

Adaptive Control Using Map-Based ECMS for a PHEV

Martin Sivertsson *

* *Vehicular Systems, Dept. of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, marsi@isy.liu.se*

Abstract: A plug-in hybrid electric vehicle(PHEV) is a promising way of achieving the benefits of the electric vehicle without being limited by the electric range. This paper develops an adaptive control strategy based on a map-based ECMS approach. The control is developed and implemented in a simulator provided by IFP Energies nouvelles for the PHEV benchmark. The implemented control strives to be as blended as possible, whilst still ensuring that all electric energy is used in the driving mission. The controller is adaptive to reduce the importance of correct initial values but since the initial values affect the consumption a method is developed to estimate the optimal initial value for the controller based on driving cycle information. This is seen to work well for most driving cycles with promising consumption results. The controller also fulfills all requirements set by the PHEV Benchmark.

Keywords: Optimal Control, Plug-In Hybrid Electric Vehicles, Equivalent Consumption Minimization Strategy

1. INTRODUCTION

A hybrid electric vehicle(HEV) utilizes both electric energy and energy from fuel to meet the demands set by the driver. This may lead to a reduction in environmental impact and fuel consumption of the vehicle. A Plug-In HEV(PHEV) is a HEV with possibility to recharge the battery from the grid. This adds the potential of using the vehicle as an electric vehicle, without the range limitations in a pure electric vehicle. The supervisory control algorithm for these more complex powertrains play an important role in realizing the full potential of the powertrain. In order to evaluate different strategies the IFP Energies nouvelles(IFPEN) is organizing a benchmark for a PHEV, see PHEV Benchmark Rules (2012).

This paper extends the adaptive map-based Equivalent Consumption Minimization Strategy(ECMS) developed in Sivertsson et al. (2011) to the PHEV problem and it is implemented for the simulator made available in the PHEV benchmark. The optimal torque distribution and generator speed is calculated offline and stored in tables. Then an online control is developed which fulfills the requirements of the benchmark for reasonable initial values and finally a strategy for estimating the optimal initial value is developed that together with the adaptive control fulfills the benchmark requirements.

2. IFPEN PHEV BENCHMARK

In the IFPEN PHEV Benchmark a simulator is provided for which a supervisory control algorithm is to be designed. This simulator is a quasi-static model of the Chevrolet Volt with vehicle and battery dynamics and all energy converters modeled using stationary maps. The Chevrolet Volt has three energy converters, internal combustion engine(ENG), electric motor(EM), and generator(GEN),

connected through a planetary gearbox(GB). Both electric machines can work in both motoring and generating mode. The powertrain also incorporates three clutches that allows the vehicle to be driven in the following four modes:

- Mode 1: One motor pure electric vehicle. Only the EM is connected to the GB.
- Mode 2: Two motor pure electric vehicle. Both the EM and GEN are connected to the GB.
- Mode 3: Series HEV. Only the EM is connected to the GB. The ENG and GEN works as an auxilliary power unit, producing electric power.
- Mode 4: Power split HEV. All energy converters are connected to the GB.

In the benchmark the controller should output desired torque from the ENG, EM, and mechanic brakes, the speed of the GEN, the position of the three clutches, and if the engine should be on or off. The inputs to the controller are the requested torque from the driver model, T_{req} , minimum allowed regenerative torque, battery state of charge(SOC), vehicle speed, average speed in the driving cycle, v_{avg} , and approximate driving cycle length, D_{tot} . The aim of the benchmark is to minimize the criteria described in Table A.1 with a battery that is fully charged at the beginning of the driving cycle and may be depleted at the end of the driving cycle. There are also rules on how close the controller has to follow the desired velocity profile, see PHEV Benchmark Rules (2012).

2.1 Models

The models implemented in the simulator are briefly described below. For more details see PHEV Benchmark Rules (2012).

Vehicle Model The vehicle motion equation is implemented as (1) where T_{wh} is the torque from the powertrain at the wheels and T_b is the torque applied by the brakes.

$$\frac{dv}{dt} = \frac{r_{wh}}{J_{veh}} (T_{wh} - T_b - r_{wh}(mg \sin \theta + c_0 + c_1 v + c_2 v^2)) \quad (1)$$

Battery Model The battery model is of equivalent circuit type and implemented as:

$$I_b = \frac{U_{oc}}{2R_c} - \sqrt{\frac{U_{oc}^2 - 4R_c P_b}{4R_c^2}} \quad (2)$$

$$P_{ech} = I_b U_{oc} \quad (3)$$

$$\Delta SOC = -dt \frac{I_b}{Q_0} \quad (4)$$

Transmission The transmission is a planetary gear set with three clutches, c_1 , c_2 , and c_3 . The kinematic relations between the energy converters and the wheels are:

$$\omega_{ENG} = \omega_{GEN} c_3 \quad (5)$$

$$\omega_{EM} = \omega_{wh} \gamma_{fd} (1 + \gamma_{rs}) - \omega_{GEN} c_2 (1 - c_1) \gamma_{rs} \quad (6)$$

$$T_s = T_{EM} \eta_{GB}^{\text{sgn}(T_{EM})} \quad (7)$$

$$T_r = T_s \gamma_{rs} c_2 \eta_{GB}^{\text{sgn}(T_s)} \quad (8)$$

$$T_{GEN} = T_r - c_3 T_{ENG} \quad (9)$$

$$T_c = (1 + \gamma_{rs}) \left((1 - c_2) T_s + \frac{c_2}{\gamma_{rs}} (T_{GEN} + c_3 T_{ENG}) \right) \quad (10)$$

$$T_{wh} = T_c \eta_{GB}^{\text{sgn}(T_c)} \gamma_{fd} \quad (11)$$

Consumption There are two consumptions provided in the simulator and used in this paper, fuel consumption and a fuel equivalent of the electricity consumption. They are defined as:

$$m_f = \frac{\int \dot{m}_f}{\rho_f D_{real}} \quad (12)$$

$$m_{f,equiv} = \frac{\int P_{ech}}{\eta_{avg} q_{LHV} \rho_f D_{real}} \quad (13)$$

Where \dot{m}_f is the fuel flow, ρ_f the density of the fuel, D_{real} is the distance travelled, P_{ech} the electrochemical power, η_{avg} the average efficiency from fuel to electricity, and q_{LHV} is the lower heating value of the fuel.

3. PROBLEM FORMULATION

Looking at the scoring metrics and CO_2 data in Table A.1-A.2 the problem can be reformulated as delivering the torque requested by the driver, or as close as possible if the requested torque is infeasible, in a fuel and computationally efficient way. Even though the CO_2 emissions are higher for electricity production, the higher efficiencies of the electric energy converters compared to the efficiency of the combustion engine, results in that the minimization of the well-to-wheel CO_2 emissions can be interpreted as fuel consumption minimization. So the aim is to minimize the energy use, with emphasis on the fuel consumption, while fulfilling the driver requests. This problem is well represented by the equivalent consumption minimization strategy (ECMS) where the sum of fuel and battery power is minimized. However, battery and fuel power aren't directly comparable and therefore an equivalence factor λ relating the two is needed, for more information on

ECMS see Paganelli et al. (2002); Sciarretta et al. (2004); Musardo and Rizzoni (2005). The problem is formulated as:

$$\min (H = P_f + \lambda P_{ech}) \quad (14)$$

$$[T_{EM}, T_{ENG}, \omega_{GEN}, Mode] = \text{argmin}(H) \quad (15)$$

Subject to:

$$\begin{aligned} T_{wh} &= T_{req} \\ T_{min}(\omega) &\leq T \leq T_{max}(\omega) \\ 0 &\leq \omega \leq \omega_{max} \\ P_{b,min}(SOC) &\leq P_b \leq P_{b,max}(SOC) \end{aligned} \quad (16)$$

Where the torque and speed limits are applied to each individual energy converter.

4. OFFLINE OPTIMIZATION

Since the kinematic relations change with the actuation of the clutches the optimization problem to be solved differs between the modes. Due to the difficult nature of the problem the optimal solution is not calculated online. Instead in order to find which mode to use when, the minimum cost for each mode is calculated offline and stored in tables for a given set of T_{req} , ω_{wh} , and λ . The SOC is found to only have minor effects on the optimal solution, therefore that effect is ignored. To ensure that (16) are all fulfilled, or in the case of $T_{req} = T_{wh}$, the produced torque is as close to the requested as possible for that mode, the cost function in (14) is augmented so that the closest point, that fulfills all the inequalities, is selected. In order to find which mode is optimal for each combination of T_{req} , ω_{wh} , and λ , the optimal torque and speed setpoints also has to be found. But instead of just storing all the control variables in tables a few insights can be gained from the kinematic relations in (5)-(11) to reduce the amount of memory used:

- T_{EM} can be calculated from T_{req} in all modes. Therefore Mode 1 requires no tables.
- Mode 2: Only ω_{GEN} has to be stored and since $P_f = 0$ it is independent of λ
- Mode 3: T_{EM} , ω_{EM} , and therefore P_{EM} are given by T_{req} and ω_{wh} . The optimal output power from the generator should be on the optimal operating line of the engine-generator combination (GENSET). Therefore only the optimal output power for each P_{EM} , λ combination has to be stored together with the optimal operating line of the GENSET.

This results in 7-tables to be stored, shown in Fig. 1. That is:

- Mode-selection (3-D)
- Mode 2: ω_{GEN} (2-D)
- Mode 3: P_{GENSET} (2-D), $\omega_{opt-line}$ (1-D), and $T_{opt-line}$ (1-D)
- Mode 4: ω_{GEN} (3-D), and T_{GEN} (3-D)

The optimization is performed for a rather dense grid in T_{req} , ω_{wh} , and λ . In order to minimize the amount of memory used the T_{req} , ω_{wh} , or λ resulting in the smallest error in the interpolation scheme used in the online implementation if removed, is removed in an iterative manner. This is performed for all tables, so each table has its own discretization. To simplify the implementation and

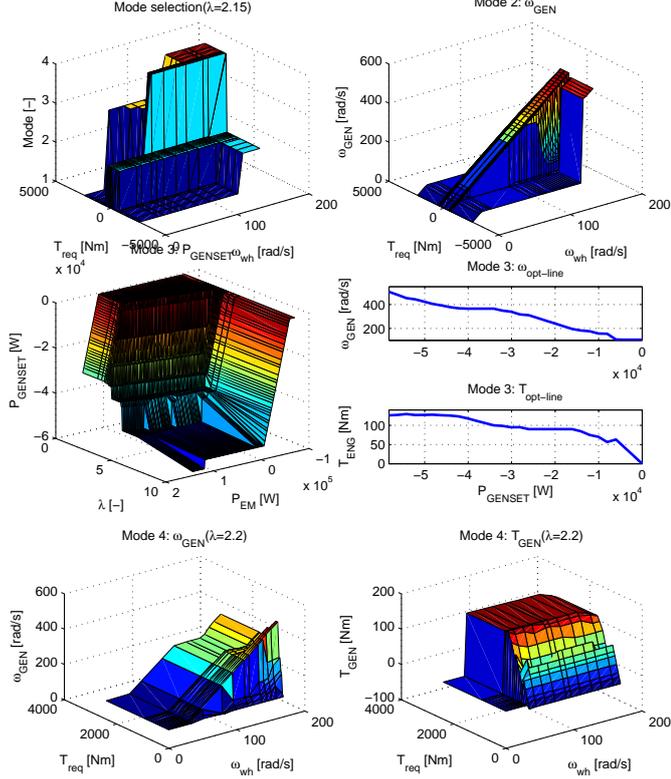


Fig. 1. Structure of the stored data

to reduce the memory consumption Mode 4 is only used when $T_{req} > 0$.

5. CONTROLLER

The outline of the controller is shown in Fig. 2. The controller consists of three main subsystems. The first subsystem calculates the value of the equivalence factor, λ , using the SOC and driving cycle data. The second subsystem controls which mode to engage and the third calculates the torque and speed setpoints for the energy converters.

The modes block consists of five subsystems, one for each mode and one for engine start. The mode controller outputs which mode to activate and if the engine should be started or not. In order to avoid too frequent engine starts/stops two thresholds are used, t_{on} and t_{off} . The controller has to try to turn the engine on/off for a duration longer than t_{on}/t_{off} before it is turned on/off. The torques and speed are then calculated using the tables calculated offline and the kinematic relations for that mode defined in (5)-(11). Care is also taken not to exceed any of the constraints in (16).

6. ENERGY MANAGEMENT

The energy management of a PHEV can be divided into two categories. The first is to make use of all the stored energy in the battery, that is run as an electric vehicle until the SOC is under a certain limit, and then operate as a hybrid in charge sustaining mode. This strategy is commonly denoted charge deplete-charge sustain strategy (CDCS). The main advantage of this strategy is that it

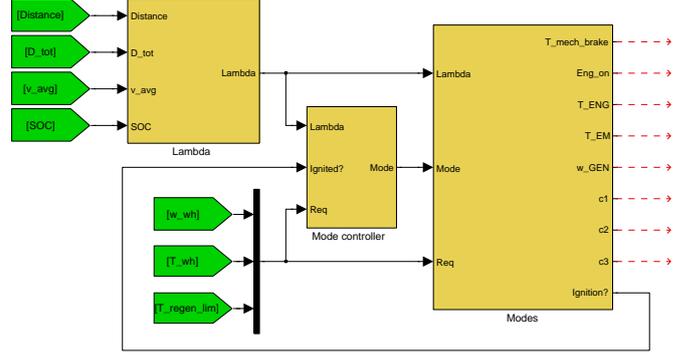


Fig. 2. Structure of the controller. The controller consists of three main subsystems, one where the equivalence factor is calculated, one where the mode is selected, and one where the torque and speed setpoints are calculated.

is guaranteed to make use of the stored electric energy and does not need information about the future driving mission. The second strategy is to mix usage of fuel and electricity throughout the driving cycle, a strategy known as blended strategy. It is well established in the literature that a blended strategy may result in lower fuel consumption than CDCS, see for instance Larsson et al. (2010). However, in order for a blended strategy to make use of all the energy in the battery the length of the driving cycle has to be known. In the driving cycles provided by the PHEV Benchmark organizers only the approximate distance as well as the mean speed is known. In the provided driving cycles this approximate distance can deviate from the actual distance of the driving cycle by up to almost 10%.

6.1 Equivalence factor adaptation

In order to make use of all the stored energy in the battery, a mix between the blended and CDCS strategies is implemented. The strategy is to undershoot the approximate distance by 10%, and use that as a horizon for the blended strategy. If the actual distance is longer than that used for the blended strategy, the control goes over into charge sustaining mode. This is achieved by setting a SOC reference, SOC_c that is linear in ratio of travelled distance vs. expected distance, see Fig. 3, a method also used in Tulpule et al. (2009). The minimum SOC_c is set to 0.315 in order to ensure that $SOC(end) \geq 0.3$. The strategy used in Sivertsson et al. (2011) is then extended to fit the PHEV problem. The strategy is to adapt the equivalence factor according to a tangent function in SOC . The idea is that as long as the SOC is near the desired SOC the control should remain rather constant. But when the SOC approaches the limits the control needs to adapt. In Sivertsson et al. (2011) this is used in a HEV where the aim is to maintain the SOC around a constant level. Here, since it is a PHEV, it is desirable to use the energy stored in the battery, therefore the center of the tan-function is SOC_c . The used SOC -window is also decreased linearly with distance traveled. This is to allow larger deviations early in the driving mission, and then make the control follow the SOC_c narrower towards the end of the driving cycle. The λ -adaptation is given by (17) where l_1 and l_s are

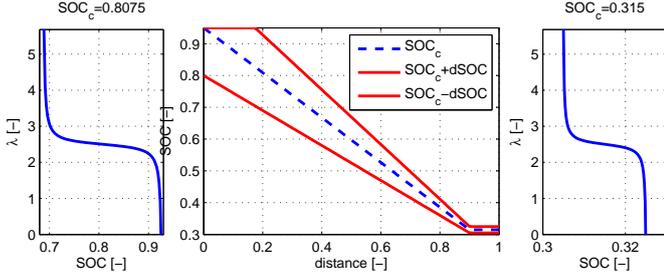


Fig. 3. The outline of the basic control shape

Table 1. The change in consumptions compared to $\lambda_{c,opt}$ for different values of λ_c . All controls ensure $SOC(end) \geq 0.3$.

| Cycle-info | λ_c | Δm_f [%] | $\Delta m_{f,equiv}$ [%] | $SOC(end)$ |
|----------------------|------------------------|------------------|--------------------------|------------|
| 10xFUDS | 3 | -0.65 | -0.19 | 0.3099 |
| $D_{tot} = 119.9km$ | $\lambda_{c,opt}=2.65$ | - | - | 0.3076 |
| $D_{real} = 119.9km$ | 2 | 10.35 | 2.32 | 0.3066 |
| 10xNEDC | 3 | 27.02 | 15.87 | 0.3420 |
| $D_{tot} = 119.9km$ | $\lambda_{c,opt}=2.63$ | - | - | 0.3197 |
| $D_{real} = 110.1km$ | 2 | 0.80 | 0.56 | 0.3162 |
| 10xUS06 | 3 | -0.03 | -0.71 | 0.3147 |
| $D_{tot} = 119.9km$ | $\lambda_{c,opt}=2.85$ | - | - | 0.3103 |
| $D_{real} = 128.9km$ | 2 | 5.30 | 2.20 | 0.3078 |

constants that control the slope and range of the tangent function, and $dSOC$ is the allowed deviation from SOC_c .

$$\lambda = \lambda_c - l_1 \tan\left(\frac{l_s \pi}{2dSOC}(SOC - SOC_c)\right) \quad (17)$$

A benefit with this formulation is that the smaller $dSOC$ is, the steeper the slope around SOC_c becomes and the faster the control reacts to deviations. In Fig. 3 the shape of the control is shown for the case when the approximate distance is correct. That is the SOC_c undershoots the distance traveled, and thus results in the control going over to charge sustaining mode. Also shown is that the allowed SOC deviation gets smaller with distance. The variable λ_c still has to be decided. In Table 1 the change in consumptions compared to the consumptions with optimal λ_c and end SOC are shown for different values of λ_c and different driving cycles. A λ_c is considered optimal if the λ trajectory follows SOC_c . But since SOC_c is based on undershooting the approximate driving cycle length, there are λ_c values that result in lower consumptions, this is however hard to predict. It is seen that the optimal value changes with the driving cycle. The control ensures $SOC(end) \geq 0.3$ for all λ_c but it might come with a substantial increase in consumption if the λ_c value is wrong.

In Fig. 4 the λ and SOC trajectories for the different values of λ_c on the US06 driving cycle is shown. Due to the driving mission length provided only being approximate, the control undershoots the length in order to make sure all electric energy is used. The US06 cycle is however 7% longer than the length provided, resulting in an undershoot of roughly 16% for the controller. It is seen that the control for $\lambda_c \neq \lambda_{c,opt}$ does not follow the SOC_c , instead it follows $SOC_c \pm dSOC$. For $\lambda_c = 2$ this results in a control that switches rapidly between $\lambda \approx 2.5$ and $\lambda \approx 5$, something that comes with a large consumption penalty.

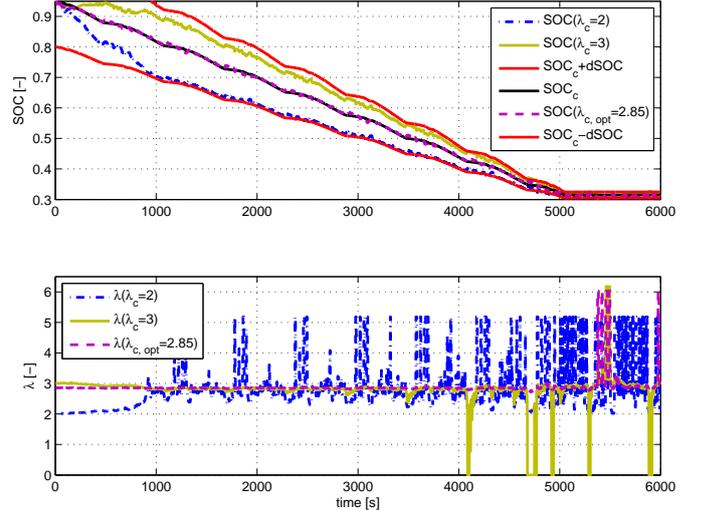


Fig. 4. The SOC and λ -trajectories for different λ_c values on the US06 driving cycle. A poor λ_c may lead to a switching characteristic of the control.

Table 2. The change in consumption with adaptive λ_c for different $\lambda_{c,init}$, compared to $\lambda_{c,opt}$.

| Cycle-info | $\lambda_{c,init}$ | Δm_f [%] | $\Delta m_{f,equiv}$ [%] | $SOC(end)$ |
|----------------------|-----------------------------|------------------|--------------------------|------------|
| 10xFUDS | 3 | 0.10 | -0.29 | 0.3110 |
| $D_{tot} = 119.9km$ | $\lambda_{c,init,opt}=2.65$ | -0.57 | -0.05 | 0.3075 |
| $D_{real} = 119.9km$ | 2 | -0.87 | -0.16 | 0.3081 |
| 10xNEDC | 3 | 15.61 | 0.45 | 0.3306 |
| $D_{tot} = 119.9km$ | $\lambda_{c,init,opt}=2.55$ | -0.16 | 0.08 | 0.3188 |
| $D_{real} = 110.1km$ | 2 | 1.69 | 0.45 | 0.3184 |
| 10xUS06 | 3 | 0.07 | -0.41 | 0.3143 |
| $D_{tot} = 119.9km$ | $\lambda_{c,init,opt}=2.85$ | 0.03 | -0.18 | 0.3109 |
| $D_{real} = 128.9km$ | 2 | 0.39 | -0.05 | 0.3101 |

6.2 Adaptive control of λ_c

In order to avoid the switching nature of the λ -control seen in Fig. 4 the idea is to adapt λ_c if the SOC deviates too much from SOC_c . This is done with a PI-controller according to:

$$\lambda_c = \lambda_{c,init} + K_p(SOC_c - SOC) + K_i \int (SOC_c - SOC) dt \quad (18)$$

The values of K_p and K_i control how fast the controller adapts, but a faster controller comes with a slight consumption penalty. In Table 2 the consumption change compared to $\lambda_{c,opt}$ without adaptive λ_c is shown for different driving cycles and $\lambda_{c,init}$. A $\lambda_{c,init}$ is considered optimal if it roughly produces a SOC trajectory that follows the desired trajectory without λ_c deviating too far from $\lambda_{c,init}$. It is seen that the adaptive λ_c performs as well as $\lambda_{c,opt}$, better in some cases, worse in some cases. But most of all it reduces the effect of poor initial values. This is also confirmed in Fig. 5 where the SOC and λ -trajectories are shown for the US06 driving cycle. The switching nature is almost completely removed, resulting in a near constant λ value during the entire blended phase.

6.3 Estimating $\lambda_{c,init}$

Even if the developed control has been seen to perform well for all reasonable initial λ_c , the consumption is still affected by it. Therefore it is desirable to achieve an estimate as close as possible to the optimal λ_c . In Fig. 6,

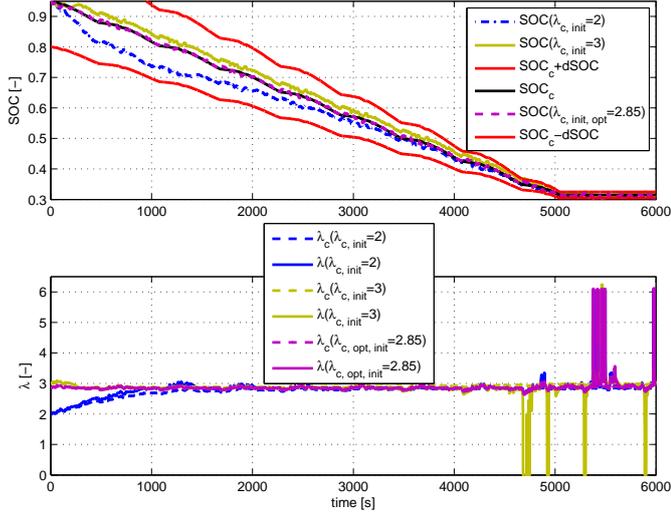


Fig. 5. The SOC and λ -trajectories with adaptive λ_c for different values on $\lambda_{c,init}$ on the US06 driving cycle. The adaptive λ_c reduces the effect of poor initial values.

the optimal $\lambda_{c,init}$ is plotted against approximate distance for the driving cycles used. It is seen that the shape of the profiles is similar for all driving cycles. The all electric range, that is the distance for $\lambda_{c,init} > 2$, differs up to almost 100% for the different driving cycles. In Fig. 7 the approximate distance required to exceed the all electric range is plotted against mean speed. Even if the mean speed isn't enough to describe the driving cycle, since neither slope nor how transient it is is captured by the mean speed, the all electric range is approximated by a linear function, shown in Fig. 7. Artemis Urban is plotted in magenta to mark that it is considered an outlier and is not included when the line is fitted. Since the losses in the vehicle motion equation (1) are quadratic in speed, a straight forward assumption would be that the all-electric range decreases with mean speed, an assumption that is also used here. The approximate distance is then corrected with the proposed linear correction, in order to compensate for the different all-electric ranges. The result is shown in Fig. 8. It is seen that the correction shifts the points to the same region, a trend that is well captured by an exponential function. The final scheme to estimate $\lambda_{c,init}$ is of the form:

$$D_{corr} = D_{tot} - (k_1 v_{avg} + k_2) \quad (19)$$

$$\lambda_{c,init} = k_3 (1 - \exp(-k_4 D_{corr} + k_5)) \quad (20)$$

In Table 3 the results for the full controller with $\lambda_{c,mod}$ is compared to the results for $\lambda_{c,opt}$. Even if the estimated $\lambda_{c,init}$ is not too far from the optimal, the consumption can differ substantially. Interesting to note is that the driving cycle with the largest $\lambda_{c,init}$ error shows the best result. Looking at Fig. 9 this appears to be due to that the $\lambda_{c,mod}$ -control has a higher λ value when entering charge sustaining-mode which results in a less switching behaviour and lower consumption.

Another important property of the $\lambda_{c,init}$ estimation is that it should be such that it avoids unnecessary engine starts if the driving mission is within the all electric range. This is achieved for all tested driving cycles except

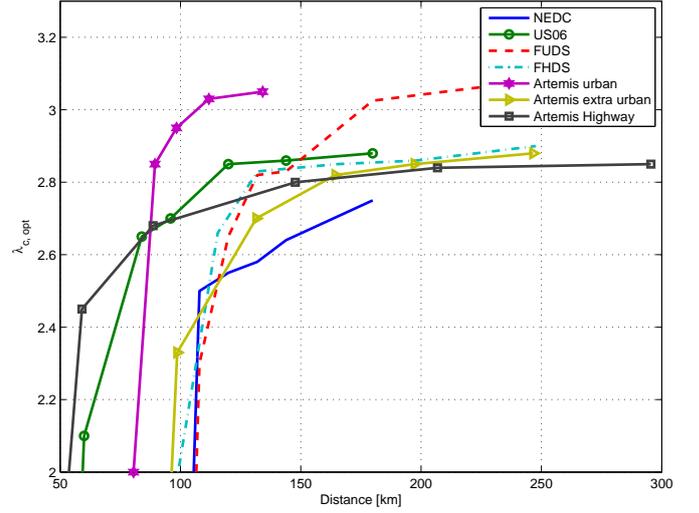


Fig. 6. The optimal $\lambda_{c,init}$ as a function of approximate distance for different driving cycles. The general shape of the curves are similar, however the all-electric range differs between the cycles.

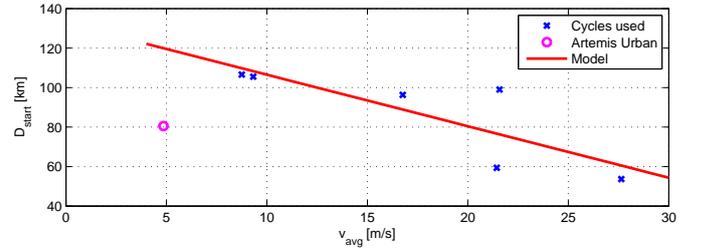


Fig. 7. The approximate distance required to exceed the all-electric range for different driving cycles and a linear model to capture the behaviour. Artemis Urban is considered an outlier and is not included when the line is fitted.

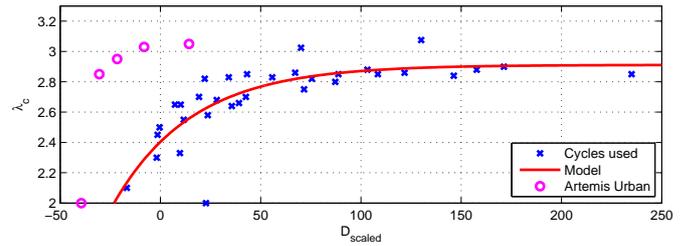


Fig. 8. $\lambda_{c,init,opt}$ vs. corrected approximate distance and how it is modeled. Artemis Urban is considered an outlier and is not included when the curve is fitted.

Table 3. The change in consumption with modeled $\lambda_{c,init}$, compared to $\lambda_{c,opt}$.

| Cycle-info | $\lambda_{c,init,mod}$ | Δm_f [%] | $\Delta m_{f,equiv}$ [%] | $SOC(end)$ |
|---|------------------------|------------------|--------------------------|------------|
| 10xFUDS($\lambda_{c,init,opt}=2.65$) | 2.518 | 13.62 | 2.32 | 0.3066 |
| 10xNEDC($\lambda_{c,init,opt}=2.55$) | 2.533 | 0.80 | 0.56 | 0.3162 |
| 10xUS06($\lambda_{c,init,opt}=2.85$) | 2.741 | 5.30 | 2.20 | 0.3078 |
| 20xArtemis Urban($\lambda_{c,init,opt}=2.84$) | 2 | -6.69 | -1.41 | 0.3104 |

FHDS (1 unnecessary start) and Artemis Extra-Urban (2 unnecessary starts), which is deemed acceptable.

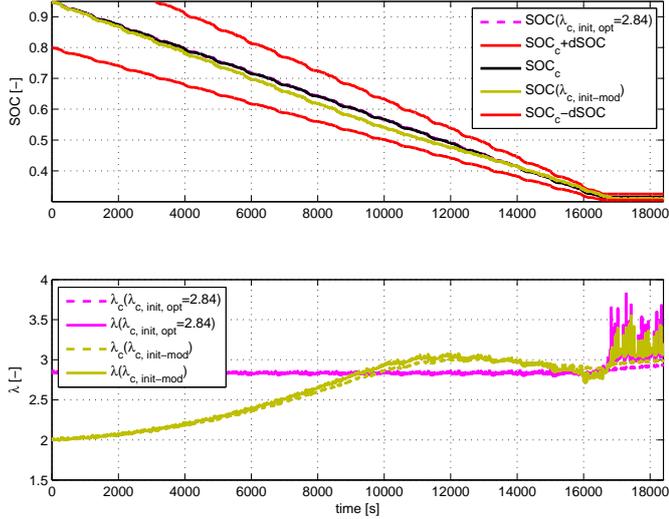


Fig. 9. SOC and λ trajectories for the full controller, with $\lambda_{c,init,mod}$, compared to the $\lambda_{c,init,opt}$. The modeled $\lambda_{c,init}$ is quite far from the optimal but still $SOC(end) \geq 0.3$.

Table A.1. The scoring metrics used in the benchmark

| | Metric | Weight |
|---------------------------------|---|--------|
| Performance(30%) | Acceleration 0-100km/h [s] | 7.5% |
| | Acceleration 70-120km/h [s] | 7.5% |
| | Acceleration 0-1000m on 4% slope [s] | 7.5% |
| | Braking Distance from 100km/h [m] | 7.5% |
| Energy and Economy(50%) | Total energy use (fuel+electricity) [MJ/km] | 15% |
| | Fuel consumption [MJ/km] | 20% |
| | Well-to-wheel CO_2 emissions [kg/km] | 15% |
| Computational performance (20%) | Processor use [simulation time] | 10% |
| | Memory use [MB] | 10% |

7. CONCLUSIONS

An adaptive map-based implementation of ECMS is developed and implemented for the IFPEN PHEV benchmark problem. The control strives to be as blended as possible, but still ensuring that all electric energy is used. The control tries to follow a SOC reference that is linear in traveled distance, but to ensure that all electric energy is used this distance is underestimated. The equivalence factor is adapted according to a function in SOC , a function whose center adapts according to how well the SOC reference is followed. Finally a method for estimating the initial value for the controller from driving cycle data is developed.

The resulting controller fulfills all the requirements of the benchmark. $SOC(end) \geq 0.3$ for all tested cycles, and the velocity profile error is within the tolerances prescribed by the benchmark. The control is also seen to adapt despite an initial estimate different from the optimal value. It also avoids unnecessary starts for most tested driving cycles that are within the all electric range. The controller provides a fuel economy that is within [-6, 13]% of what is considered optimal and is implemented in such a way that it only needs approximate driving mission length and average velocity.

Table A.2. Data for CO_2 emissions

| | |
|--|-------------------------------------|
| Gasoline well-to-tank emissions | 12.5g CO_2 /MJ of fuel |
| Gasoline combustion | 73.4g CO_2 /MJ of fuel |
| Electricity production(Europe average) | 94.7g CO_2 /MJ of electric energy |

Appendix A. BENCHMARK DATA

REFERENCES

- Larsson, V., Johannesson, L., and Egardt, B. (2010). Impact of trip length uncertainty on optimal discharging strategies for phev. In *IFAC Symposium on Advances in Automotive Control*. Munich, Germany.
- Musardo, C. and Rizzoni, G. (2005). A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management. In *IEEE Conference on Decision and Control and the European Control Conference*, 44, 1816–1823.
- Paganelli, G., Delprat, S., Guerra, T., Rimaux, J., and Santin, J. (2002). Equivalent consumption minimization strategy for parallel hybrid powertrains. In *IEEE Conference on Vehicular Technology*, 55, 2076–2081.
- PHEV Benchmark Rules (2012). ”<http://www.ecosm12.org/sites/ecosm12.org/files/PHEV%20benchmark%20rules.pdf>”. Downloaded 2012-07-18.
- Sciarretta, A., Back, M., and Guzzella, L. (2004). Optimal control of parallel hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 12(3), 352–363.
- Sivertsson, M., Sundström, C., and Eriksson, L. (2011). Adaptive control of a hybrid powertrain with map-based ECMS. In *IFAC World Congress*. Milano, Italy.
- Tulpule, P., Marano, V., and Rizzoni, G. (2009). Effects of different PHEV control strategies on vehicle performance. In *American Control Conference*. St. Louis, USA.