact final position. Then, based on a simplified quasi-static vehicle model, DP is used to precompute an (implicit) optimal state-feedback law along the route representation. The simulation study is performed with the high-fidelity vehicle modeling software Autonomie [20], in which an equivalent consumption minimization strategy (ECMS) [21], [22] is implemented as the real-time control strategy. The equivalence factor is then given by either the precomputed DP state-feedback law or a proportional–integral (PI) controller tracking a linearly decreasing state-of-charge (SoC) reference.

Outline: The paper is divided into eight sections. Following the introduction, the route identification procedure and the corresponding route representation are explained. The next section describes how DP can be used to precompute an optimal control strategy for a commuter route. The simplified vehicle model used during the precomputations and the more detailed simulation model with the associated real-time control strategy are described in the two subsequent sections. In the second half of the paper, the simulation study and its results are presented before the paper ends with a discussion and conclusion. The database containing the logged driving data used in the simulation study is described in Appendix A.

II. Identification of Commuter Routes

In this paper, the concept of a commuter route is defined as a recurring driving trajectory between two different geographical areas, which is roughly along the same driving path. The definition is intentionally somewhat vague, in order to cover the variability of everyday driving. The idea is to keep the number of routes to a bare minimum and not form new routes for situations where the vehicle is parked in slightly different positions, slightly different paths around an obstruction are equiprobable, or when the driver stops for errands near the final destination, e.g., to shop at the local supermarket.

Trip Definition: A trip is defined as the drive between two consecutive parking periods irrespective of the duration of the parking periods.[1]

Trip Features: To facilitate the clustering, each trip is associated with a finite number of features, e.g., driving length, starting time, weekday, and GPS coordinates at different positions.

A. Trip Clustering Procedure

The process of clustering can be thought of as the task of grouping objects (trips) in an n-dimensional space, according to some measure of similarity. The idea is to assign nearby objects (trips) to an aggregated data structure denoted as a cluster (route), containing objects (trips) with a high degree of similarity.

Assuming that the driving pattern of a vehicle has been logged over the course of a few weeks, routes can be identified using the trip clustering procedure outlined in the following.

1) Assign each of the k logged trips with q features and form a trip data observation matrix $\Theta \in \mathbb{R}^{k\times q}$. The row vectors $r_i \in \mathbb{R}^q$, $i = 1, \ldots, k$, correspond to feature vectors of the logged trips, and the column vectors $c_j \in \mathbb{R}^k$, $j = 1, \ldots, q$, consequently represent all observed values of a certain feature.
2) Calculate the symmetric trip distance matrix $D \in \mathbb{R}^{k\times k}$, between all the k trips using the standardized Euclidian distance

$$d_{ij} = \sqrt{(r_i - r_j)V^{-1}(r_i - r_j)^T}, \qquad \{i, j\} = 1, \ldots, k. \quad (1)$$

   The diagonal matrix $V \in \mathbb{R}^{q\times q}$ is defined by the variances of the column vectors of Θ, i.e., $v_{jj} = \mathrm{var}[c_j]$.
3) Perform hierarchical agglomerative clustering and form a dendrogram tree. That is, start by defining each trip as an individual cluster; nearby clusters are then recursively merged into larger clusters until there is only one cluster remaining. For a more comprehensive description of hierarchical clustering, refer to any standard textbook covering unsupervised learning, such as [23].
4) Commuter routes are defined by clusters containing at least γ trips at a certain cutoff height h_c in the dendrogram tree.

Note that the cutoff height is a tuning parameter that must be determined iteratively. Parts 3) and 4) of the outlined procedure have previously been implemented in [11]. However, in that paper, the trip distance matrix was determined using another approach; rather than working with different features, the average pointwise spatial distance between the trips was calculated, which is a much more tedious approach.

[1] This definition was used in the database that forms the basis for this paper.

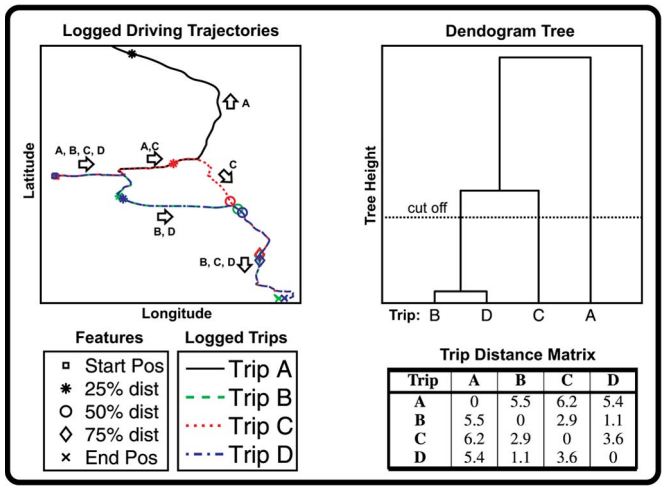


Fig. 2. Example of the trip clustering procedure consisting of four trips, which are characterized by five GPS positions at different fractional driving distances along the trip. The trip distance matrix shows the feature distance between the trips, and the dendrogram illustrates the resulting clustering steps.

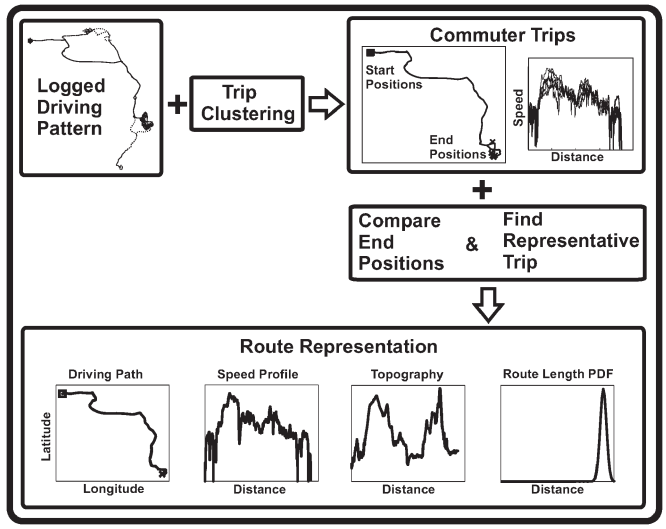


Fig. 3. Illustration of the workflow used to derive a route representation.

Observe that the trip definition used in this paper implies that some journeys might be divided into several shorter trips, separated by relatively short parking durations. Therefore, not all commuter journeys will be identified correctly with the trip definition used. In an actual implementation of the system, it is important to preprocess the driving data, in order to merge trips that probably belong to the same journey. It is, however, deemed to be beyond the scope of this paper to also investigate a more exhaustive trip definition.

Trip Clustering Example: To illustrate the trip clustering procedure, a small example consisting of four trips $\{A, B, C, D\}$ is shown in Fig. 2. Three of the trips, i.e., $\{B, C, D\}$, have the same starting and ending positions but follow two different paths; the fourth trip $\{A\}$ has the same starting position but has a different ending position. The trip features considered in the example are the starting and ending positions, as well as the positions where 25%–50%–75% of the trip length is driven, i.e., a total of ten features since the positions are specified with longitude and latitude. The resulting clustering dendrogram first merges the two trips that follow the same path $\{B, D\}$ into one cluster. In the next step, the third trip $\{C\}$, which has the same starting–ending positions, but along the other path, is merged with the first cluster. The final step merges the remaining trip $\{A\}$ with the first three. Finally, with the cutoff height indicated in the figure, there is only one cluster $\{B, D\}$ containing more than one trip.

B. Route Representation

In order to optimize the energy management of a hybridized vehicle for a particular route, the driving conditions along the route must be modeled using a suitable method. The authors have previously investigated a stochastic approach in [15] and [16], where a Markov model representation was used to precompute near-optimal control strategies for an HEV driving along a prescribed route. In those papers, it was also shown that a single previous trip logged along the route could be used with only slight degradation in fuel economy, at least if the driving conditions are recurring. The observation that commuter trips typically start within a narrow time interval, on an almost daily basis, means that the driving conditions are often relatively similar from day to day. Hence, in this paper, the driving conditions will be modeled by the most *representative trip* logged along the commuter route. Described very briefly, the idea is to assign each logged commuter trip with an additional feature vector (here characterizing the driving conditions rather than the driving trajectory) and then select the trip with the most representative feature vector; the procedure is outlined more in depth in Appendix A. The representative trip will define the nominal driving path (GPS trajectory), topography, and velocity trajectory of the route representation.

However, since parking locations and driving paths often vary slightly from day to day, all trips associated with a commuter route might not end in exactly the same position or after exactly the same driving distance. It is important to capture this uncertainty when optimizing the EMS, since the remaining driving distance is a key parameter when planning the battery usage of a hybridized vehicle, particularly with a PHEV, for which an overestimation of the remaining driving distance can be very costly [24]. To model the uncertainty, it is assumed that the route length is a normally distributed stochastic variable

$$z_f \in \mathcal{N}\left(z_n, \sigma_d^2\right) \tag{2}$$

where z_n is the mean trip length of the trips associated with the route, and σ_d^2 represents the corresponding variance. A graphic illustration of the workflow used to derive a route representation is shown in Fig. 3.

III. Precomputing an Optimal Control Strategy for the Route

The energy management control problem is to minimize the overall expected energy cost along the route representation, while adhering to the constraints and dynamics of a simplified

vehicle model with SoC as the only dynamic state x. The resulting optimal control problem can be formulated as

$$\begin{aligned}\min_{u(\cdot)}\ \mathbb{E}_{t_f}\left\{G(x(t_f))+\int_{t_0}^{t_f} g(u(t))\,dt\right\},\\ \text{s.t.}\quad \dot{x}(t)=f\left(x(t),u(t)\right)\\ T_p\big(u(t)\big)=T_r\left(v(t),\theta(t)\right)\\ x(t_0)=x_0\\ x(t_f)\in[x_f^{\min},x_f^{\max}]\\ x(t)\in[x_{\min},x_{\max}]\\ u(t)\in U\left(x(t),v(t),\theta(t)\right)\end{aligned}\tag{3}$$

where $f(x,u)$ represents the battery SoC dynamics, g represents the instantaneous fuel cost of the engine, and G represents the cost to recharge the battery. The decision variables in the control signal u vary with powertrain configuration, but it may include torque/power split(s), engine state, and choice of gear. Furthermore, the control signal must be chosen, such that the powertrain traction torque $T_p(u)$ meets the torque request of the route representation $T_r(v,\theta)$, where v and θ represents the route speed and road slope trajectories, respectively. Observe that the final time t_f is a stochastic variable, since the route length is assumed to be nondeterministic.

A. Numerical Solution With DP

The optimal control problem, which is illustrated by (3), is solved numerically using Bellman's principle of optimality and DP [19]. However, since the computed solution will be used as an (implicit) state feedback along the route, it is more convenient to define the cost-to-go over the state x and distance z along the route, rather than time. Described very briefly, the methodology is to time discretize the problem using the Euler method, with sample time t_s, grid the state and then solve the recursive equation

$$\begin{aligned}J(z_k,x_k)=\min_{u_k\in U_k}\ \{t_s g(u_k)+(1-p(z_{k+1}|z_k))\,G(x_k)\\ +p(z_{k+1}|z_k)J(z_{k+1},x_{k+1})\}\end{aligned}\tag{4}$$

backward in time, starting from the final sample N at driving distance z_N, where the cost-to-go is initialized with the final cost $J(z_N,x_N)=G(x_N)$. Note that the expected value operation in (3) is equivalent to discounting the cost-to-go, at each sample k, with the conditional probability $p(z_{k+1}|z_k)$ that the trip reaches the next position z_{k+1} given that position z_k has been reached, where the conditional probability is derived from (2). To improve numerical stability and reduce interpolation effects when enforcing the state constraints in (3), the DP algorithm is implemented using level-set functions to distinguish between backward-reachable and nonbackward-reachable grid points [25].

Extending the Route Representation: The assumption that the route end distance z_f is a stochastic variable implies that the length of the route representation will not be identical to the length of the representative trip. Hence, the driving conditions of the representative trip must be extrapolated to cover driving distances up to the length of the route representation, which is defined here as $z_N=z_n+4\sigma_d$. For simplification reasons, the speed profile is extended back to back with its initial part, and the topography is (during the extended part) assumed to be flat. Observe that the extrapolated driving conditions are of minor importance, since this part will have a significant discount when solving the energy management control problem.

TABLE I
VEHICLE DATA

Chassis		
Mass / Air res.	m, C_dA	1720 kg, 0.675
Wheel radius / inertia	r_w, J_w	0.3 m, 4 kgm^2
Gearbox ratios	$r_{gb,i}$	$2.6, 1.6, 1.0, 0.7, 0.5$
Gearbox efficiency	$\eta_{gb,i}$	$0.88+0.02i$
Final gear ratio / efficiency	r_f, η_f	4.4, 0.97
Aux. load.	P_a	200 W
Battery		**Li-Ion**
Cell volt./cap./res.	V, Q, R	3.6 V, 37 Ah, 2.1 $m\Omega$
Number of series cells	n_c	45 (6 kWh)
Depth of discharge	DoD	55% (3.3 kWh)
Engine		**4 Cyl. Spark Ignited**
Max power	$P_{e,max}$	65 kW@520 rad/s
Max torque	$T_{e,max}$	137 Nm@393 rad/s
Electric Motor		**Permanent Magnet**
Max power	$P_{m,max}$	60 kW
Max torque	$T_{m,max}$	458 Nm@[0, 130] rad/s

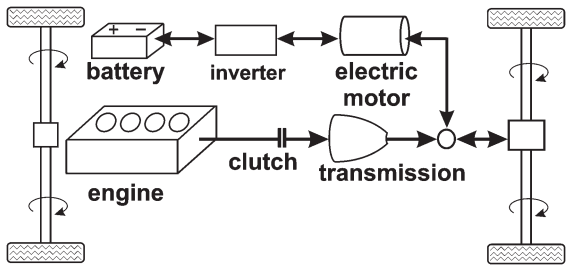


Fig. 4. Vehicle configuration.

IV. VEHICLE MODELING

A post-transmission parallel PHEV is considered, i.e., the electric motor is connected directly to the front axle, and the engine is coupled through a clutch and a stepped automatic transmission. The vehicle configuration is one of the preparameterized models available in the Autonomie software [20]; the only modification made with respect to the original powertrain model is the number of cells in the battery pack, which has been altered to give an all-electric range comparable to that of the Toyota Plug-in Prius. The powertrain parameters are summarized in Table I, and the configuration is shown in Fig. 4.

A. Detailed Simulation Model in Autonomie

To obtain a realistic assessment of the potential of a commuter route optimized EMS, the simulation study is carried out in the Autonomie software, which is a high-fidelity vehicle modeling environment for MATLAB/Simulink developed by

Argonne National Laboratory. In an Autonomie vehicle model, all principal powertrain components are described by a plant model and a separate controller. The model is based on a causal forward-looking modeling approach, where a drive cycle is used as input to a driver model. The driver's traction demand is then interpreted by the EMS, where appropriate torque requests for the engine and electric motor are decided. The full Autonomie model is nonlinear and contains several (> 5) dynamic states, meaning that the computational demand would be immense if the simulation model is used for optimization. Hence, a simplified model with SoC as the only dynamic state x is extracted from the Autonomie model. The simplified model is outlined next, and powertrain components not described are not considered when precomputing the EMS.

B. Simplified Vehicle Model for the Precomputations

In the simplified model, the forces acting on the powertrain are determined using a noncausal and inverse approach, e.g., see [26], meaning that the torque requested at the wheels, i.e., T_r, to follow a given velocity and road slope trajectory, is given by

$$T_r = r_w \left(1/2\rho C_d A v^2 + m_e a + mg(f_r \cos\theta + \sin\theta)\right) \quad (5)$$

where v represents the velocity, a is the acceleration, θ is the road slope, ρ is the density of air, $C_d A$ is the air drag resistance, f_r is the rolling resistance, m is the vehicle mass, and m_e is the equivalent vehicle mass, i.e., including moments of inertia of the rotating parts. The traction torque of the powertrain T_p, at the wheels, is given by

$$T_p = \eta_f r_f \left(T_m + \eta_{gb,i} r_{gb,i} T_e\right) + T_b \quad (6)$$

where T_m represents the motor torque, T_e represents the engine torque, and T_b represents the torque of the friction brakes. The ratio of the final gear is denoted by r_f, and the corresponding efficiency η_f depends on the sign of the torque demand at the wheels. The gears, $i = 1, \ldots, 5$ are represented by a drive ratio $r_{gb,i}$ and a mechanical efficiency $\eta_{gb,i}$. It is assumed that torque responses are instantaneous and that gear shifts and the engine-state transitions are both instantaneous and lossless. The instantaneous fuel mass rate of the engine is approximated to be affine in engine torque, and the instantaneous fuel cost g is thus given by

$$g = c_f \left(c_0(\omega_e) T_e + c_1(\omega_e)\right) e_{\text{on}} \quad (7)$$

where c_f represents the price of fuel, and e_{on} represents the engine state. Furthermore, the combined electrical power demand of the motor and the inverter is assumed to be quadratic in motor torque. The resulting battery power P_b is hence

$$P_b = d_0(\omega_m) T_m^2 + d_1(\omega_m) T_m + d_2(\omega_m) + P_a \quad (8)$$

where P_a represents the auxiliary electrical loads. The speed-dependent coefficients $c_{0:1}$ and $d_{0:2}$ are determined by linear least squares from the engine and motor maps available in Autonomie. A Li-ion battery is considered, and it is modeled as an equivalent circuit consisting of n_c battery cells connected in series. The battery is assumed to have a constant internal resistance R_i and an open-circuit voltage V_{oc} that is affine in x, i.e., in SoC. Consequently, the state dynamics are given by

$$\frac{dx}{dt} = f(x,u) = -\frac{V_{\text{oc}}(x) - \sqrt{V_{\text{oc}}^2(x) - 4P_b R_i}}{2R_i Q} \quad (9)$$

where Q represents the cell capacity.

To summarize, the control signal u for the simplified vehicle model is the engine-state decision and the torques of the engine and the electric motor, i.e.,

$$u = [e_{\text{on}}, T_e, T_m] \in U(\omega_e, \omega_m) \quad (10)$$

where U represents the torque constraints of the engine and motor. Moreover, the control signal must be chosen, such that the torque request is satisfied, i.e., T_p should be equal to T_r. The choice of gear is not included in the control signal of the simplified model, since it is decided by a separate gearbox controller in the Autonomie model. Hence, when precomputing the EMS with the simplified model, the gear selected is the highest possible gear that will not cause the engine to stall; an assumption that will be nearly fuel optimal since the engine and electric motor do not share transmission.

V. Real-Time Control Strategy in Autonomie

In the Autonomie vehicle model, the torque split controller is modified from a rule-based strategy to an ECMS strategy [21], [22]. The optimal control signal is determined using the simplified vehicle model and is thus given by

$$u^* = \arg\min_{u \in U} \{g(u) + s \cdot f(x,u) + \delta(e_{\text{on}})\} \quad (11)$$

where s represents the ECMS equivalence factor, and δ is a small cost for turning on the engine, which is added to decrease the number of engine-on decisions. Furthermore, since the simplified vehicle model is based on affine and quadratic expressions, the ECMS equation (11) is solved analytically to obtain a closed-form expression for the control signal. The expression is, however, rather complicated and will therefore not be shown here, although it is straightforward to derive using a symbolic solver. During the simulations, the torque requests are updated at 100 Hz, i.e., the sample time of the Autonomie model, and the engine-state request at 0.5 Hz. The value of the equivalence factor s is determined differently depending on the discharge strategy.

A. Route Optimized Discharge Strategy

With the route optimized strategy, the equivalence factor is determined by linear interpolation in a 2-D lookup table $h(x,z)$, with SoC and distance position along the route representation as the interpolation variables. The lookup table is

determined by computing the partial derivative (numerically) of the cost-to-go J, with respect to the state x, i.e.,

$$s = h(x,z) = \left.\frac{\partial J(x,z)}{\partial x}\right|_{\{x,z\}}. \tag{12}$$

The distance position z along the route representation can be determined as the distance traveled since the start of the trip, which is corrected with a distance offset against the route representation Δz, as the trip might not have begun at the exact nominal starting position. The offset can, for example, be determined by calculating the normalized 2-D cross-correlation between the GPS trajectories of the route representation and the ongoing trip [27].

B. Linear Discharge With Respect to Route Energy Demand

To assess the benefit of the route optimized discharge strategy, it is compared against a heuristic discharge strategy, which is implemented by a PI controller and a SoC reference that is decreased gradually along the route. If the topography is relatively flat, it has been shown that it is near optimal to decrease the reference linearly with distance; see [6] and [27]. However, if the route is hilly, it is important to account for the potential energy that can be recuperated. Therefore, the reference is decreased linearly with respect to route energy demand, rather than distance, thus ensuring an increasing reference during segments of negative traction request, e.g., during downhill driving. The SoC reference x_r is hence given by

$$x_r(z) = \begin{cases} x_0 - (x_0 - x_f)\frac{E(z)}{E(z_n - 3\sigma_d)}, & \text{if } z \leq z_n - 3\sigma_d \\ x_f, & \text{if } z > z_n - 3\sigma_d \end{cases} \tag{13}$$

where x_f represents the desired final SoC, and the cumulative route energy demand is given by $E(z(t)) = (1/r_w)\int_{t_o}^{t} v(\tau)T_r(\tau)d\tau$. Observe that the reference is defined such that the lower SoC limit is reached at a three-sigma distance before than the mean route length; this is to ensure that the battery will be fully depleted. Finally, the equivalence factor is given by

$$s = s_0 - F_{\mathrm{PI}}(e) \tag{14}$$

where s_0 corresponds to the nominal value of the equivalence factor. The PI controller is represented by F_{PI}, which is defined by a proportional gain K_p and an integral gain K_i. The tracking error e is defined as $e = x_r - x$.

C. CDCS Discharge Strategy

The trivial charge-depletion–charge-sustaining (CDCS) discharge strategy is also investigated as a baseline strategy, representing the case when there is no knowledge of the route. In the CDCS strategy, the equivalence factor is zero during the depletion phase, and in the sustaining phase, commencing below x_s, it is given by a PI controller

$$s = \begin{cases} 0, & \text{if } x > x_s \\ s_0 - F_{\mathrm{PI}}(e), & \text{if } x \leq x_s \end{cases} \tag{15}$$

where $e = x_f - x$. Furthermore, a hysteresis relay is used to ensure that no chattering effects occur around the switching level.

TABLE II
DATA AND PARAMETER VALUES USED IN THE SIMULATION STUDY

Route Clustering		
Trip features	7 lat/long positions & trip length (15 feat.)	
Cut-off height	h_c	1.5
Min. cluster size	γ	10
Training Period	**Pattern A**	**Pattern B**
Number of trips	103	89
Total driving [km]	2058	1977
Trips to & from work	23 & 22	13 & 11
Validation Period	**Pattern A**	**Pattern B**
Number of trips	121	163
Total driving [km]	2808	2178
Trips to & from work	26 & 21	14 & 13
EMS Parameters		
Electricity / Petrol cost	c_e, c_f	€ 0.12/ kWh, € 1.75 / L
Initial / Final SoC	x_0, x_f	0.85, 0.30
Route Optimized Strategy		
SoC grid	x	$[0.25 : 0.001 : 0.90]$
Final cost	$G(x)$	$\max\{5 - 6x, 20 - 56x\}$
Energy Reference & CDCS Strategy		
Parameters	K_p, K_i, s_o, x_s	150, 2.5, -30, 0.325

VI. SIMULATION STUDY

The simulation study in Autonomie is carried out using real-world driving patterns extracted from the Swedish Car Movement Database, i.e., a database containing GPS-logged driving data from roughly 500 passenger cars, gathered between 2010 and 2012 in the Göteborg area in western Sweden. More information regarding the database can be found in Appendix B. Two driving patterns are examined in the simulation study, each roughly two months long. The patterns were selected based on the following two criteria: 1) the vehicle should have a distinct commuter pattern with a route exceeding 30 km, and 2) the commuter route must be driven at least 50 times during the logging period.

A. Training and Validation Periods

The logged data for each driving pattern were divided into a training period and a validation period, which are both equal in duration. The training period was used to identify the commuter routes and to derive the corresponding route representations, as outlined in Section II. During the route clustering, each trip was associated with seven evenly distributed GPS positions (as in Fig. 2) and the trip length, i.e., in total 15 features. The validation period was reserved for the simulation study in Autonomie.

B. Investigated Driving Patterns

The two driving patterns are described briefly in the following, and the main driving statistics are summarized in Table II; the resulting route representations are shown in Figs. 5 and 6.

Pattern A has a commuter route going from the suburb of Kungsbacka to an industrial facility on the outskirts of

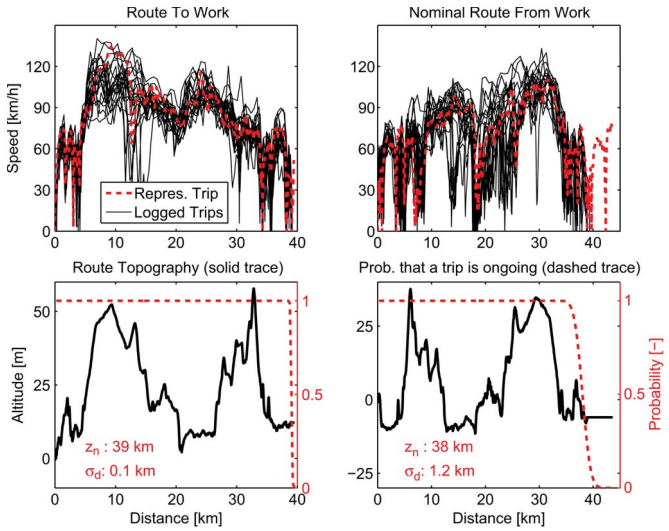


Fig. 5. Driving pattern A—Route representation. The upper plots show the representative trip and the logged commuter trips. The lower plots depict the route topography and the probability that the trip is ongoing.

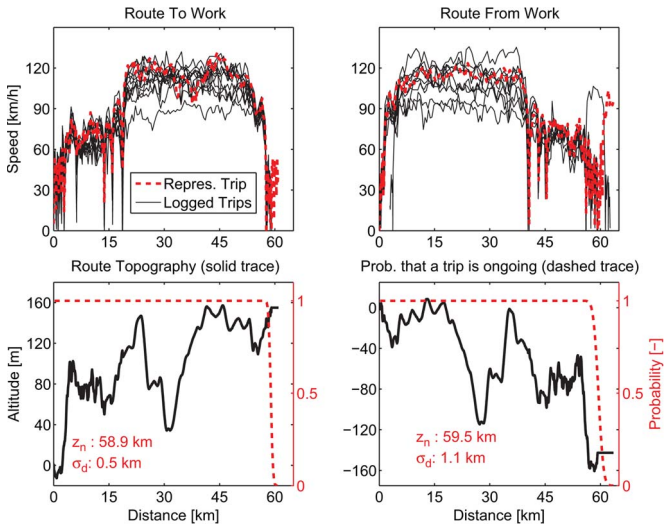


Fig. 6. Driving pattern B—Route representation. The upper plots show the speed profiles of the logged commuter trips. The lower plots depict the route topography and the probability that the trip is ongoing.

Göteborg, which is the region's main city. The route is mainly along the E6 motorway, which (by Swedish standards) has a fairly high volume of traffic.

Pattern B has a commuter route between Lerum, which is a suburb of Göteborg, and the medium-sized city of Borås. Two thirds of the route follows national road 40, which is a trunk road with a relatively low volume of traffic, and the remaining third follows a smaller country road.

C. Simulation Setup

The commuter trips found in the validation period were used as speed references when simulating the Autonomie vehicle model with the ECMS controller described in Section V; the route optimized discharge strategy, the linear discharge strategy, and the baseline CDCS discharge strategy were simulated. During the study, it was assumed that the vehicle was informed

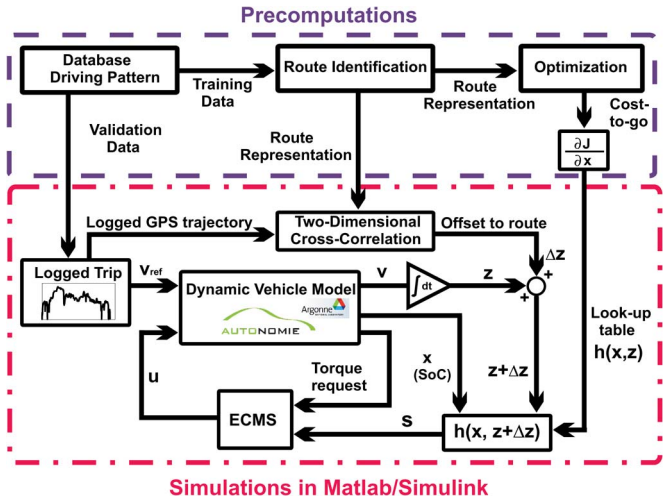


Fig. 7. Block diagram illustrating the workflow used during the simulation study with the route optimized discharge strategy.

of the choice of route before the start of each trip. The workflow during the simulation study is illustrated in Fig. 7, and the parameter values used are shown in Table II. To make the study more realistic, the PI controller, which is used by the linear discharge strategy and the CDCS strategy, had the same parameterization for both driving patterns. Furthermore, the route optimized cost-to-go was also initialized with the same final cost (G) for both driving patterns. The penalty for turning on the engine (δ) was the same for all three strategies. For simplification reasons, the route representations for each driving pattern were kept constant during the simulation study; however, it would be straightforward to update the route representations recursively after each additional commuter trip.

D. Simulation Results

The speed profiles and the corresponding SoC discharge trajectories for the commuter trips in the validation period are shown in Figs. 8 and 9. By studying the speed profiles, it is clear that the representative trips for both driving patterns are also good representations of the driving conditions during the validation period. The simulation results for the two driving patterns are summarized in Tables III and IV, which display the mean values of the fuel mass consumed, the C-rate, the ampere-hour throughput, the final SoC, and the standard deviation of the final SoC. As the difference in final SoC is relatively small, the fuel mass shown in the table has not been adjusted with respect to the difference in final SoC.[2]

From the simulations, it is clear that the route optimized strategy tends to end near the desired final SoC, with low variance, since the uncertainty in trip length is considered when the route optimized strategy is computed. Furthermore, also the linear discharge strategy tends to end near the desired final SoC, although, in this case, it is mainly due to the ad hoc definition of the SoC reference. The CDCS strategy has no route knowledge and is therefore more likely to end at an undesired SoC level, particularly if there is a downhill segment during the

[2]This was investigated by the authors, but the influence is relatively small.

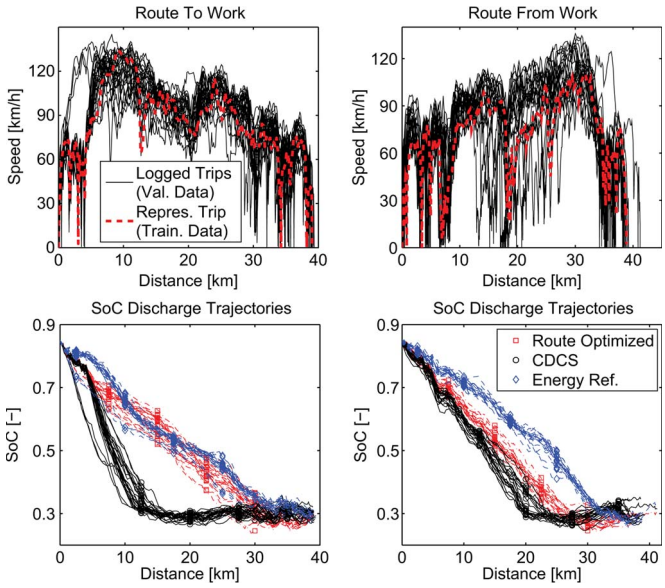


Fig. 8. Driving pattern A—Simulated SoC trajectories in Autonomie for the commuter trips in the validation period.

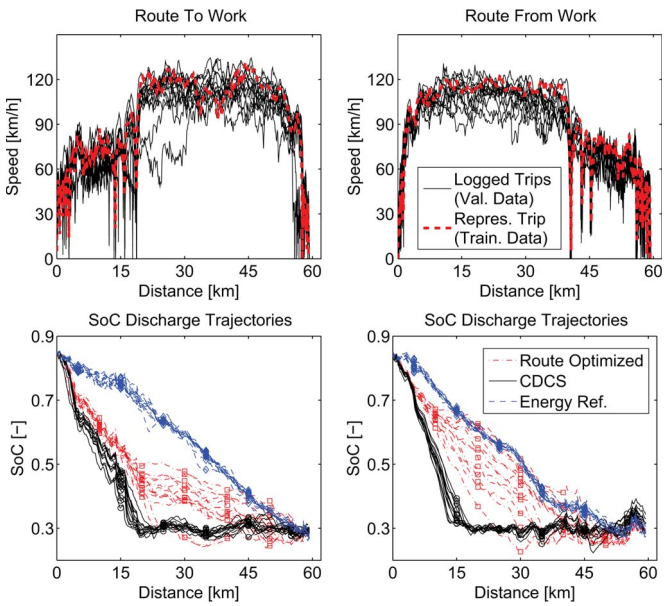


Fig. 9. Driving pattern B—Simulated SoC trajectories in Autonomie for the commuter trips in the validation period.

TABLE III
DRIVING PATTERN A—SIMULATION RESULTS. THE FIGURES IN PERCENT ARE WITH RESPECT TO THE OPTIMIZED STRATEGY

	CDCS		Energy Ref.		Optimized	
Mean	To	From	To	From	To	From
SoC_f	0.299	0.305	0.297	0.289	0.294	0.294
$Std(SoC_f)$	0.014	0.023	0.014	0.016	0.005	0.006
Fuel kg	0.824	0.613	0.748	0.597	0.731	0.592
	+12.8%	+3.5%	+2.4%	+1.0%	-	-
Ah through.	43.5	41.2	37.3	39.1	36.6	38.9
	+18.7%	+6.0%	+1.9%	+0.6%	-	-
C-rate	2.42	1.97	2.07	1.87	2.03	1.86
	+18.9%	+6.3%	+1.9%	+0.6%	-	-

TABLE IV
DRIVING PATTERN B—SIMULATION RESULTS. THE FIGURES IN PERCENT ARE WITH RESPECT TO THE OPTIMIZED STRATEGY

	CDCS		Energy Ref.		Optimized	
Mean	To	From	To	From	To	From
SoC_f	0.290	0.314	0.280	0.293	0.290	0.297
$Std(SoC_f)$	0.010	0.023	0.009	0.013	0.004	0.011
Fuel kg	1.70	1.39	1.69	1.30	1.67	1.31
	+1.9%	+6.3%	+1.4%	-0.7%	-	-
Ah through.	55.2	57.3	49.2	50.6	50.3	50.9
	+9.8.%	+12.6.%	-2.1%	-0.6%	-	-
C-rate	2.00	2.04	1.78	1.80	1.82	1.81
	+10.0%	+12.7%	-2.1%	-0.5%	-	-

very end of the route; this effect is clearly illustrated in driving pattern B for the route from work. Furthermore, since the linear discharge and the CDCS strategy does not explicitly consider the uncertainty in route length, the variance in final SoC is thus noticeably higher for these strategies.

In terms of fuel economy, it is evident that both the route optimized strategy and the linear discharge strategy can be significantly better than the trivial CDCS strategy. More specifically, the CDCS strategy is particularly inefficient if there is a segment of high power demand during the initial part of the route, i.e., high speed and/or uphill driving. Operation in the charge depletion mode during a high-power segment will result in high discharge currents and high resistive losses, since these are quadratic in current. Therefore, a CDCS strategy will have poor fuel economy if there is a high-power segment during the all-electric range of the PHEV. Consequently, as illustrated in Tables III and IV, the improvement in fuel economy is highly correlated with the direction of travel along the route.

Moreover, both the route optimized and the linear discharge strategy have additional advantages compared to the CDCS strategy. The two former strategies discharge the battery at a slower rate, i.e., at a lower C-rate. Thus, less time is spent in the charge sustaining mode, and the overall battery ampere-hour throughput is reduced as well. Both the C-rate and the ampere-hour throughput are correlated with the capacity fade of Li-ion batteries [28], meaning that the route optimized discharge strategy and the linear discharge strategy should improve battery life length.

VII. DISCUSSION

Three important aspects in this paper are identification and representation of routes and how to adapt the EMS toward the identified routes. These aspects should be investigated further and are therefore discussed briefly in the following.

A. *Route Identification Via Trip Clustering*

One aspect that has not been investigated in the paper is the reliability and robustness of the trip clustering procedure. Agglomerative clustering is an unsupervised learning procedure, meaning that there is no guarantee that the correct number of routes will be identified. If the cutoff height, in the dendrogram tree, is poorly chosen, it might very well happen that too many

(or too few) routes are formed, e.g., the daily commute might be divided into several closely related routes. To counteract such behavior, the route clustering method must be refined further. However, it is less likely that trips going between completely different locations are merged into a route, at least if the trip features are GPS coordinates and trip length.

B. Route Representation

In this paper, the driving conditions along the route are modeled by a *representative trip*, which is chosen from historical driving data to reflect typical driving conditions. This approach might not be very robust if the traffic conditions along the route change from day to day. One way to improve robustness could be to consider several *representative trips*, e.g., to cover both free-flowing and congested traffic flow. Another possibility would be to derive one or several stochastic representations, e.g., Markov models, of the route driving conditions.

C. Optimization of the EMS

The simulation results indicate that the route optimized strategy and the linear discharge strategy are essentially equally good, in terms of both fuel economy and battery usage. Hence, one might argue that it is not worth the computational effort (and cost) to compute a route optimized strategy on a server. Nevertheless, a (heuristic) linear discharge strategy is not guaranteed to give a near-optimal fuel economy for all possible vehicle configurations and routes.

VIII. Conclusion

This paper has investigated the concept of a commuter route optimized EMS, where the bulk of the computations are performed on a server. The simulation results for two real-world driving patterns, using a high-fidelity model of a post-transmission parallel PHEV, indicate that fuel consumption along the two commuter routes (A and B) can be reduced by an average of 9% and 4%, respectively. Furthermore, the battery C-rate and ampere-hour throughput are also reduced by about 10%–15%. Despite the relatively high figures, the corresponding fuel cost savings are only about € 60 per year[3] with the current fuel costs. The fuel savings alone are most likely not substantial enough to motivate a server-based solution. However, if the servers are already in place for other systems, e.g., infotainment and safety, the marginal cost for implementing a route optimized system should be relatively low.

Appendix A
Selecting the Most Representative Trip

Each of the logged commuter trips is associated with a set of characteristic features aimed at describing the general driving conditions of the trip, e.g., fraction of time spent in different speed intervals, average deceleration, standstill time, etc. The features considered in this paper have been selected on the basis of the results in [29], in which more than 2000 trips were analyzed using factorial analysis to obtain 16 independent features correlated with fuel economy. The features that were possible to determine from the logged driving data were computed for each commuter trip. The most *representative trip* was then chosen as the trip with the least feature distance (in a standardized Euclidian-distance sense) to all the other commuter trips.

[3] Assuming European fuel costs and 226 working days per year with two trips a day.

Appendix B
Swedish Car Movement Database

The database contains logged driving data from roughly 500 passenger cars, which were randomly selected from a subset of the Swedish vehicle fleet. The cars were driven by private households in Kungsbacka municipality and Västra Götaland county, with the latter containing Sweden's second largest city, i.e., Göteborg. Each vehicle was logged for approximately two consecutive months, between 2010 and 2012, using a commercial GPS unit. The key parameters logged were latitude, longitude, altitude, velocity, and system time, which are all sampled from the GPS receiver at 2.5 Hz. If the GPS signal was lost for less than 10 s, the missing data were interpolated, and if the signal was lost for more than 10 s, a new trip was defined. Therefore, some journeys were broken up into several smaller trips, which are not necessarily connected, time- or position-wise, at least if the signal was lost for a longer time period. In this paper, no effort was made to reconstruct longer journeys that were broken up into two or more trips. For a more detailed description of the measurement equipment and the data collection, see [30].

Acknowledgment

The authors would like to thank the Swedish Hybrid Vehicle Centre and Chalmers Energy Initiative for funding the work.

References

[1] A. Sciarretta and L. Guzzella, "Control of hybrid electric vehicles," *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 60–70, Apr. 2007.
[2] F. R. Salmasi, "Control strategies for hybrid electric vehicles: Evolution, classification, comparison, and future trends," *IEEE Trans. Veh. Technol.*, vol. 56, no. 5, pp. 2393–2404, Sep. 2007.
[3] S. G. Wirasingha and A. Emadi, "Classification and review of control strategies for plug-in hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 111–122, Jan. 2011.
[4] Q. Gong, Y. Li, and Z.-R. Peng, "Optimal power management of plug-in HEV with intelligent transportation system," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, 2007, pp. 1–6.
[5] D. Ambühl and L. Guzzella, "Predictive reference signal generator for hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 4730–4740, Nov. 2009.
[6] C. Zhang and A. Vahidi, "Route preview in energy management of plug-in hybrid vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 2, pp. 546–553, Mar. 2012.
[7] Active Hybrid. [Online]. Available: http://www.bmw.com
[8] On Call. [Online]. Available: http://www.volvocars.com/
[9] ConnectedDrive. [Online]. Available: http://www.bmw.com
[10] R. Simmons, B. Browning, and V. Sadekar, "Learning to predict driver route and destination intent," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2006, pp. 127–132.
[11] J. Froehlich and J. Krumm, "Route prediction from trip observations," presented at the Society of Automotive Engineers (SAE) World Congr., Detroit, MI, USA, Apr., 2008, Paper 2008-01-0201.

[12] Q. Ye, L. Chen, and G. Chen, "Predict personal continuous route," in *Proc. 11th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2008, pp. 587–592.
[13] D. Filev, F. Tseng, J. Kristinsson, and R. McGee, "Contextual on-board learning and prediction of vehicle destinations," in *Proc. IEEE Symp. CIVTS*, Apr. 2011, pp. 87–91.
[14] A. Vogel, D. Ramachandran, R. Gupta, and A. Raux, "Improving hybrid vehicle fuel efficiency using inverse reinforcement learning," in *Proc. 26th AAAI Conf. Artif. Intell.*, 2012, pp. 384–390.
[15] L. Johannesson, M. Åsbogård, and B. Egardt, "Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 71–83, Mar. 2007.
[16] L. Johannesson, S. Pettersson, and B. Egardt, "Predictive energy management of a 4QT series-parallel hybrid electric bus," *Control Eng. Practice*, vol. 17, no. 12, pp. 1440–1453, Dec. 2009.
[17] S. Kermani, R. Trigui, S. Delprat, B. Jeanneret, and T. M. Guerra, "PHIL implementation of energy management optimization for a parallel HEV on a predefined route," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 782–792, Mar. 2011.
[18] V. Larsson, L. Johannesson, and B. Egardt, "Comparing two approaches to precompute discharge strategies for plug-in hybrid electric vehicles," presented at the 7th IFAC Symposium Advances Automotive Control, Tokyo, Japan, 2013, Paper ThB-2.
[19] R. Bellman and S. Dreyfus, *Applied Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1962.
[20] Autonomie. [Online]. Available: http://www.autonomie.net/
[21] G. Paganelli, S. Delprat, T. Guerra, J. Rimaux, and J. Santin, "Equivalent consumption minimization strategy for parallel hybrid powertrains," in *Proc. IEEE 55th VTC*, Spring 2002, pp. 2076–2081.
[22] A. Sciarretta, M. Back, and L. Guzzella, "Optimal control of parallel hybrid electric vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 3, pp. 352–363, May 2004.
[23] T. Hastie, T. Robert, and F. Jerome, *The Elements of Statistical Learning*, 2nd ed. New York, NY, USA: Springer-Verlag, 2009, ser. Springer Series in Statistics.
[24] V. Larsson, L. Johannesson, and B. Egardt, "Impact of trip length uncertainty on optimal discharging strategies for PHEVs," in *Proc. 6th IFAC AAC*, Munchen, Germany, 2010, pp. 55–60.
[25] P. Elbert, S. Ebbesen, and L. Guzzella, "Implementation of dynamic programming for n-dimensional optimal control problems with final state constraints," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 3, pp. 924–931, May 2013.
[26] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems*, 2nd ed. Berlin, Germany: Springer-Verlag, 2007.
[27] V. Larsson, L. Johannesson, and B. Egardt, "Benefit of route recognition in energy management of plug-in hybrid electric vehicles," in *Proc. Amer. Control Conf.*, Montreal, QC, USA, 2012, pp. 1314–1320.
[28] J. Wang, P. Liu, J. Hicks-Garner, E. Sherman, S. Soukiazian, M. Verbrugge, H. Tataria, J. Musser, and P. Finamore, "Cycle-life model for graphite-LiFePO4 cells," *J. Power Sources*, vol. 196, no. 8, pp. 3942–3948, Apr. 2011.
[29] E. Ericsson, "Independent driving pattern factors and their influence on fuel-use and exhaust emission factors," *Transp. Res. D*, vol. 6, no. 5, pp. 325–345, Sep. 2001.
[30] S. Karlsson, The Swedish car movement data project Final report, Dept. Energy Environ., Chalmers Univ. Technol., Göteborg, Sweden, Tech. Rep. [Online]. Available: http://publications.lib.chalmers.se/records/fulltext/172812/local_172812.pdf

Viktor Larsson received the M.Sc. degree from Luleå University of Technology, Luleå, Sweden, in 2008 and the Lic. Eng. degree from Chalmers University of Technology, Göteborg, Sweden, in 2011.

Since 2009, he has been with the Automatic Control Group, Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden. His research interest includes optimal control of hybrid and plug-in hybrid electric vehicles.

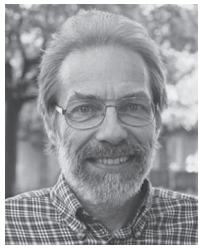
Lars Johannesson Mårdh (M'12) received the M.Sc. degree in automation and mechatronics and the Ph.D. degree in automatic control from Chalmers University of Technology, Göteborg, Sweden, in 2004 and 2009, respectively.

Since 2011, he has been with the Electromobility Group, Viktoria Swedish ICT, Göteborg, working with research on powertrain control within the Chalmers Energy Initiative. His main research interests include optimal control of hybrid and plug-in hybrid electric vehicles, control of auxiliary systems in trucks, active cell balancing, and system studies of hybrid vehicles.

Bo Egardt (SM'90–F'03) received the M.Sc. degree in electrical engineering and the Ph.D. degree in automatic control from Lund Institute of Technology, Lund, Sweden, in 1974 and 1979, respectively.

During 1980, he was a Research Associate with the Information Systems Laboratory, Stanford, CA, USA. From 1981 to 1989, he was with Asea Brown Boveri, where he was heavily involved in the introduction of adaptive control in the process industry. In 1989, he was appointed Professor of automatic control with Chalmers University of Technology, Göteborg, Sweden. His main areas of interest include adaptive and hybrid control and applications of control in the automotive area.

Dr. Egardt is a member of the editorial board for the *International Journal of Adaptive Control and Signal Processing*. He was an Associate Editor of IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY and of the *European Journal of Control*.



Sten Karlsson received the Ph.D. degree from Chalmers University of Technology, Göteborg, Sweden, in 1990.

He is a Senior Lecturer with the Department of Energy and Environment, Chalmers University of Technology. His current research focuses on energy efficiency and technology assessment, particularly concerning private cars and the electrification of vehicles.