Assessing the Potential of Predictive Control for Hybrid Vehicle Powertrains Using Stochastic Dynamic Programming

Lars Johannesson, Mattias Åsbogård, and Bo Egardt, *Fellow, IEEE*

***Abstract*—The potential for reduced fuel consumption of hybrid electric vehicles by the use of predictive powertrain control was assessed on measured-drive data from an urban route with varying topography. The assessment was done by evaluating the fuel consumption using three optimal controllers, each with a different level of information access to the driven route. The lowest information case represents that the vehicle knows that it is being driven in a certain environment, e.g., city driving, and that the controller has been optimized for that type of environment. The second highest information level represents a vehicle equipped with a GPS combined with a traffic-flow information system. In the highest information level, the future power demand is completely known to the control system, hence, the corresponding optimal controller results in the minimal attainable fuel consumption. This paper showed that good performance (1%–3% from the minimal attainable fuel consumption) can be achieved with the lowest information case, with a time-invariant controller that is optimized to the environment. The second highest information level results in less than 0.2% higher consumption than the minimal attainable on the studied route. This means that it is possible to design a predictive controller based on information supplied by the vehicle-navigation system and traffic-flow-information systems that can come very close to the minimal attainable fuel consumption. A novel algorithm that uses information supplied by the vehicle-navigation system was presented. The proposed algorithm results in a consumption only 0.3% from the minimal attainable consumption on the studied route.**

***Index Terms*—Hybrid vehicles, predictive control, stochastic optimal control.**

I. Introduction

OVER THE last years, many new telematic systems have been introduced in road vehicles. For instance, global-positioning systems (GPS) and mobile phones have become a *de facto* standard in premium cars. With the introduction of these systems, the amount of information about the traffic environment available in the vehicle has increased. The main reasons for collecting this type of information have been to improve comfort or to reduce the workload of the driver, e.g., through the use of active cruise control and navigation aids. However, emerging technologies for active vehicle safety also rely on the existence of accurate traffic-environment information.

Manuscript received February 1, 2006; revised July 6, 2006, September 8, 2006, and September 14, 2006. This work was supported by the Swedish Energy Agency. The Associate Editor for this paper was B. De Schutter.

The authors are with the Department of Signals and Systems, Chalmers University of Technology, SE41296 Gothenburg, Sweden.

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.2006.884887

Another trend in automotive research and development is the introduction of hybrid electric vehicles (HEVs). This trend is driven by the need for reduced environmental impact of road transport. An HEV is characterized by the existence of an electric energy buffer (EB) in the powertrain (i.e., engine and transmission), enabling the vehicle to be propelled by a combination of power from a primary power unit and power from the buffer. The primary power unit can be an internal combustion engine (ICE) or a fuel cell, and the EB is normally a battery or a supercapacitor. HEVs offer two main advantages compared to conventional vehicles: The primary power unit can be operated close to its optimal working point, while temporary surplus or deficit of propulsion power is handled by the buffer, and braking energy can be regenerated and stored in the buffer. It is well known that the energy-management strategy (EMS) of an HEV largely influences its competitiveness [1].

With the availability of traffic information, predictions of the vehicle-propulsion load can be made. This enables predictive control of the hybrid powertrain, potentially increasing the overall efficiency. Previous work within this field includes [2], where model predictive control (MPC) is used in a test vehicle equipped with a telematic system. Also, in [3], a charge/discharge control system is proposed, where the controller schedules the battery usage based on information about traffic conditions and road grade along the traveled route. In contrast to earlier publications, the main purpose of this paper is not to develop a predictive-control scheme that can meet the demands of real-time implementation. Instead, the focus is on investigating what can be achieved if the available information is used optimally. This is done in order to assess the potential of predictive control for HEV powertrains, as well as to determine what type of traffic-environment information that should be provided to the powertrain controller.

The approach in this paper is to model a specific transport mission, i.e., a specific vehicle traveling a specific route, as a stochastic process based on collected data. The route is divided into a number of intervals, and the vehicle's state at the end of each interval is modeled by a discrete time Markov chain.[1] Here, we exploit the fundamental property of Markov chains: Conditioned on the present state, future states are independent of the past. The transition probabilities of the Markov chain are identified from collected driving data.

Three types of optimal controllers are studied. Each controller represents a different level of access to information about

[1] There is a vast amount of literature covering the properties of Markov chains, e.g., [4].

the transport mission. The controller with the most detailed information access simulates the unrealistic situation where the future power demand is completely known to the control system. Hence, this controller can be regarded as an ideal controller. Under these conditions, the controller provides an optimal-control trajectory for the specific velocity trajectory, or drive cycle, considered.

The second highest information level represents the case, where the controller has feedback of the position along the specific route. In addition to this, the controller has access to statistical data about the traffic flow along the route. This represents a vehicle equipped with a GPS combined with a traffic-flow information system. The information access is modeled with a position-dependent Markov chain, with the Markov state consisting of the velocity and the power demand. The optimal-control law is derived by solving a finite-horizon stochastic dynamic programming problem. The resulting optimal-control law is position dependent and optimal in an average sense when evaluated against the position-dependent Markov chain.

The lowest information level represents what can be achieved if the controller knows the transport mission but has no feedback of the exact position. This information level results in a time-invariant-control law that is optimized against a homogeneous, i.e., position-invariant Markov-chain model of the collected driving data, this is similar to studies in [5]–[7]. Hence, there is no position dependence, and the Markov state consists of the velocity and the power demand that the powertrain must deliver to fulfill the driver's request. This information level represents a vehicle that is being driven in a certain environment, e.g., city driving, and that the controller has been optimized for that type of environment.

The potential fuel savings with predictive control are assessed by comparing the fuel consumption resulting from the application of each of the three controllers. Results for three variants of the powertrain with different degrees of hybridization are presented.

Conclusions made from this paper forms the inspiration for a novel predictive algorithm for the case when the controller has feedback of the position along the given route. The proposed algorithm is suitable for real-time implementation.

II. Stochastic Modeling of a Transport Mission

As indicated already in the Introduction, the basis for the assessment of different predictive-control strategies is stochastic modeling of a specific transport mission in urban traffic. Depending on the amount of information made available to the controller, different models are needed. Two different stochastic processes that model the transport mission were identified, from the data collected during actual driving. The processes are of Markov type. This means that the future evolution of the processes, conditioned on their present state, is independent of the past.

The various steps involved in the modeling work will be described in more detail below.

A. Data Collection

Drive data were collected in Gothenburg during two weeks in June of 2004. The data were collected between 9:30–11:30 h and 13:00–16:00 h. The time intervals were chosen in order to avoid rush hours and to measure under similar conditions: relatively low traffic density and no traffic congestions. Measurements were done with a Toyota Prius 2, with a hired driver not affiliated with the research project. The driver was experienced to city traffic and had a calm drive style. The weather was sunny, partially cloudy, or overcast with scattered showers. It is therefore concluded that the data were collected under similar weather conditions. The data were collected on the same route by driving it 37 times, in total ≈50 000 s of measured driving. The studied route stretches along 16 km of mixed city driving. The altitude varies between 2 and 95 m, and the speed limits are 30, 50, and 70 km/h.

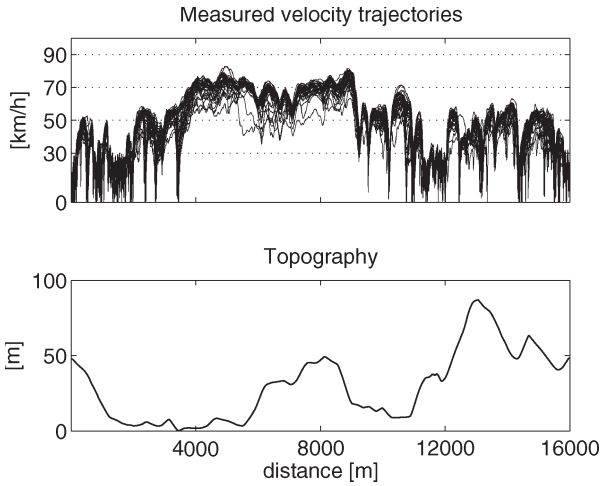

Fig. 1. Measurements. The 37 measured velocity trajectories are shown at the top. Below is the altitude profile of the route.

The measurements were done with a GPS which logged velocity and position at every second. Before collecting the drive data, the route was carefully measured, and the position for every crossing, crossroad, speed bump, and traffic light was logged with the GPS. The distance traveled between every logged position was calculated by integrating the velocity given by the GPS. This measurement of the route is then used as a nominal route, which all following runs are calibrated against, and the measured positions are used as checkpoints for this calibration. The calibration is done in order to improve the position accuracy, meaning that the relative position error between the measured trajectories is reduced. This reduction is achieved by manually logging when the vehicle passes a checkpoint. The measured velocity trajectories are then fitted to the manually logged checkpoints under the assumption that the error in traveled distance compared to the nominal route grows linearly. This method results in a repeatability accuracy of 5 m when measured at standstill positions. The calibrated-velocity trajectories and the altitude profile are shown in Fig. 1.

B. Homogeneous Markov-Chain Model

The objective of the first model constructed is to describe the "average" (stationary; time and position invariant) distribution of velocity and power demand along the route without taking into account the specific position. Instead of directly determining this model from raw data, an intermediate model—here

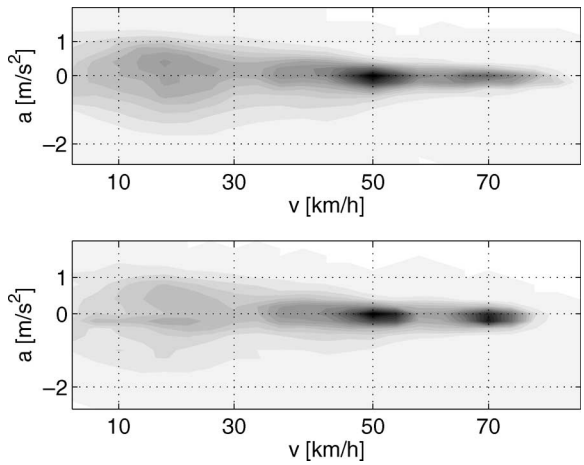

Fig. 2. Velocity–acceleration diagrams of the measurement and the model. The model is shown at the top. The gray scale shows the density function of velocity and acceleration. A darker color means a higher density.

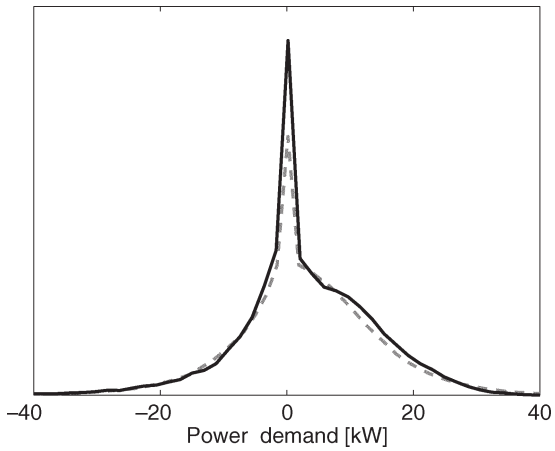

Fig. 3. Distribution of power demand for a vehicle with a mass of 1800 kg. The dashed line is the resulting distribution of power demand when simulated on the stochastic model, and the solid line is the distribution of power demand when simulated on the measured drive cycles.

referred to as a stochastic drive-cycle generator—was identified in [8] from the collected data. The sole purpose of this drive-cycle generator is to smoothen the data before creating the Markov matrix in the Markov-chain model. The drive behavior is in [8], modeled as a discrete-time continuous-state-space time-invariant Markov process with acceleration and velocity as the Markov states. The acceleration A is a stochastic variable modeled as beta distributed with the beta parameters dependent on previous acceleration a and velocity ν.[2]

$$A_{k+1} \sim \beta(a_k, \nu_k). \tag{1}$$

The model is verified by comparing simulated velocities and accelerations with the measurements. A velocity–acceleration diagram for both measured and simulated data is shown in Fig. 2. In Fig. 3, the stationary distribution of the power demand for simulations with a vehicle of 1800 kg is presented. The figures show that the model is able to reproduce the behavior exhibited in the measurements.

By extensive simulations with the drive-cycle generator, the data set is expanded from ≈50 000 s in the measurements to 2 500 000 s. This expansion is done in order to smoothen the data and to enable a straightforward identification of the transition probabilities when constructing later the Markov-chain matrix of the power demand and velocity. A time interval of 2 500 000 s is enough to guarantee convergence of the transition probabilities in the Markov-chain matrix. The next step is to simulate a chassis on the generated drive data. The chassis model has three inputs: a, ν, and the road slope θ. The road slope comes from the topography shown in Fig. 1. The output from the chassis model is the power demand P_{demand} that the powertrain needs to deliver in order to follow the generated drive cycle. By quantizing P_{demand} and V, a Markov-chain model is now identified from the simulated data. The state vector in the Markov chain is defined as

$$X_k^h = (\nu_k, P_{\text{demand},k}). \tag{2}$$

The conditioned probability distribution of the state at the next time step, given the current state, is given by the transition probabilities

$$\mathbb{P}(P_{\text{demand},k+1} = p_{\text{demand},k+1}, V_{k+1} = \nu_{k+1} | p_{\text{demand},k}, \nu_k) \tag{3}$$

which are defined for all possible combinations of p_{demand} and ν. All these transition probabilities are determined from the simulated data.

Note that the Markov-chain model is time invariant, which implies that the transition probabilities are independent of k.

The described prediction model is by no means the only way to predict the power demand and velocity without position-dependent information. It might, for instance, be beneficial to base the predictions on information of the vehicle's state at several previous time steps or to invoke dependence of the road slope. However, the computational time for explicitly finding a time-invariant optimal controller increases exponentially for every extra variable that is included in the prediction model's state. More advanced controllers are, therefore, not likely to be implemented in HEVs in the foreseeable future.

C. Position-Dependent Markov-Chain Model

The second Markov-chain model is aimed at describing the ensemble of velocity trajectories as a function of position along the route. First, the route was divided into a number of discrete intervals measured in distance. The velocity and the slope at the end of each interval were approximated by one of a finite number of values. The result of this quantization is a representation of the data, which is discrete in both velocity and position. Details for the two different models are given below. At a number of positions along the route, the vehicle may come to a complete stop. Naturally, it will then remain at zero velocity for some period of time. To handle this, a model describing the length of the time interval is needed. Using the exponential

[2]Capital letters are, in this section, used to denote stochastic variables; the corresponding realizations are denoted by small letters. Subscript k is the time index.

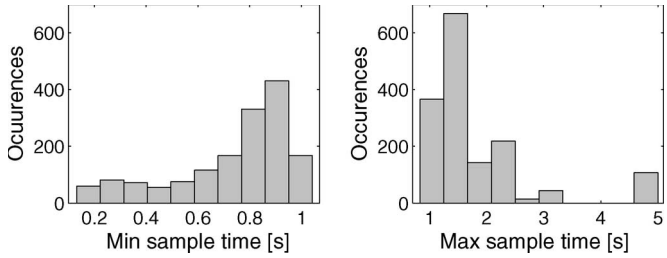

Fig. 4. Occurrences of maximum and minimum sample time.

distribution, a good approximation of the time-interval lengths of the collected data is achieved. However, it is required that the mean value of the interval length is individually set for each possible stop position.

With the route divided into discrete steps, the Markov model is defined by the transition probabilities, linking the state at one position to the state at the next position. In this case, the state vector is defined as

$$X_k^p = (V_k, A_k) \tag{4}$$

where, in this case, k is a position index. Since the slope is given as a deterministic function of the position, p_{demand} is a function of the state X_k^p.

To be more specific, given the velocity ν_k and the acceleration a_k at a certain position k along the route, the probability distribution for the velocity at the position $k+1$ is given by the transition probabilities

$$\mathbb{P}_k(V_{k+1} = \nu_{k+1} | \nu_k, a_k) \tag{5}$$

of a nonhomogeneous Markov chain. Note that the transition probabilities now depend on the position k and, therefore, implicitly on the slope. In a similar way, the transition probabilities for the acceleration are given by

$$\mathbb{P}_k(A_{k+1} = a_{k+1} | \nu_k, a_k). \tag{6}$$

The transition probabilities are determined from the collected data. The velocity is assumed to change linearly between every position. Since the velocity is a stochastic process, the time needed to go between two fixed positions will vary. Therefore, the distances between the points on the route are adapted to the possible velocities at each position, keeping the time step as close to 1 s as possible. With the velocity resolution set to 1 m/s, the longest sample time that can occur at any position is approximately 5 s, and the shortest is approximately 0.2 s. However, as shown in Fig. 4, the vast majority of the time steps lie around 1 s. The reason for choosing a low-velocity resolution of 1 m/s is the relative lack of data. Remember that each position is only visited 37 times in the data set. This sparse quantization and the varying time step means that the drive cycles generated from (5) will not have the same stationary distribution for the power demand as the measured drive cycles. The acceleration is quantized with three levels, distinguishing between deceleration, acceleration, and constant velocity.

TABLE I
MAIN PARAMETER VALUES FOR FULL-SIZE TAXI CAR

Vehicle concepts T1-T3		
Code	*Parameter*	*Value*
T1-T3	Vehicle configuration	Parallel, FWD
	Total Vehicle Mass	1800 kg
	Gearbox	5 stepped automatic
	Gearbox efficiency	0.97
	ICE friction torque ($T_{ICE,friction}$)	30 Nm
T1-T2	ICE max power	125 kW
T1	Maximal EM power	12 kW
	Maximum EM torque	160 N
	Buffer type and max power	NiMH Battery,12 kW
	Effective buffer capacity	300 kJ
	Buffer capacity	2 MJ
T2	Maximal EM power	16 kW
	Maximum EM torque	160 N
	Buffer type and max power	NiMH Battery,16 kW
	Effective buffer capacity	400 kJ
	Buffer capacity	2.67 MJ
T3	ICE type and max power	Atkinson, 43 kW
	Maximal EM power	30 kW
	Maximum EM torque	320 N
	Buffer type and max power	NiMH Battery, 32 kW
	Effective buffer capacity	800 kJ
	Buffer capacity	5.2 MJ

III. VEHICLE MODEL

A. Vehicle Description

The studied HEV is a city taxi with a parallel configuration for the hybrid powertrain. Three variants of the powertrain are studied. The vehicle specifications are given in Table I. Note that total vehicle mass includes the driver and passengers. The HEV is powered by an ICE with a nickel metal-hydride (NiMH) battery as a buffer. The mechanical power produced by the ICE is transferred to the driving wheels via a five-step gearbox and a differential gear. At the ICE flywheel, an electric machine (EM) can add or absorb torque. The resulting electric power, to or from the EM, is transferred to the EB via an electric-power converter. An overview of the chassis and powertrain modeling is shown in Fig. 5.

B. Chassis Model

The chassis model outputs at the differential gear the power demand p_{demand} and the shaft speed ω_{demand} that is required to propel the vehicle at a certain velocity, acceleration, and road slope. Longitudinal driving dynamics need to be considered in order to assure vehicle stability during regenerative braking. Therefore, a longitudinal bicycle model is used to represent the chassis. During braking, the vehicle is assumed to absorb as much energy as possible through the driving wheels but without ever locking them.

C. Powertrain Model

Three input signals are used in the powertrain model: p_{demand}, ω_{demand}, and the ICE power p_{ICE}. The signals p_{demand} and ω_{demand} come from the chassis model and arise from the driver's intention, while p_{ICE} is determined by the EMS. The gear is chosen so that p_{ICE} is delivered at the engine speed ω_{ICE} that results in the highest possible engine efficiency $\eta_{\text{ICE}}(p_{\text{ICE}}, \omega_{\text{ICE}})$. The engine efficiency is given by linear

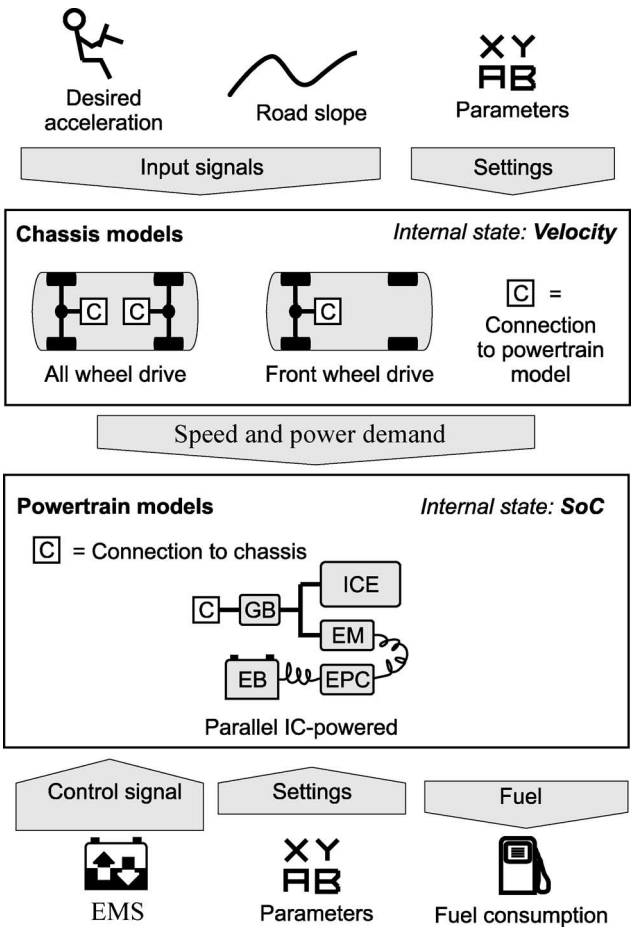

Fig. 5. Overview of chassis and powertrain modeling. The chassis models are used to precalculate the required torque and power for all combinations of chassis state and input signals. The powertrain models are used to determine the fuel consumption for all possible combinations of power, torque, control signal, and powertrain state.

interpolation in an ICE efficiency map. The efficiency maps for the two engines used in this paper are shown in Figs. 10 and 11. There are no restrictions on gear changes, and the inertia of the engine is not modeled. With ω_{ICE} and p_{ICE} determined, the buffer power p_{EB} is for $p_{\mathrm{demand}} > 0$, given by

$$p_{\mathrm{EB}} = p_{\mathrm{ICE}} - p_{\mathrm{demand}}/\eta_g - \omega_{\mathrm{ICE}} \times T_{\mathrm{ICE,friction}} \quad (7)$$

and for $p_{\mathrm{demand}} \leq 0$ by

$$p_{\mathrm{EB}} = p_{\mathrm{ICE}} - \eta_g p_{\mathrm{demand}} - \omega_{\mathrm{ICE}} \times T_{\mathrm{ICE,friction}}. \quad (8)$$

Thus, p_{EB} is defined as positive when charging the buffer. The gearbox efficiency η_g is assumed to be the same for all gears. If $p_{\mathrm{ICE}} > 0$, the engine friction is included in the ICE efficiency map, and $T_{\mathrm{ICE,friction}} = 0$. The EM efficiency $\eta_{\mathrm{EM}}(p_{\mathrm{EB}}, \omega_{\mathrm{ICE}})$ is given by linear interpolation in an efficiency map, and the electric-converter efficiency η_{EC} is modeled as constant. The battery efficiency $\eta_{\mathrm{BAT}}(p_{\mathrm{BAT}})$ is modeled by a simple resistive circuit, where the open-circuit voltage is independent of the SoC. The SoC at the next time point is, thus, determined by

$$\mathrm{SoC}_{k+1} = \mathrm{SoC}_k + \Delta t \times \eta_{\mathrm{BUF}}(p_{\mathrm{EB}}, \omega_{\mathrm{ICE}}) \times p_{\mathrm{EB}}/Q \quad (9)$$

where $\eta_{\mathrm{BUF}}(p_{\mathrm{EB}}, \omega_{\mathrm{ICE}})$ denotes the total efficiency of electric energy flow, Δt the time step, and Q the effective capacity. Figs. 12 and 13 show $\eta_{\mathrm{BUF}}(p_{\mathrm{EB}}, \omega_{\mathrm{ICE}})$ for the T1 and T3 concepts. A number of restrictions must be considered when simulating and optimizing the powertrain energy flow. The buffer power p_{EB} and the SoC are limited according to

$$p_{\mathrm{EB}} \in [p_{\mathrm{EB,min}}, p_{\mathrm{EB,max}}] \quad (10)$$

$$\mathrm{SoC} \in [0, 1]. \quad (11)$$

It should be noted that SoC is scaled in such a way that the allowed interval [0, 1] refers to effective buffer capacity. To avoid excessive wear of the EB, the SoC interval is only 15% of the total battery capacity and placed symmetric around 50% of the total buffer capacity. The ICE power p_{ICE} is limited according to

$$p_{\mathrm{ICE}} \leq p_{\mathrm{ICE,max}}. \quad (12)$$

The ICE and EM torques T_{ICE} and T_{EM} are further limited by rotational speed-dependent constraints

$$T_{\mathrm{ICE}}(\omega_{\mathrm{ICE}}) \leq T_{\mathrm{ICE,max}}(\omega_{\mathrm{ICE}}) \quad (13)$$

$$T_{\mathrm{EM}}(\omega_{\mathrm{ICE}}) \in [T_{\mathrm{EM,min}}(\omega_{\mathrm{ICE}}), T_{\mathrm{EM,max}}(\omega_{\mathrm{ICE}})]. \quad (14)$$

IV. Dynamic Programming

Dynamic programming is used to find the optimal control, minimizing the expected fuel consumption for each vehicle concept and each information level. The control signal u is: $u = p_{\mathrm{ICE}}$. When quantizing u, care should be taken so that at all states where it is possible, one control action should result in a p_{EB} exactly equal to zero.

A. Position-Invariant Controller

An infinite-horizon problem is formulated on the homogenous Markov chain. The optimization finds an optimal policy, $u = \pi(x)$ that minimizes the expected fuel consumption over an infinite horizon [5]–[7]:

$$J_\lambda^\pi(x_0) = \lim_{N\to\infty} E\left\{\sum_{k=0}^{N-1} \lambda^k c\left(x_k, \pi(x_k)\right)\right\} \quad (15)$$

where c is the cost for one time step, λ is the discount factor, x_k is the dynamic-state vector at the kth time point, and $E\{\ldots\}$ denotes the expectation with respect to the considered prediction model defined by the time-invariant Markov chain. The dynamic-state vector is defined as

$$x_k = [\mathrm{SoC}_k, p_{\mathrm{demand},k}, \nu_k]. \quad (16)$$

The discount factor $\lambda < 1$ assures convergence of the infinite sum. The optimization is subjected to constraints (10)–(14). The cost for one time step $c(p_{\mathrm{demand}}, \nu_k, u_k)$ is simply the instantaneous fuel consumption, which is precalculated in order to reduce computational time.

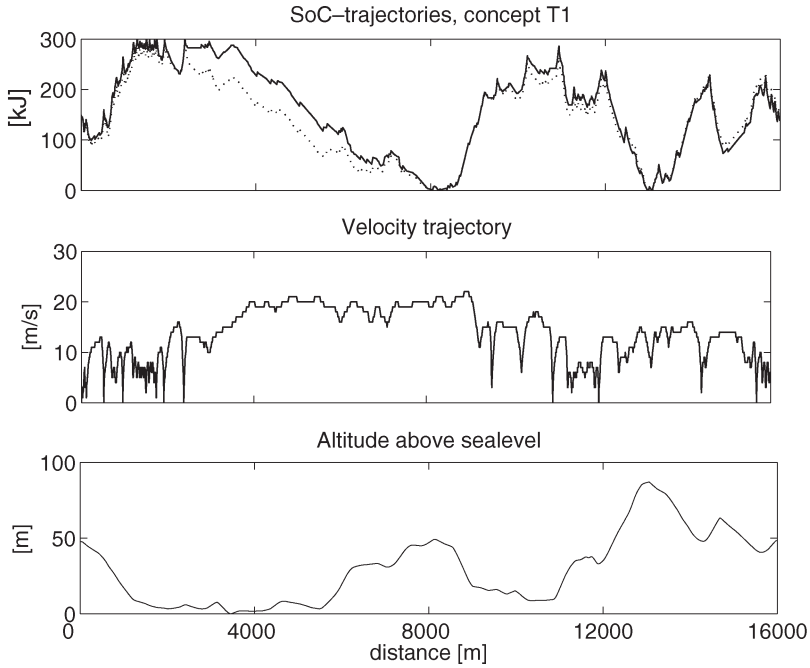

Fig. 6. Upper graph shows simulated SoC trajectories for the concept T1. The solid line shows the results from the ideal controller, and the dotted line is the position-dependent controller. The second graph shows the quantized version of the measured drive cycle, on which the simulation is conducted. The third graph shows the topography of the route.

The optimal policy $u = \pi(x)$ is found by using a modified policy-iteration algorithm [9]. The steps in the modified policy iteration are as follows.

1) Initial guess: Set $i = 1$. Provide an initial guess for $\pi_i(x)$ and the discounted infinite-horizon future cost $J_\lambda^{\pi_{i-1}}(x)$.
2) Policy evaluation: Calculate the cost $J_\lambda^{\pi_i}(x)$ of the policy $\pi_i(x)$ by iterating

$$J_\lambda^{\pi_i}(x_k) = c(x_k,u) + \lambda \sum_{x_{k+1}\in X} \mathbb{P}(X_{k+1} = x_{k+1}|x_k,u) J_\lambda^{\pi_i}(x_{k+1})$$

backward N times, starting from $J_\lambda^{\pi_i}(x_N) = J_\lambda^{\pi_{i-1}}(x)$ and ending with the truncated cost $J_\lambda^{\pi_i}(x) = J_\lambda^{\pi_i}(x_0)$. If $|J_\lambda^{\pi_i}(x_k) - J_\lambda^{\pi_i}(x_{k+1})| < \epsilon$ when iterating backwards, terminate the iterations with the answer $J_\lambda^{\pi_i}(x) = J_\lambda^{\pi_i}(x_k)$.

3) Policy improvement: Improve the policy by taking one value-iteration step:

$$\pi_{i+1}(x_0) = \arg\min_u \left\{ c(x_0,u) + \lambda \sum_{x_1\in X} \mathbb{P}(X_1 = x_1|x_0,u) J_\lambda^{\pi_i}(x_1) \right\}$$

$$J_\lambda^{\pi_i+1}(x_0) = \min_u \left\{ c(x_0,u) + \lambda \sum_{x_1\in X} \mathbb{P}(X_1 = x_1|x_0,u) J_\lambda^{\pi_i}(x_1) \right\}.$$

4) Check for convergence: If $|J_\lambda^{\pi_{i+1}}(x_0) - J_\lambda^{\pi_i}(x_k)| < \epsilon$, then terminate the algorithm with the answer $\pi(x) = \pi_{i+1}(x)$. If $\pi_{i+1}(x)$ has not converged, increase the index: $i = i + 1$, and go to step 2).

Note that since SoC_{k+1} is not constrained to be on the state grid, calculating $J_\lambda^{\pi_i}(x_{k+1})$ involves interpolation. Linear interpolation is used here.

The difference between a modified policy-iteration algorithm and a standard policy-iteration algorithm is that in the policy iteration, $N = \infty$. The cost of the policy at the policy-evaluation step is then found by solving the linear equation

$$J_\lambda^{\pi_i}(x_0) = c(x_0,u) + \lambda \sum_{x_1\in X} \mathbb{P}(X_1 = x_1|x_0,u) J_\lambda^{\pi_i}(x_0) \quad (17)$$

for $J_\lambda^{\pi_i}(x_0)$. The policy iteration converges in a finite number of iterations with no need for an ϵ. However, the computational cost for solving the linear equation motivates the use of the modified policy iteration, where the cost of each evaluated policy is approximated with a truncated horizon $N < \infty$. The computational time for one iteration in the policy-evaluation step is considerably less than the computational time for one value-iteration step.

The optimal policy $\pi(x)$ is a nonlinear time-invariant control law given as a lookup table on the discrete grid x. When the policy is used as an EMS in a real vehicle, interpolation is needed.

B. Position-Dependent Controller

Dynamic programming applied to the position-dependent Markov chain leads to a finite-horizon problem. The expected fuel consumption, if policy π_k is used for the finite horizon, and the system is in state x_0 at the initial position is defined by

$$J_N^\pi(x_0) = E\left\{ \sum_{k=1}^{N-1} c_k(x_k, \pi_k(x_k)) + r_N(\text{SOC}_N) \right\} \quad (18)$$

where c_k is the cost for one step, x_k is the dynamic-state vector at the kth position, and $r_N(\mathrm{SOC}_N)$ is a final cost. The final cost $r_N(\mathrm{SOC}_N)$ is used to control the final SoC. The dynamic-state vector is defined as

$$x_k = [\mathrm{SoC}_k, \nu_k, a_k]. \tag{19}$$

The optimization is subjected to constraints (10)–(14).

The optimal policy $u_k = \pi_k(x_k)$ is found by backward iterations. Standstill is modeled by one extra decision epoch, where the ICE power is constant during the stop. Note that SoC_{k+1}, here, is constrained to be on the state grid; thus, interpolation is not used. This means that the discretization of the SoC must be done with a high resolution to keep the discretization noise on the control signal $u = p_{\mathrm{ICE}}$ at an acceptable level.

C. Ideal Controller

The drive cycle here is completely known to the controller. The optimal policy $u_k = \pi_k(x_k)$, where k is either a position or time index, is found by backward iterations from the final position or time sample. Note that since the drive cycle is perfectly known, the dynamic state is simply $x_k = \mathrm{SoC}_k$.

V. Results and Discussion

Three different variants of a parallel hybrid taxi are included in this paper, representing different degrees of hybridization. The fuel consumption for the optimal controllers is used to assess the value of information and to quantify the potential of predictive control. The sparse quantization and the varying time step for the position-dependent Markov chain means that the controllers cannot be evaluated in a straightforward and fair manner using the same drive cycle. The evaluation is instead done in two steps. First, the position-dependent controller is compared with the ideal controller on quantized versions of the measured drive cycles. Remember that the position-dependent controller represents that the EMS uses a GPS with traffic-flow information to plan the use of the buffer. The ideal controller represents the unrealistic situation where the future power demand is completely known to the control system. The quantization of the measured drive cycle is the same as the one used when constructing the position-dependent Markov chain. Simulating with this quantization of the measured drive cycles removes the need for interpolation in the policies and reduces the effect of the sparse gridding of the velocity. This means that, in this first comparison, there is no interpolation error. Instead, the sparse quantization of the drive cycle can be considered as a modeling error, which results in more aggressive driving than what was measured.

The second step in the evaluation is to compare the position-invariant controller with the ideal controller. Remember that the position-invariant controller is derived from the lowest information level representing what can be achieved if the vehicle "knows" that it is being driven in a certain environment, e.g., city driving, and has a controller that is optimized for the environment. Here, the comparison is done by simulating the vehicle variants with the two controllers both on the measured drive cycles and on drive cycles generated from (1). The drive cycles are not quantized and the time step is 1 s. Interpolation techniques must be used to find $u(x)$ from the policies. This means that in this second comparison there is an interpolation error.

TABLE II
Optimization Parameters

Optimization parameters	
Parameter	*Value*
Nr of gridpoints for p_{demand}	40
Nr of gridpoints for v	23
Nr of gridpoints for *SoC*	41
λ	0.995

TABLE III
Overview of the Results

Simulated fuel consumption $\frac{liters}{100km}$			
Code	*Controller*	*Stochastic model*	*Measured drive cycle*
T1	position invariant	6.51	6.36
	ideal	6.37	6.31
	buffer deactivated		8.78
T2	position invariant	6.37	6.30
	ideal	6.31	6.24
T3	position invariant	5.11	5.00
	ideal	4.97	4.88

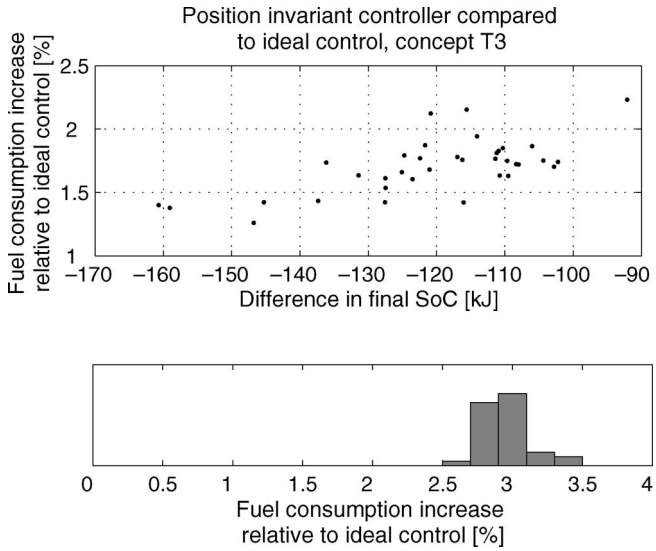

Fig. 7. Fuel consumption increase relative to ideal control with position-invariant control when simulated separately on all measured drive cycles. The relative fuel consumption and difference in final SoC is shown at the top. The dots are the outcome for each of the measured drive cycles. The y-axis is the relative increase in fuel consumption compared to ideal control, and the x-axis is the difference in final SoC compared to ideal control. The bottom graph shows the distribution of the relative increase in fuel consumption compared to ideal control when the difference in final SoC is weighted linearly into the fuel consumption. The average relative increase is then 2.9%.

A. Comparison Between the Position-Dependent Controller and the Ideal Controller

The vehicle concepts are simulated on ten quantized versions of the measured drive cycles. This paper shows practically no difference in consumption between the ideal and the position-dependent controller on all simulated drive cycles and for all studied concepts. The difference in consumption is less than 0.2% for all simulated drive cycles with no difference in the final SoC. This means that if the information gathered on the 37 runs along the route is used optimally, the performance will

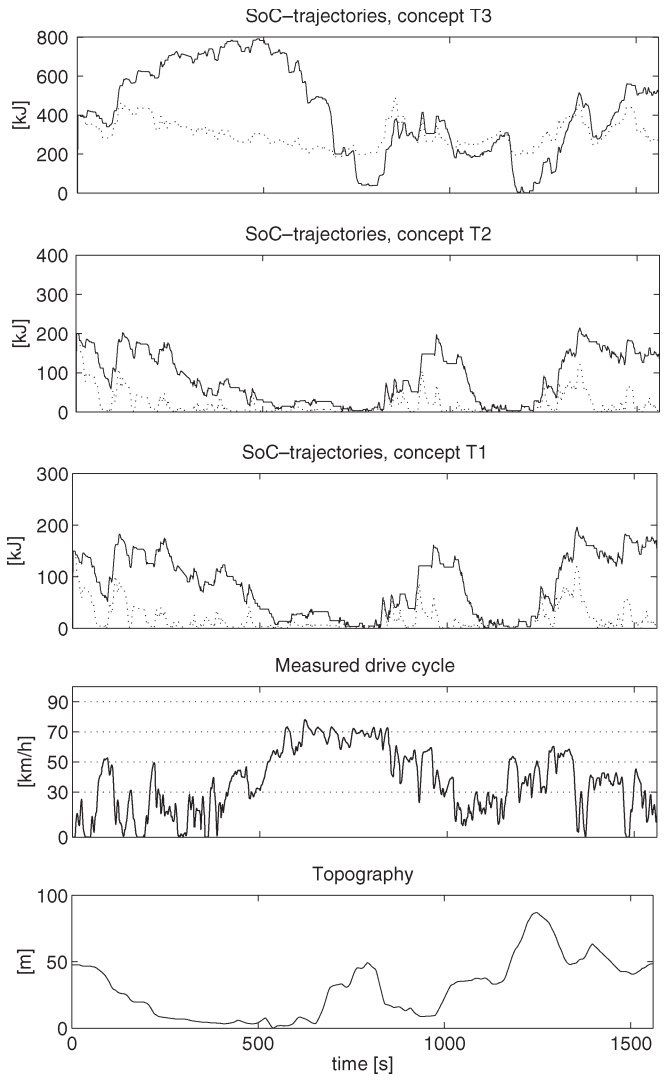

Fig. 8. SoC trajectories for the three concepts when simulated on one of the measured drive cycles. In the three upper graphs, the solid lines show the results from the ideal controller, and the dotted lines are the position-invariant controller.

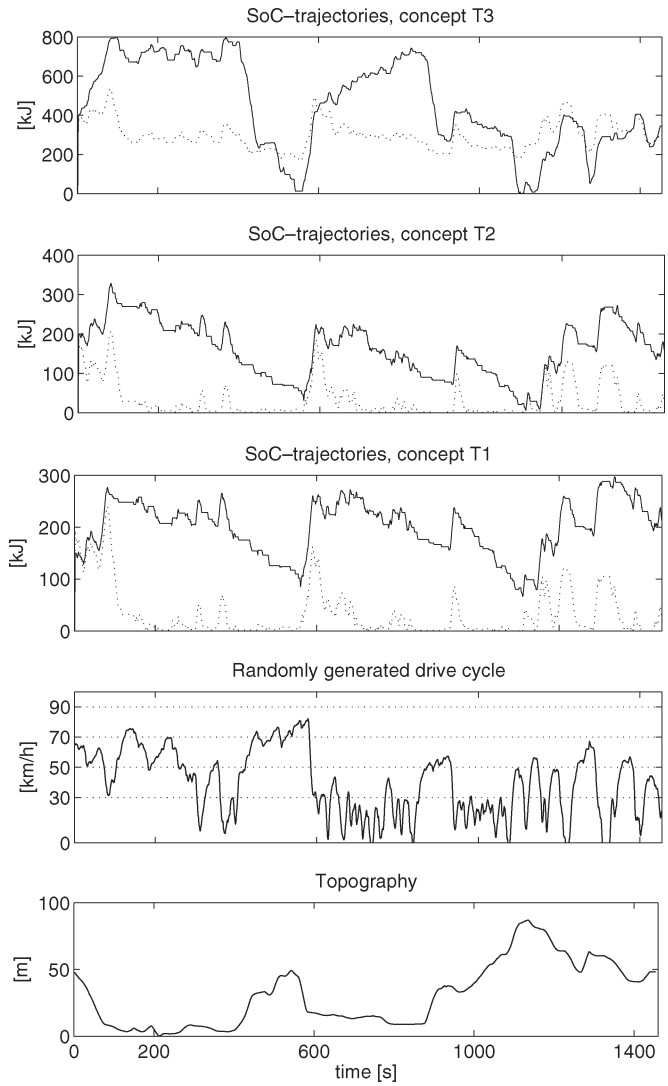

Fig. 9. SoC trajectories for the three concepts when simulated on a randomly generated drive cycle. In the three upper graphs, the solid lines show the results from the ideal controller, and the dotted lines are the position-invariant controller.

be the same as if the future power demand was completely known. This result means that the planning of the use of the buffer is insensitive to the fine structure of the driving that lies ahead, only the "average" information about the route is significant. Fig. 6 shows the SoC trajectory for the T1 concept, and it is shown that the planning is done on a long horizon, where energy is saved for the high-speed section and the buffer is depleted at the two altitude peaks. This behavior is the same for all concepts and all simulated drive cycles.

B. Comparison Between the Position-Invariant Controller and the Ideal Controller

The vehicle concepts are simulated both on the measured drive cycles and on random drive cycles generated from (1). The sample time is 1 s. The quantization grid used for calculating the position-invariant controller is decided by a stepwise refinement starting from an initial sparse grid. For each refined grid, the optimal policy $\pi(x)$ is calculated (using the policy from the previous gridding as an initial guess) and simulated on the stochastic model. This refinement of the grid is done until no significant improvement of the fuel consumption is noted in the simulations. The length of the simulations is set to be 20 000 s: long enough to remove the need to consider the final SoC. When simulating the policies, a final policy-improvement step is done at each time step instead of interpolating in $\pi(x)$. The control signal is, at each time step, given by

$$u(x_0) = \arg\min_u \left\{ c(x_0,u) + \lambda \sum_{x_1 \in X} \mathbb{P}(X_1 = x_1 | x_0, u) J_\lambda^{\pi_i}(x_1) \right\}. \tag{20}$$

This improvement step is done in order to increase the accuracy without increasing the detail of the grid. Increasing λ shows no improvement in the simulations.

The optimization settings are given in Table II, and the simulated fuel consumptions are presented in Table III. With the results in the previous section showing that ideal performance is attainable with an energy-management system that plans

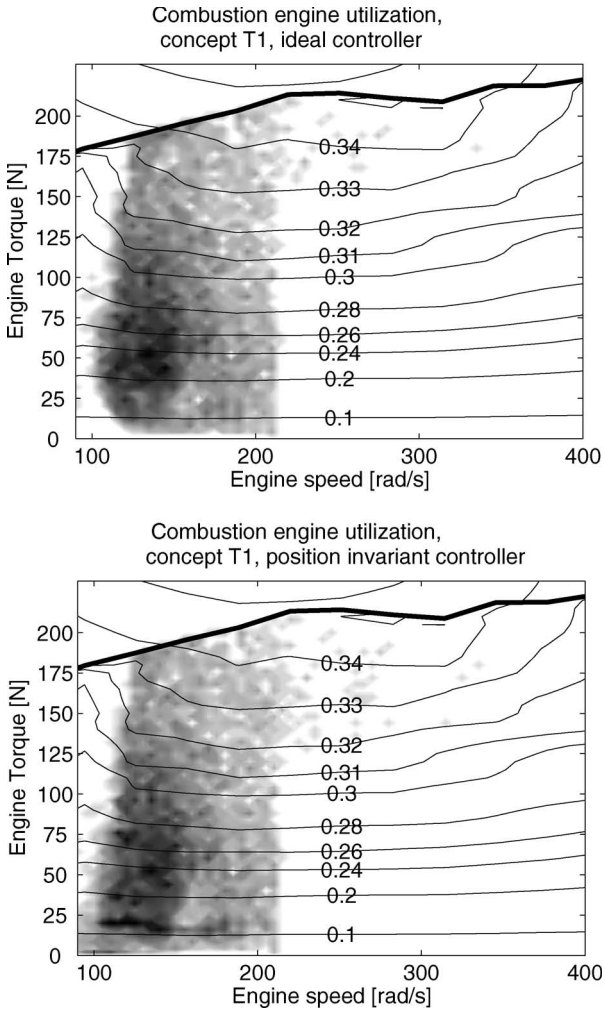

Fig. 10. ICE-utilization concept T1. A darker color means a higher frequence for the operating point. The bold line is the maximum engine torque.

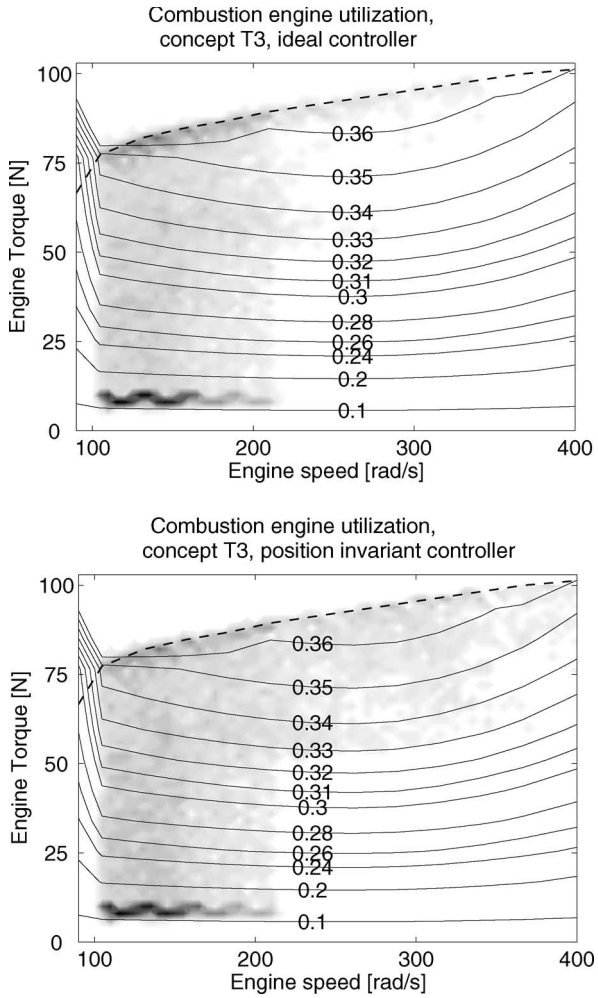

Fig. 11. ICE-utilization concept T3. A darker color means a higher frequence for the operating point. The dashed line is the maximum engine torque.

the activation of the buffer along a given route, the difference between the ideal and the position-invariant controller is interpreted as the savings that can only be achieved if position-dependent predictions of the route are used. When simulated on the measured drive cycles, these savings are 0.8% for the T1 concept, 0.9% for the T2 concept, and 2.2% for the T3 concept. When simulated on the randomly generated drive cycles, the savings are 2.1% for the T1 concept, 0.9% for the T2 concept, and 2.5% for the T3 concept. The T3 concept, with a higher degree of hybridization, benefits more from using position-dependent predictive control than the two concepts with low hybridization. For the T3 concept, the potential for position-dependent predictive control is higher when simulated on the stochastic model compared to the measurement. An explanation for this result is that in the stochastic model, power demands over 30 kW sligthly occur more often. This higher occurrence is due to that (1) is independent of θ, the slope of the road. At higher power demands, the EM needs to be activated. Thus, in order to fulfill the driver's request, energy need to be stored in the buffer, making the position-invariant controller more conservative than the ideal controller, which has perfect information about the future power demands. For the T3 concept, the ideal and the position-invariant controller is also compared by simulating on each measured drive cycle separately. The difference in final SoC here is weighted linearly to the fuel consumption. The weight is calculated using linear regression. The result of this comparison is shown in Fig. 7. The average relative increase using position-invariant control compared to ideal control is 2.9%, which means an average saving of 2.8%.

It is difficult to explain why, for the T1 concept, position-dependent predictive control is more important when simulated on the stochastic model than on the measurement.

Fig. 8 shows the SoC trajectories for the three concepts when simulated on one of the measured drive cycles. Studying the SoC trajectory for the T3 concept, it is shown that the planning is done on a long horizon, where energy is saved for the high-speed section, and the buffer is depleted at the two altitude peaks. This behavior is similar, but not as clear, for the T1 and T2 concept. A comparison between the ideal SoC trajectories for the T1 concept in Fig. 6 and in Fig. 8 shows that the entire buffer is used only when simulated on the quantized drive cycle. The reason for this difference is that the quantized drive cycle contains higher torque demands where the EM is forced to be activated. This forced activation means that it is more important to plan the use of the buffer. Fig. 8 shows that the position-invariant controller for the T1 and T2 concepts simply

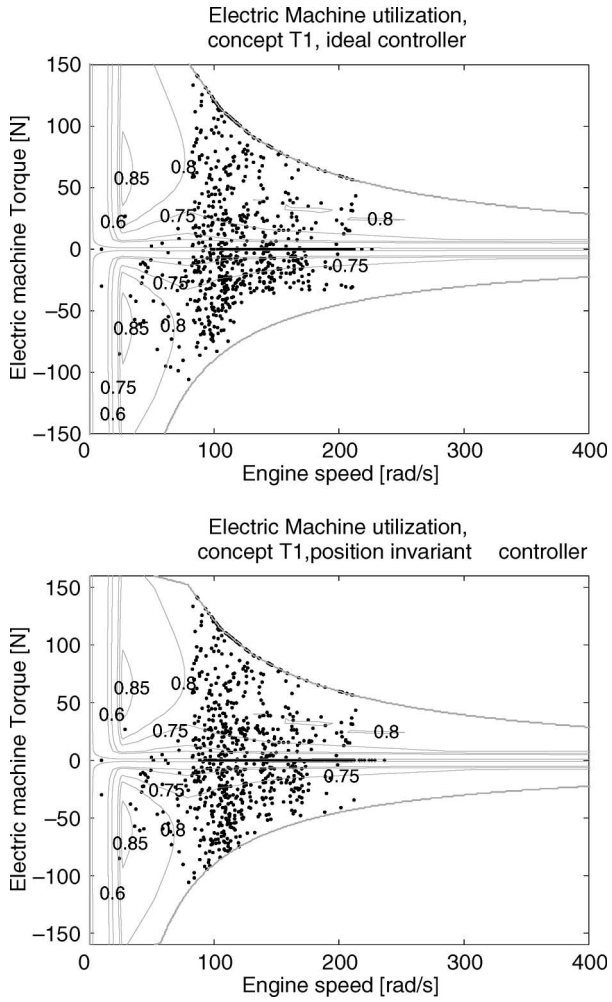

Fig. 12. EM-utilization concept T1. Negative torques means discharging. The concept has been simulated 1500 s on the stochastic model with the sample time of 1 s. Each dot is the occurrence of one operating point. The efficiency lines are the total buffer efficiency. The EM has low efficiency for low torques.

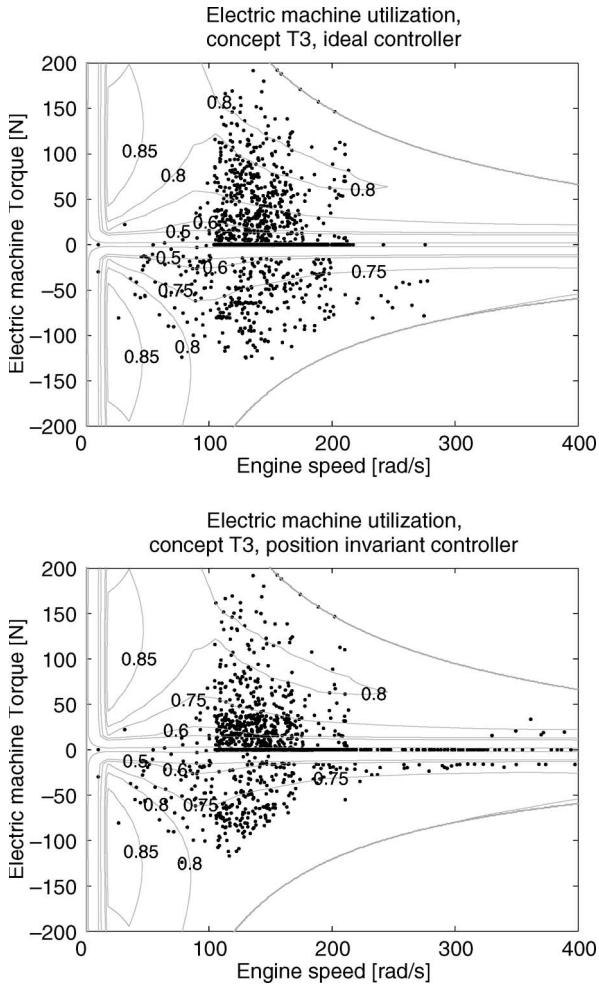

Fig. 13. EM-utilization concept T3. Negative torques means discharging. The concept has been simulated 1500 s on the stochastic model with the sample time of 1 s. Each dot is the occurrence of one operating point. The efficiency lines are the total buffer efficiency. The EM has low efficiency for low torques.

discharges the regenerated brake energy until the buffer is depleted. The roundtrip buffer efficiency is simply not high enough to motivate charging from the ICE and, since the ICE is powerful enough to propel the vehicle at all times, there is no danger if the SoC is low. For the T3 concept, the behavior is different. The position-invariant controller for the T3 concept has been derived with a small quadratic penalty for deviations from SoC $= 0.5$. The penalty is small enough not to affect the fuel consumption but big enough to ensure that the buffer is cycled around SoC $= 0.5$. If the quadratic penalty is zero for the T3 concept, the buffer is cycled around a slightly lower level with aggressive charging of the buffer when reaching low-SoC levels. The charging of the buffer then resembles an on–off strategy. Adding the quadratic penalty makes the usage of the buffer calmer.

Fig. 9 shows the SoC trajectories for the three concepts when simulated on a randomly generated drive cycle. Studying the SoC trajectory for the T3 concept, it is shown that energy is saved for the uphill sections, and the buffer is depleted at the two altitude peaks. This shows that the topography plays an essential part in the planning. Again, this behavior is similar but not as clear for the T1 and T2 concept.

Figs. 10 and 11 show that the ideal controller moves the operating points for the ICE slightly to higher efficiencies compared to the position-invariant controller. The operating points for the EMs are shown in Figs. 12 and 13. Note that the efficiency lines in the figures are the total buffer efficiency. It is shown that both the ideal controller and the position-invariant controller avoid discharging the buffer at lower efficiencies than 50%. The policy for the T3 concept is shown in Fig. 14. At 10 km/h, the vehicle is propelled by the EM except when the buffer is low.

VI. Realtime Implementation, One-Step MPC

By studying the results, an algorithm suitable for real-time implementation for the case of traveling along a given route is conceived. The comparison between the position-dependent controller and the ideal controller showed that if the information gathered on the 37 runs along the route was used optimally, ideal performance was attainable. The results also show that the planning is done on a long horizon and that the planning is insensitive to the fine structure of each drive cycle.

From this knowledge, the question then naturally arises: What can be achieved if we use information gathered from only one previously measured drive cycle to plan the use of

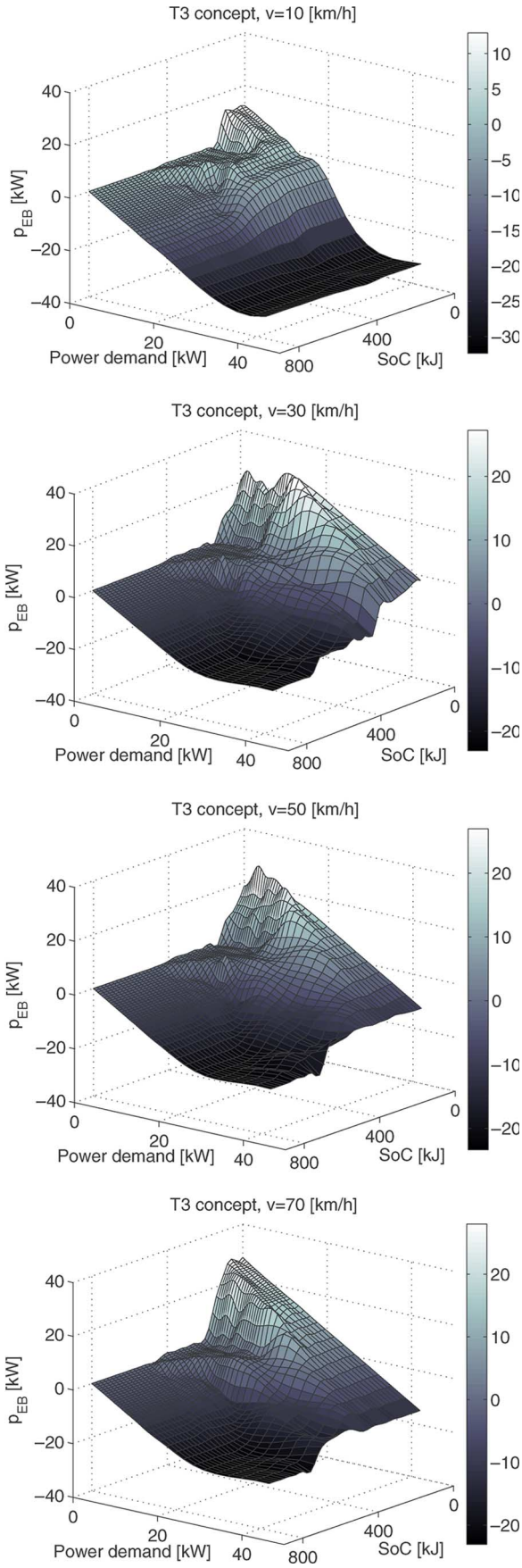

Fig. 14. Time-invariant policies concept T3. The graphs shows the EM activation for the velocities 10, 30, 50, and 70 km/h. Negative value means discharging.

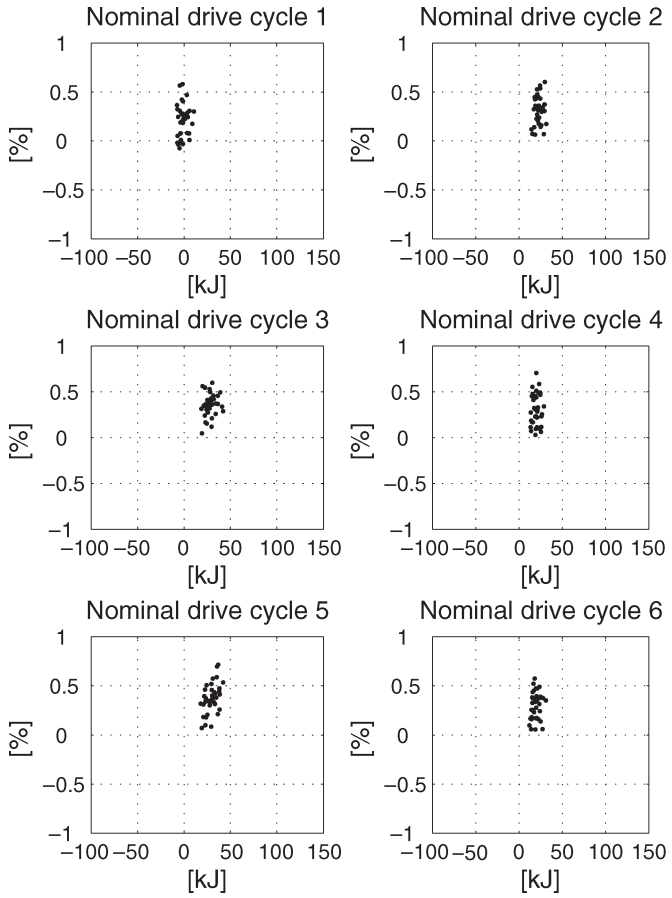

Fig. 15. Relative fuel consumption and the difference in final buffer energy compared to ideal control when using one-step MPC with a previously measured drive cycle along the route as nominal drive cycle. Six of the measured drive cycles are used as nominal drive cycles and evaluated on the rest of the measured drive cycles. The dots show the outcome for each of the measured drive cycles. The y-axis is the relative increase in fuel consumption compared to ideal control. The x-axis is the difference in final buffer energy compared to ideal control.

the buffer for the following runs? This drive cycle used for the planning is in the sequel referred to as the nominal drive cycle. The first step in the proposed algorithm is to calculate the ideal controller for the nominal drive cycle. The future cost at each time step $J(x_k)_k$ is found by backward iterations from the final cost $J(x_k)_N$. Remember that the dynamic state is simply $x_k = \text{SoC}_k$. The trick is now to connect the future costs J_k, $k = 1, 2, \ldots, N$ for the nominal drive cycle to their positions along the route and to store them on the onboard computer. When the route is driven the next time, the future costs from the nominal drive cycle will be used as the final cost in a one-step-MPC algorithm. At every sampling time, the control signal is thus calculated from

$$u = \arg\min_u \{c(\nu, p_{\text{demand}}, u) + J_k(\text{SoC}, u)\} \tag{21}$$

where $c(\nu, p_{\text{demand}}, u)$ is the instantaneous cost, and $J_k(\text{SoC}, u)$ is the stored future cost from the closest position k.

A. Results and Discussion

The algorithm is evaluated on all measured drive cycles using the T3 concept and six randomly chosen measured drive cycles

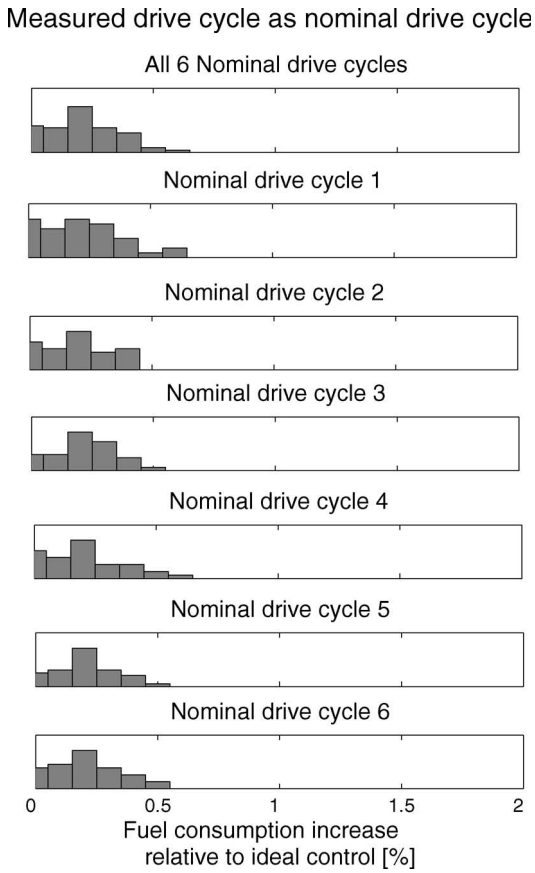

Fig. 16. Distribution of the relative increase in fuel consumption compared to ideal control when using one-step MPC with a previously measured drive cycle along the route as nominal drive cycle. Six of the measured drive cycles are used as nominal drive cycles and evaluated on the rest of the measured drive cycles. Since the final SoC differ from 0.5, the final buffer energy is weighted linearly to the fuel consumption. At the top, the results from all six nominal drive cycles are shown. The average relative increase is 0.3%. Results from each of the nominal drive cycles are shown below.

as nominal drive cycles. The ideal controller is calculated with a quantization step of 20 kJ. This calculation is done in MATLAB and takes 6 s on a 2.8 GHz computer. The future cost is saved every second and, in total, the planning requires 0.5 Mb of memory storage space. The simulation results are shown in Figs. 15 and 16. If the difference in final SoC is weighted linearly to the fuel consumption, the result is on average 0.3% higher consumption than ideal control. This means that the proposed algorithm achieves close to ideal performance using only one previously measured drive cycle. Moreover, the variance of the relative consumption is low with almost all outcomes achieving lower than 0.5% relative difference from the ideal control for all nominal drive cycles. A comparison with the ideal SoC trajectory is shown in Fig. 17. The SoC trajectories are very similar, and the deviations are small. However, at instants when the one-step-MPC trajectory is lower, the buffer is not charged from the ICE.

Calculation times and memory usage could probably be reduced further. For instance, the storage space can be reduced significantly since the future cost changes relatively slow along the route and need, therefore, not be stored at every second. However, even at this early stage, the algorithm is suitable for implementation in a real vehicle.

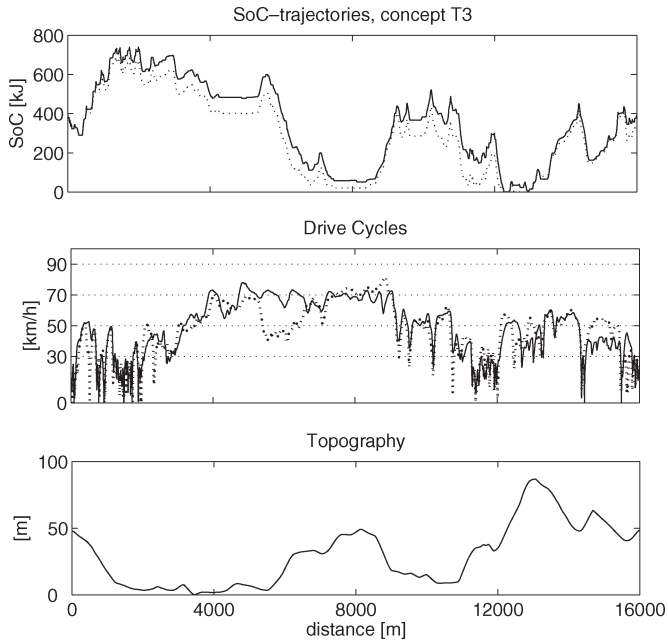

Fig. 17. SoC trajectories for the ideal and the one-step-MPC controller. In the upper graph, the solid line is the SoC trajectory for the ideal controller, and the dotted line is the SoC trajectory for the one-step-MPC controller. In the graph below, the dotted line is the nominal drive cycle, and the solid line is the drive cycle on which the simulation is conducted. The bottom graph shows the topography of the route. The one-step-MPC controller results in 0.2% higher consumption for the tested drive cycle.

The perfect application, although not studied here, would be a city bus traveling the same route. If it is found that the traffic conditions vary significantly, a nominal trajectory and an ideal controller for each of the conditions could be stored.

When using the navigator and a GPS, the nominal drive cycle for routes not yet traveled could be based on information such as speed limits and topography. When the driver inputs the destination in the navigator, a nominal drive cycle would then be provided and an ideal controller calculated. While waiting for the calculation of the ideal controller, the energy management is handled by the vehicle's standard controller. When the calculation is finished, the energy-management system switches over to the one-step-MPC algorithm. If the calculation of the ideal controller is done in 1 min for a 30 min route, an approach like this would mean that the one-step-MPC algorithm would be active 97% of the time.

VII. Conclusions and Future Work

The performance of the ideal and the position-dependent controller is almost identical. The conclusion is that the planning is done on a long horizon where the topography plays a crucial part. Furthermore, this also means that if position-dependent information is used in the EMS, the planning is insensitive to the particular fine structure of the drive cycle, only the "average" information about the route is significant. The comparison between the fuel consumption between the position invariant and the ideal controller shows that if the controller uses position-dependent information, the potential fuel savings is between 1% and 2.8% for the studied route and concepts. The concept with high degree of hybridization benefits more from

using position-dependent predictive control than the concepts with low hybridization. This paper, therefore, shows that good performance is attainable without position-dependent information, conditioned that the EMS is well tuned to the route. If position-dependent information is used, the fuel consumption is reduced further. However, the proposed one-step-MPC algorithm shows that with feedback of the position and the use of a single nominal drive cycle, the performance is only on average 0.3% from the ideal controller on the studied route. The variance of the relative consumption compared to the ideal controller is low, indicating a high robustness. An extensive examination of the robustness properties is a topic for future work. Measurement data on totally different drive cycles covering all types of driving will then be required.

The current results are promising and are an indication that further simplifications can be made. The nominal drive cycle used in the planning is a topic for further research. Relying on a previously measured drive cycle, as nominal drive cycle. is impractical and would restrict the usefulness of the one-step-MPC algorithm. Instead, it would be more practical if the planning could be based primarily on information that could be stored in a digital map.

The discrepancy from the results presented in [10] is due to that, in the previous paper, the evaluation of the controllers was done on a quantized drive cycle. The new approach taken in this paper guarantees a just evaluation.

ACKNOWLEDGMENT

The authors would like to thank A. Cindric for his help with the collection of the drive data.

REFERENCES

[1] J. Hellgren, "A methodology for the design of cost effective hybrid vehicles," Ph.D. dissertation, Chalmers Univ. Technol., Gothenburg, Sweden, 2004.

[2] E. Finkeldei and M. Back, "Implementing an mpc algorithm in a vehicle with a hybrid powertrain using telematics as a sensor for powertrain control," in *Proc. IFAC Symp. Advances Autom. Control*, Apr. 19–23, 2004, pp. 446–450.

[3] Y. Deguchi, K. Kuroda, M. Shouji, and T. Kawabe, "HEV charge/discharge control system based on car navigation information," in *Proc. JSAE Spring Conf.*, Yokohama, Japan, May 21–23, 2003, pp. 105–112.

[4] G. Grimmet and D. Stirzaker, *Probability and Random Processes*. London, U.K.: Oxford Univ. Press, 2004.

[5] I. Kolmanovsky, I. Siverguina, and B. Lygoe, "Optimization of powertrain operating policy for feasibility assessment and calibration: Stochastic dynamic programming approach," in *Proc. Amer. Control Conf.*, May 8–10, 2002, pp. 1425–1431.

[6] P. Rutquist, "Optimal control for the energy storage in a hybrid electric vehicle," in *Proc. 19th Int. Battery, Hybrid and Fuel Cell Electric Vehicle Symp. and Exhib. (EVS19)*, Oct. 2002, pp. 1133–1139.

[7] C.-C. Lin, P. Huei, and J. Grizzle, "A stochastic control strategy for hybrid electric vehicles," in *Proc. Amer. Control Conf.*, Jun. 30–Jul. 2, 2004, pp. 4710–4715.

[8] L. Johannesson, "Development of a time invariant stochastic model of a transport mission," in "Signals and Systems," Chalmers Univ. Technol., Gothenburg, Sweden, Tech. Rep. R023/2005, 2005.

[9] M. Puterman, *Markov Decision Processes*, ser. Wiley Series In Probability And Mathematical Statistics. Hoboken, NJ: Wiley, 1994.

[10] L. Johannesson, M. Åsbogård, and B. Egardt, "Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming," in *Proc. ITSC*, Vienna, Austria, Sep. 13–16, 2005, pp. 366–371.

Lars Johannesson received the M.Sc. degree from Chalmers University of Technology, Gothenburg, Sweden, in 2003, where he is currently working toward the Ph.D. degree at the Automatic Control Group at the Department of Signals and Systems.

His research interests involve hybrid electric-powertrain control, predictive control, and stochastic control.

Mattias Åsbogård received the M.Sc. degree from Chalmers University of Technology, Gothenburg, Sweden, in 2001. In 2005, he received the Lic.Eng. degree in mechatronics, specializing in the evaluation of hybrid electric-vehicle concepts.

After working with production systems within the mechanical industry, he returned to Chalmers University of Technology in 2003, where he joined the Department of Machine and Vehicle Systems. He is currently working with modeling and simulation of automotive powertrains with the Powertrain Attribute Center, Volvo Car Corporation, Gothenburg. His research interests involve configuration and component sizing of automotive powertrains, particularly hybrid electric powertrains.

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 at Information Systems Laboratory, Stanford, CA. 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 at Chalmers University of Technology, Gothenburg, Sweden. His main areas of interest include adaptive and hybrid control and applications of control in the automotive area.

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