Battery Lifetime Emulator for a Hybrid Electric Vehicle

Modeling and Implementation

Master of Science Thesis in Systems Control and Mechatronics, SSYX04

CORIU OCTAVIAN
ALDO CHABLE

Department of Signals and Systems
Division of Automatic Control, Automation and Mechatronics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2012
Report No. EX040/2012

Abstract

For hybrid, plug-in hybrid and electric cars, battery lifetime is an important characteristic to study for car manufacturers. An adequate monitoring is necessary not only for maintenance planning (based on information about battery degradation over time), but also to provide the vehicle's hybrid strategy with proper information with which to make smarter decisions as to the battery use.

Test procedures and monitoring can be performed on laboratory conditions, but their region of validity has some limitations; it does not take external conditions (temperature, vibration, etc.) into account, and also cannot tell how the driver affects the battery lifetime. Therefore, laboratory tests are not sufficient, and finding a way to study actual driving cycles becomes necessary. Real-world data is a necessary step to study the battery lifetime, but producing test vehicles with hybrid drivetrains is costly.

This work presents an alternative to these test vehicles for analysing the battery usage influence on the lifetime of the battery. The hybrid components of a Hybrid Electric Vehicle (HEV), as well as the strategy that regulates their operation, are modelled in the MATLAB/Simulink software (components and strategy), and a method is introduced to estimate the battery cell's state of charge (SOC). The model and the estimation algorithm are then tested against several driving cycles, and the results are presented and discussed.

The idea is to insert a piece of software into an Electronic Control Unit (ECU). The hardware will be mounted on a "normal" car with an internal combustion engine; the software will collect information from the car to emulate the HEV operation and will transmit commands to a single battery cell via a Battery Management Unit (BMU). The battery cell will be discharged and charged with a current, as if it was placed in a battery system in a hybrid vehicle. By monitoring the battery usage and the degradation of the cell, knowledge of the battery lifetime is collected.

Acknowledgements

We want to thank Erik Durling at Volvo Cars for his constant support in all technical matters, his insights were at times crucial to carry on the work. Thanks also to Patrik Larsson and the thesis supervisor at Volvo, Theresa Granerus. Their constant input on the methods used were valuable to make this project solid, and they were also of great help during the learning period at the company. Additional thanks to Niklas Smith for hardware support during the implementation phase of the thesis. Finally, many thanks to our supervisor at Chalmers, Jonas Fredriksson. His continued interest in maintaining rigor in the thesis helped to make the project a better contribution to ongoing work, both at the company and at Chalmers.

Coriu Octavian. I want to thank Mihaela and my family for their constant support.

Aldo Chable. I want to thank my parents, Margot and Miguel, for their support and valuable feedback, and did not hesitate to point out mistakes, making this a better work than it would have been. I would also like to thank Delia, who in solidarity endured the longest hours of this project with me.

Preface

The work presented in this thesis was carried out at Volvo Cars Corporation, from them all the information regarding components and their parameters was obtained.

Contents

List of Figures

1 Introduction

The lifetime of the batteries used in hybrid electric vehicles (HEV) has a complex aging process and is therefore hard to predict. The aging process depends, among other factors, on the charge/discharge rates and the environmental conditions such as mechanical vibrations and temperature variation. Therefore it is necessary to perform tests on the batteries in order to understand how they are ageing. These tests are mostly performed in the lab environment with specific or standardized test cycles which are intended to simulate the actual usage. One of the main drawbacks with the uncertainty in the battery lifetime estimation is that it can influence the battery service plan in a negative way, by adding more costs, [1].

In an ideal case, the battery would have the same life span as the vehicle, resulting in no cost for the customer; it is therefore necessary to understand how real-life driving influences the battery's ageing, then a system can be designed so that the expected lifetime matches the desired lifetime.

Real life tests can usually be performed only with full-scale HEV in a late development phase. For this reason, there is a need for a test equipment which can be used early in the development phase and is less costly, less complex. In this thesis, a cost-efficient method for performing real life HEV battery tests in conventional vehicles is presented. The equipment can be used to provide information of how the different driving cycles and geographical areas affect the battery performance. Furthermore, different hybrid strategies can be tested and compared with regard to the battery ageing, [2].

1.1 HEV Technology Overview

A hybrid electric vehicle (HEV) is one in which the outputs of an internal combustion engine (ICE) and an electric machine are combined to propel the vehicle. In what proportion should each of these machines propel the vehicle depends on a series of decisions, often called the *Hybrid Strategy*. A Hybrid Strategy is usually designed to either increase vehicle performance or to reduce fuel consumption.

The electric machine is fed by a battery pack, which can be charged in a number of ways. The "plug-in" variant of the HEV can take power from the grid to charge the battery; additionally, *regenerative braking* technology can be used to take advantage of the energy spent on braking a vehicle, which is wasted in the form of heat in conventional vehicles. Moreover, HEVs can be fitted with yet another electric machine which, when connected to the ICE, can work as a generator, providing current to the battery pack and hence the eletric motor.

1.2 Scope of this work

The aim of this thesis is to develop software (to be inserted on the controller of the field test equipment) that emulates the electric parts of the powertrain in a hybrid car. Based on data collected from a car with a conventional powertrain, the current through a single battery cell of a hybrid car is calculated. This will be used to perform lifetime tests of a battery cell, by charging and discharging it according to the calculated currents, "on-the-run" so to say.

The test equipment will provide information about how the lifetime of the battery cell is affected by how the driver uses the HEV together with the hybrid strategy. By installing the equipment in a real car, information of how the environment (like temperature and vibrations) affects the battery is also obtained. The equipment will also monitor and perform, periodically, a number of parameter tests on the battery cell during the time in the field.

1.3 Related Work

The main body of this work consists of the development of a test equipment which can be used in the field; this involves the software modeling of the necessary components as well as the strategy that regulates their activity, it also involves the implementation of the software into an on-board computer. This part of the work is largely based on the work by Pontus Svens, [1, 2, 3], where a similar project was discussed for hybrid electric trucks, with some promising results.

The tests carried out, as well as the interpretation of data, in the work by Svens is reused in this work, and it is based on the guidelines provided by the Advanced Vehicle Testing Activity [4, 5], where large scale tests of electric vehicles from different manufacturers are conducted. Different vehicles are tested during 160,000 miles and are compared according to the performance of both the vehicle and the vehicle battery.

Another part of this work is the development of an adequate state of charge estimation algorithm for the battery cell. The algorithm presented in this work has its foundation on Kalman Filtering theory, which can be reviewed in [6]. In short, a Kalman Filter is a model-based, state-feedback observer that is capable of providing information about a system, when this is not readily available. The filter is also provided with some information about the variations in the system due to uncertainties, whether they are in the process or can be traced to faulty measurements; the state of the system is calculated iteratively and corrections are made based on the information about uncertainties.

The Kalman Filter needs some additional adaptations, due to the fact that they are designed for linear systems (which can be time-varying or otherwise) and a battery cell is a nonlinear system. The resulting algorithm is known as the Extended Kalman Filter, and it is based on the work by Wahlstrom [7]. A systematic approach to understand a battery performance through predictive models and simulations was also conducted by [8].

1.4 Limitations

There are some aspects that are worth mentioning, since they are limitations which, for one reason or another, this work must go around. One of these limitations is that the state-of-charge measurement algorithm is done only for a single cell, that means functions such as cell-balancing (and how they affect battery lifetime) are not taken into account. Another limiting aspect is that the hybrid strategy developed in this work provides accuracy only for "normal", function-oriented behavior in the vehicle; the strategy is not suited to study aggresive, performance-oriented behavior. In addition, only the components with significant power consumption are modelled.

Another limitation relates to the battery's environment. Active cooling systems are not modelled

and so it is assumed that the cell has a passive cooling system (for example, a heat sink). For this reason, care must be taken as to the amount of current allowable on the battery.

2 The Big Picture

In this section the big picture (hardware) will be presented in outline. The system is divided into several blocks, and their relations to each other are briefly explained, as well as a short description on the main components.

2.1 System Description

The system consists of four main parts: the vehicle, the electronic control unit (ECU), the battery management unit (BMU) and the battery cell. The blocks communicate using the CAN (Controller Area Network) protocol. Relevant data (e.g. pedal position, vehicle speed and current gear) is read from the CAN bus by the ECU and used as inputs by the subsystems implemented in software (which is the main focus of this thesis). The user commands and the component ouputs are used by the hybrid strategy (implemented in software inside the ECU) to compute torque requests to the most active components. These requests then generate a current command that is sent to a Battery Management Unit (BMU), and finally the BMU takes the request and takes corresponding actions on the battery cell.

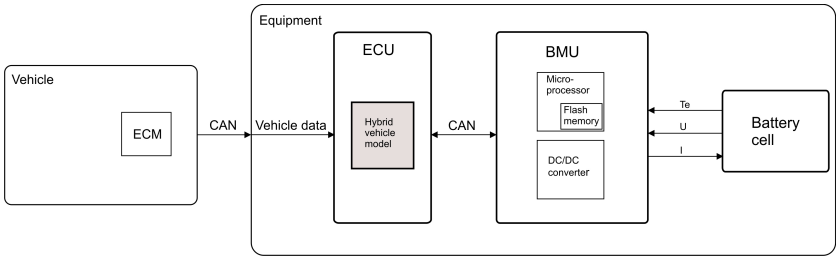


Figure 1: The global equipment

2.2 Battery Management Unit (BMU)

The Battery Management Unit is a third party device that handles the current inputs to and from the battery cell, it consists primarily of a couple of step converters (step-up for battery discharge and step-down for battery charge) and a microprocessor to handle commands, as well as a logging module (uses microSD card). [9]

2.3 Electronic Control Unit (ECU)

All the emulated functions (test functions, electric components, etc.) will be connected together and implemented in an ECU in the form of C-code. The controller is in charge of sending a current command to the BMU.

2.4 The Battery Cell

The battery cell used in this thesis is a commercial Li-Ion battery with 15Ah capacity and 3.75 V nominal voltage.

3 HEV Architecture

The hybrid electric vehicle, see figure 2, consists of a number of components that relate to each other and whose actions are coordinated by the hybrid strategy. The main components are listed below.

- Electric motor and clutch (ERAD),
- Generator (ISG),
- Internal Combustion Engine (ICE),
- Battery pack with control module (BECM),
- Transmission,
- DC/DC converter,
- Inverters,
- Air conditioning unit (AC),
- Electronic control module (ECM),
- Brake ECU.

The architecture studied in this thesis is the parallel architecture. It uses an electric motor in parallel with an internal combustion engine to propel or brake the vehicle, engaging one or both machines as needed. The motor's energy input comes from the battery and both the engine and the electric motor are regulated by the electronic control module (where the hybrid strategy is located). Whether the engine or the electric motor is used depends heavily on a power threshold, which are derived from the internal combustion engine's (ICE) power map. The operating modes of the electric motor depend on the current state of charge (SOC) of the battery cell. These modes are the charge depletion (the electric motor provides the necessary power to propel the vehicle) and charge sustaining (the electric motor is used to aid the combustion engine in propelling the vehicle).

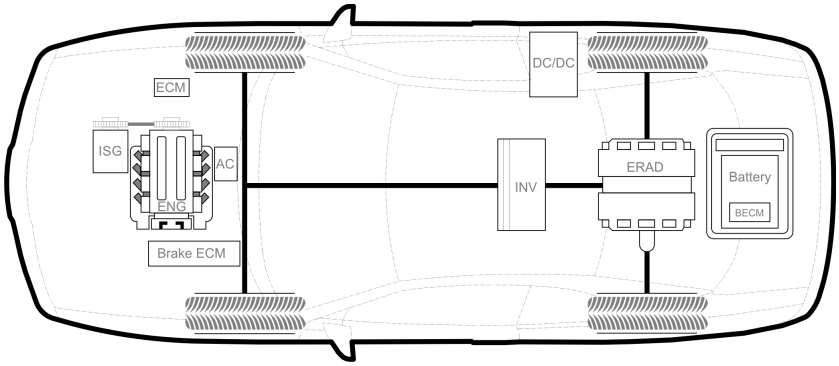


Figure 2: HEV Architecture

Another key component in this architecture is the generator which is connected by a belt system to the internal combustion engine and can be used to either provide a boost to the engine (at engine start-up) or recharging the battery (at low battery state of charge).

There are some lower-level components that have been taken into consideration. One of these components is the electrical air conditioning unit, which regulates both internal environment and operating temperatures of components. The DC/DC that powers the low-voltage components, and the inverters that control the electric machines (electric motor and generator) were also considered.

4 Component modelling

In this section, the components mentioned in section 3 are discussed. Their functions and their relations to other components are studied in order to implement a software model. These models will be integrated within the frame of a hybrid strategy, which will emulate the behavior of the hybrid part of the vehicle and send commands to the battery.

4.1 The Electric Machine and Power Electronics

The Electric Rear Axle Drive (ERAD), as its name implies, is the electric part of the hybrid powertrain and it is connected to the rear axle. It consists mainly of an electric machine, its drive system (a 3-phase inverter and a controller scheme) and a clutch that links it to the chassis.

The machine draws power from the battery pack to propel the vehicle, and in generator mode it can feed power back by using regenerative braking. The behavior of the electric machine can be modelled in several ways, one possibility is to use look-up tables, also known as *power maps* for the machine, for the motor mode the input is the requested torque (which is sent by the hybrid strategy). Another input is the estimated speed of the motor calculated from the chassis speed, and the output will be the power consumed by the motor, as well as other parameters needed by the strategy such as rotor speed. This method was applied effectively in [1, 2, 3].

The approach has the advantage of simplicity, and it is sufficiently adequate for steady state operation. However the dynamics of the machine itself are left out, and while this has not proven to be critical in previous work, in this work a more detailed approach will be sought, as it is possible that more realistic behavior will increase the depth of the main object of study (the battery cell and its behavior).

4.1.1 Electric machine

The alternative approach consists on using a mathematical model of the machine, the model to be used is the standard model used for Permanent Magnet Synchronous Machines (PMSMs), and it is taken from [10, 11]. The model was developed using the rotor as the reference frame, and set in dq coordinates.

The main assumptions of this model are:

- Saturation is neglected.
- The induced electromotive force (emf) is sinusoidal.
- Eddy currents and hysteresis losses are negligible.
- There are no field current dynamics.

Flux linkages are given by:

$$\lambda_q = L_q * i_q \tag{1}$$
$$\lambda_d = L_d * i_d + \lambda_f \tag{2}$$

where L_q and L_d are the inductances in q and d directions, i_q and i_d are the currents in the q and d directions and λ_f is the stator winding. The voltage equations are given in matrix form according to the circuit shown in figure 3:

$$\begin{bmatrix} V_q \\ V_d \end{bmatrix} = \begin{bmatrix} R_s + \rho L_q & \omega_r L_d \\ -\omega_r L_q & R_s + \rho L_d \end{bmatrix} \begin{bmatrix} i_q \\ i_d \end{bmatrix} + \begin{bmatrix} \omega_r \lambda_f \\ \rho \lambda_f \end{bmatrix} \tag{3}$$

where R_s is the stator resistance, ω_r is the electrical angular frequency and p is the number of poles. The developed electromagnetic torque is given by:

$$T_e = \frac{3}{2}(\lambda_d i_q - \lambda_q i_d) \tag{4}$$

The mechanical part is described by Newton's second law for rotational motion:

$$T_e = T_{load} + B\omega_m + J\frac{d\omega_m}{dt} \tag{5}$$

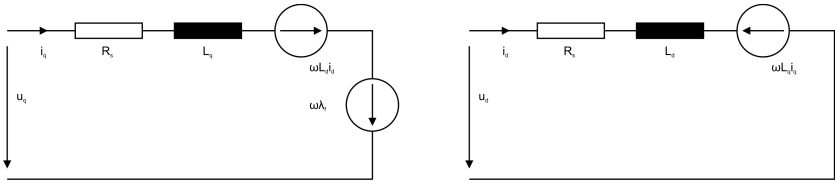


Figure 3: Two-axis (dq) equivalent circuit of PM synchronous machine [10].

where the term $B\omega_m$ represents the friction losses. In this work the friction losses will be modeled not by estimation of B but by a look-up table, also called a *loss map* in this context. ω_m is the mechanical speed of the rotor, and is related to the electrical speed in the following way:

$$\omega_m = \frac{\omega_r}{P} \tag{6}$$

where P is the number of pole pairs.

The system will have the electromagnetic torque as input, and the rotor speed as the output, so now ω_m is isolated on the previous equation:

$$\omega_m = \int \frac{T_e - T_{load} - B\omega_m}{J} dt \tag{7}$$

4.1.2 Drive system

As mentioned before, the electric motor is a 3-phase AC machine, yet its main power source is the battery (a DC source), therefore a DC/AC converter, also known as *inverter*. The inverter is also used to control the torque and speed of the machine, according to hybrid strategy commands. The triphasic *abc* current waveforms, which are the ouput of the inverter, can be regulated by several techniques, of which two were tested. These are *Carrier-Wave Pulse Width Modulation* (Carrier-Wave PWM) and hysteresis PWM. They were modelled and tested with some success but it was preferred to keep computational demands low, and so only critical functions were implemented (control functions).

The ERAD system specifications require a torque controller and a speed controller. The torque controller is used in motor and generator mode and the speed controller is used when the ERAD shaft needs to be synchronised with the wheel shaft. The control used for both torque and speed which is a PI controller was tuned manually, [10, 12].

4.1.3 Model implementation

During normal operation (when the clutch is closed), the load torque on the ERAD is given by the mass, speed and inclination angle of the car. Since it is hard to accurately estimate this load torque, its effects on the machine are emulated by dividing the model into static (steady-state) and dynamic models. The dynamic model is used only when the clutch changes state. A conceptual map is shown in figure 4.

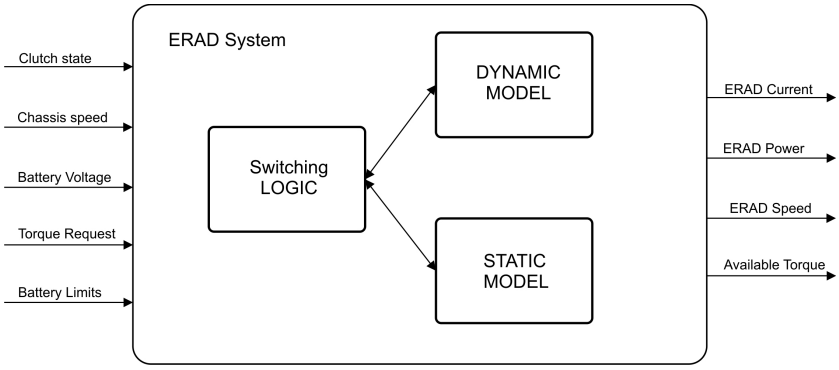


Figure 4: The ERAD system

Since the two models cannot operate at the same time, a switching logic has to be included. The logic has four operating states which depend on the ERAD clutch, see figure 5. When the clutch is closed the static model is active and when the clutch switches between states the dynamic model is active for a short time. In case the clutch state changes from open to closed, the dynamic ERAD model has to ramp up the speed of the motor from zero speed until it reaches the chassis speed. When the clutch changes from closed to open, the same dynamic model ramps down the speed from the chassis speed to zero speed. If the clutch is open no model is active since the ERAD system is unused.

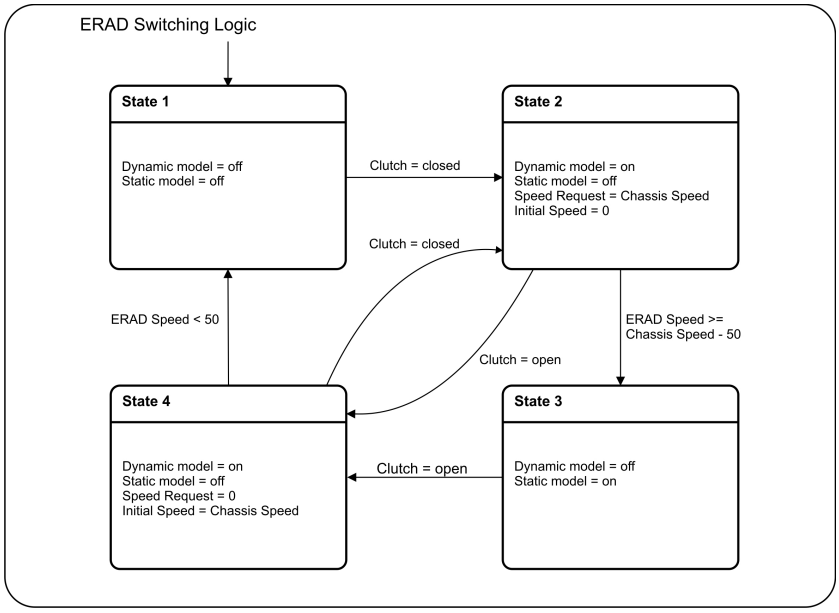


Figure 5: The ERAD switching logic

The dynamic model simulates a synchronous motor with speed control, see figure 6. It consists

of a PI speed controller, a battery and motor limitation block, a torque control block and the motor model.

The speed controller is a manually tuned PI controller which gives torque commands to the torque controller. The function of the battery and motor limitations block is to limit the requested torque from the speed controller according to the battery power limits and the motor limits. The torque controller ensures that the electromagnetic torque of the motor, equation 4, follows the reference. This was initially done by controlling the current of the electrical model of the motor, equation 3, but since the simulation proved to be very slow because of this, the solution was discarded and only a torque delay is used instead of the whole torque controller. As for the mechanical motor block, it is implemented according to equation 7 where the load torque is considered zero.

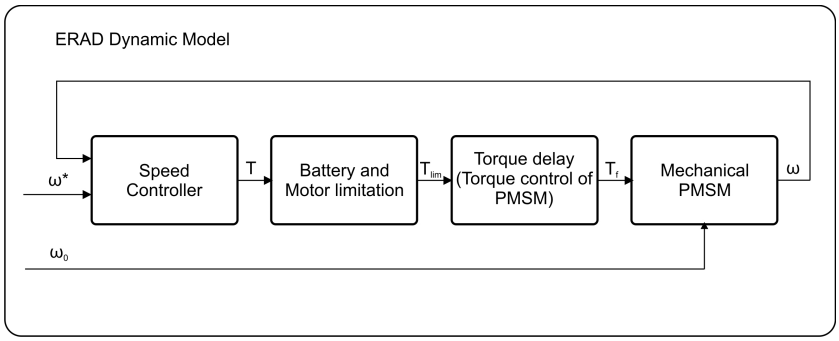


Figure 6: The ERAD dynamic model.

As can be seen in figure 4, the main output of the model is the machine current, which can be estimated in both models with the following function:

$$I_{ERAD} = \frac{T_{e,ERAD} \cdot \omega + P_{loss}}{U} \tag{8}$$

The static model consists only of equation 8. The only difference is that the implemented control function to estimate T_e is only a system time constant for the requested torque and that the speed of the motor ω is calculated from the chassis speed by multiplying it with the erad gear ratio. U is the battery voltage at that time. The power losses in the motor and in the inverter P_{loss} are also taken into consideration in the form of look-up tables.

4.2 The Generator

4.2.1 Component Description

The hybrid vehicle's generator is similar to the ERAD but without the clutch. The machine is connected to the engine crankshaft through a belt and is mainly used to help the engine at start-up and to charge the battery when the state of charge is below a certain value, i.e. not for propulsion.

4.2.2 Model implementation

The generator only operates when the engine is running and when the hybrid strategy sends a torque request. Because it's dynamic behaviour is of low importance, the generator is not modeled dynamically, implemented instead as a simple lookup table. The machine and the drive system consumed current is estimated according to the following function:

$$I_{ISG} = \frac{T_{e,ISG} \cdot \omega + P_{loss}}{U} \tag{9}$$

The controller that regulates the electromagnetic torque T_e is simple a system time delay (which means that the reference and the actual output are identical, but delayed) and a rate limiter for the requested torque. The generator speed ω is considered the same as the crankshaft speed and U is the battery voltage at that time. The power losses in the motor and in the inverter P_{loss} are also taken into consideration in the form of look-up tables.

Beside the consumed current (needed to calculate the final current sent to the cell) the generator model also reports its speed, power and the available torque to the hybrid strategy. The available torque is simply the computed maximum torque for the component, but passed through all of the hybrid strategy's requirements and limitations. This means that the actual output of the generator may be smaller that its available output.

4.3 The Battery System

The cell studied in this work is a lithium-ion cell. The cell is part of a battery pack which is used to collect and distribute electrical power (DC Power), mainly to and from the electrical drive, the air conditioning system, DC/DC converter and the generator.

On a standard implementation, the battery pack is expected to provide power for both high-voltage and low-voltage systems. The low-voltage systems involve only safety functions in the form of a series of contactors which are turned on or off to engage or disengage the battery pack (or in this case, the cell) according to a pre-determined set of circumstances. The low-voltage distribution is of no relevance to this work because the cell will be isolated from the actual powertrain, and thus safe operation modes need not be taken into account. Furthermore, the operating range of the cell will be controlled using software, which emulates the actual control module on a real hybrid car and is described on the next section.

4.3.1 BECM: Component Description

In a hybrid vehice, the cell is controlled by a separate module, known as the Battery Electronic Control Module (BECM) which takes voltage, current and temperature readings and uses them to estimate power limitations at a given point in time. Safe operation of the battery cell is assured by estimating such variables as state of charge (SOC)and state of health (SOH). All this information is reported to the global vehicle control module, which will then decide how to use the power available from the battery.

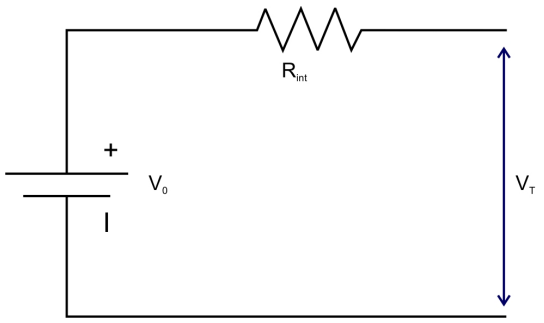


Figure 7: Simple Thevenin Battery Model. [7]

In this work, the module is implemented on the Electronic Control Unit (ECU) with reduced functionality, i.e. the model is stripped off most of its auxiliary functions (such as fault detection on the physical implementation).

4.3.2 State of Charge

The state of charge (SOC) in a battery is a measure of the amount of capacity that a battery can provide, in relation to the capacity it had when it was fully charged. The BECM needs this information in order to determine whether the battery can be used to provide power, or be charged to a certain level before it can be used again.

The quantity used to calculate the state of charge is usually the capacity, measured in Ampere-hours. And the actual figure is expressed as a percentage from the current capacity to the rated capacity at the present conditions.

There are a variety of methods to estimate the state of charge, from which the three most commonly used will be discussed in the next sections. These methods are the Open Circuit Voltage, the Coulomb Counting Method method and the Adaptive (or Kalman Filtering) Method [13, 14, 15, 16, 17, 7].

It is important to note that all these methods are dependent on parameters that vary according to changing conditions in the cell, these conditions being temperature, charge/discharge regimes and the age of the cell. If reliable state of charge estimations are desired, these variations need to be taken into account. Ways to monitor these variations will be discussed in the following sections.

Open Circuit Voltage The Open Circuit Voltage (OCV) method for estimating the SOC is fairly well known. It consists of performing laboratory measurements of the unloaded battery terminals and compare the readings against remaining capacity, and compare this to the rated capacity. The end result is a curve that describes the behavior of the state of charge as the voltage changes. These curves have to be performed for several operating temperatures, and also account for discharge rates and age of the battery.

A factor to consider is the variations of the internal resistance with discharge rates and temper-

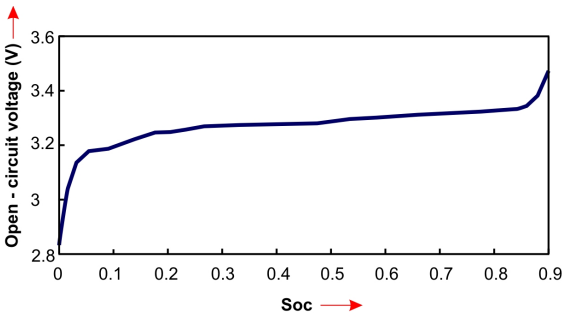


Figure 8: Generic Open Circuit Voltage (OCV) versus State of Charge (SOC) curve for a lithium-ion battery. [18]

ature. To get an accurate measurement of the OCV, a good estimation of the internal resistance is needed, methods to achieve this will be discussed in section 4.3.4.

The adequacy of this method varies with battery chemistry, and while it may prove to be sufficient for lead-acid batteries, it is not the case for lithium ion batteries, as can be seen on figure 8 for the typical behavior observed in one of these batteries, where there is little variation in the OCV for a large variation in remaining capacity, so innacurate measurements of as little as 0.05V can produce estimation errors of up to 30% in the SOC's reported measurement. Therefore, this method alone is inadequate for the battery studied in this work.

Coulomb Counting Another well-known method to estimate the state of charge is the one known as Coulomb Counting. This method measures capacity directly, by measuring the current and then integrating with respect to time. The battery capacity is defined as follows:

$$C = \int i \, dt \tag{10}$$

The capacity is measured during a full charging of the battery and then the net current is monitored (Inet=Iin-Iout) and integrated over time, the state of charge is obtained from the following equation:

$$\text{SOC} = \frac{1}{C_{max}} \int i_{net} \, dt. \tag{11}$$

The advantage of this method is its simplicity, however there could be some problems if the sensor readings are innacurate. The error produced by a bad measurement will be carried into the integration and the state of charge reading will drift over time.

Another amendment must be made for the coefficient C_{max} in equation 11, for this value varies with battery age, temperature and discharge rates, among other possible factors. Therefore to get an adequate estimation this coefficient must be corrected periodically, either by laboratory measurements, tests on the field or adaptive methods. This situation will be considered again in section 4.3.4.

Adaptive Methods: Kalman Filtering The previous methods are in essence open-loop approaches (no correction feedback) to the problem of estimating the state of charge of the battery, and due to their inadequacies they are often used together, relying on the OCV method for the regions of large changes of the open-circuit voltage and on the Coulomb Counting method for regions of small voltage variations.

The adequacy of the measurements are still an issue, however, rather than invest in better quality (and costly) sensors, a better approach can be to turn to self-corrective methods provided by the field of control theory. Of particular interest is the adaptation of the Kalman Filter for SOC estimation. In short, the Kalman Filter is a recursive (and optimal) algorithm that uses information about the system, as well as disturbance sources (from the process and/or the measurement) to predict the next state of the system.

An important feature of the Kalman Filter is that it is suited only for Linear Time-Variant Systems (LTV Systems). Since battery systems are non-linear in nature, another approach must be used. The Extended Kalman Filter overcomes this problem by linearizing the system around an operating point with every iteration, so that the system can be solved as a LTV System. This is done at a cost, for the optimality inherent in the Kalman Filter is no longer present in this method, see [6].

Kalman Filtering theory is beyond the scope of this work and will be presented only in outline, which is taken from [15, 7].

Extended Kalman Filter (EKF) Consider the non-linear system:

$$x_{k+1} = f(x_k, u_k) + w_k \tag{12}$$
$$y_k = g(x_k, u_k) + v_k \tag{13}$$

The system will be linearized to take the form of a standard linear model:

$$x_{k+1} = A_k x_k + B_k u_k + w_k \tag{14}$$
$$y_k = C_k x_k + D_k u_k + v_k \tag{15}$$

where w_k and v_k are process disturbances and measurement noise, respectively. This uncertainties are usually characterized by white Gaussian noise with covariances Σ_w for the process disturbance and Σ_v for the measurement noise, such that:

$$w_k \ (0, \Sigma_w)$$
$$v_k \ (0, \Sigma_v)$$

The EKF, like the classic Kalman Filter, has two stages, a prediction stage and an update stage. The process model is used to predict the belief of the state and the measurement model is used to update the belief.

During the prediction stage, an estimate of the state based on a model of the process is calculated, as well as prediction of the covariance. This belief is denoted as $\tilde{x}_k$. Prior to the update, the belief is denoted as $\tilde{x}_k^-$, and after the measurement update is denoted as $\tilde{x}_k^+$

At each time step, $f(x_k, u_k)$ and $g(x_k, u_k)$ are linearized using a first order Taylor series expansion around the point $x_k = \tilde{x}_{k-1}^+$ for the process and $x_k = \tilde{x}_k^-$ for the measurement model.

$$f(x_k, u_k) \approx f(\tilde{x}_{k-1}, u_k) + \left.\frac{\partial f(x_k, u_k)}{\partial x_k}\right|_{x_k=\tilde{x}_k-1} (x_k - \tilde{x}_{k-1}) \tag{16}$$

$$g(x_k, u_k) \approx g(\tilde{x}_k^-, u_k) + \left.\frac{\partial g(x_k, u_k)}{\partial x_k}\right|_{x_k=\tilde{x}_k^-} (x_k - \tilde{x}_k) \tag{17}$$

Substituting equations 16 and 17 into equations 12 and 13, and defining A_k and C_k as follows:

$$\hat{A}_k = \left.\frac{\partial f(x_k, u_k)}{\partial x_k}\right|_{x_k=\tilde{x}_k-1}$$

$$\hat{C}_k = \left.\frac{\partial g(x_k, u_k)}{\partial x_k}\right|_{x_k=\tilde{x}_k^-}$$

The linearized process and measurement models become:

$$x_{k+1} \approx \hat{A}_k x_k + f(\tilde{x}_k, u_k) - \hat{A}_k \tilde{x}_k + w_k \tag{18}$$

$$y_k = \hat{C}_k x_k + g(\tilde{x}_k, u_k) - \hat{C}_k \tilde{x}_k + v_k \tag{19}$$

To define the EKF iteration process, the B_k of the linear model is replaced with the two middle terms of equation 18, and the D_k is replaced with the two middle terms of equation 19.

Now that the process is linearized, the Kalman Filtering process can be applied:

Prediction stage:

$$\tilde{x}_k^- = f(\tilde{x}_{k-1}^+, u_{k-1}) \tag{20}$$

$$\Sigma_{\tilde{x},k}^- = \hat{A}_{k-1} \Sigma_{x,k-1}^+ \hat{A}_{k-1}^T + \Sigma_w \tag{21}$$

Update stage:

$$L_k = \Sigma_{\tilde{x},k}^- \hat{C}_k^T [C_k \Sigma_{\tilde{x},k}^- \hat{C}_k^T + \Sigma_v]^{-1} \tag{22}$$

$$\tilde{x}_k^+ = \tilde{x}_k^- + L_k [y_k - g(\tilde{x}_k^-, u_k)] \tag{23}$$

$$\Sigma_{\tilde{x},k}^+ = (I - L_k \hat{C}_k) \Sigma_{\tilde{x},k}^- \tag{24}$$

The main product of this filter is the corrective gain L_k, which also depends on an additional *error covariance*, denoted as $\Sigma_{\tilde{x}}$

4.3.3 Implementation of EKF algorithm

To estimate the state of charge (SOC) of the battery cell by use of an Extended Kalman Filter, begin with a model of the process, which is the charging and discharging of the battery. The model chosen is the Coulomb counting model described before:

$$\text{SOC} \;=\; \text{SOC}_0 + \frac{1}{C_{max}} \int (i_{net} - i_{loss})\, dt. \tag{25}$$

Where i_{net} is the net current on the battery. To implement this model in a controller, equation 25 needs to be discretized, which gives the following equation:

$$\text{SOC}_k = \;\text{SOC}_{k-1} + \frac{\eta i_k \Delta t}{C_{max}} + w_k \tag{26}$$

Where SOC is the state to be estimated, i_k is the control input C_{max} is the nominal capacity of the battery in Ampere-seconds, and η is the coulombic efficiency of the battery for either charge or discharge. The efficiency is obtained from look-up tables, but in general it tends to be 1 for discharge and around 99% for charge [13, 14, 17, 7].

Available measurements from the cell are current, voltage and temperature. Since the current is the control input, and there is no simple way to relate the temperature and the state (SOC), the battery voltage will be used to construct a measurement model. There is already data available that relates the open circuit voltage with SOC. Recall the simple electrical model of the battery, the measurement model is then the voltage equation:

$$y_k = \;\text{OCV}(\text{SOC}_k) - Ri_k \tag{27}$$

On this model, the internal resistance R is assumed to be the same for charge and discharge cycles, furthermore, there are lookup tables available that have resistance values according to operating temperature and cell age.

At this point all that is left is to substitute these terms into the Kalman Filter equations. This model has only one state (SOC), therefore $\hat{A} = 1$, furthermore the linearization of the output model is shown:

$$\hat{C}_k = \frac{\partial g(x_k, u_k)}{\partial x_k} \tag{28}$$

Since neither the internal resistance or the current depend directly on SOC, the term Ri_k is discarded, and the only remaining term is $OCV(SOC)$. A standard differentiation is then made for small deviations in the state, denoted by δ:

$$\hat{C}_{k,\text{OCV}} = \frac{\text{OCV}(\text{SOC}_{k-1} + \delta) - \text{OCV}(\text{SOC}_{k-1} - \delta)}{(\text{SOC}_{k-1} + \delta) - (\text{SOC}_{k-1} - \delta)} \tag{29}$$

An additional step before being able to calculate the gain L_k is the characterization of the disturbances (covariances for the process and measurement disturbances). This characterization is an expression of which part of the system is more reliable, and for a lithium-ion battery the answer

is not obvious. What this means for the disturbance characterization is that, by looking at figure 8, it can be seen that measurements should not be trusted while the system lies in the flat region, rather the mathematical model should be trusted. On the other hand, for the steep region the measurements can be trusted more than the mathematical model.

The values used in this work are found by trial and error, and are qualitative expressions of how the algorithm should estimate the state of charge; they are 1.8 for the process disturbance covariance v_k and 1.3×10^{-17} for the measurement noise covariance w_k. It should be noted that the covariances used by the EKF (not the ones chosen as initial covariances) are updated with every iteration, and these variations depend on the OCV to SOC curve. An initial Kalman gain of 0.0001 was also used in order to trust the measurement model at the beginning where the error in SOC is assumed to be large.

A shortcoming of the algorithm implemented in this work is that the rated capacity (C_{max}) is being taken from lookup tables; a better way would be to implement a second EKF to estimate the value of this parameter. This is left as future work.

4.3.4 Parameter Estimation

As mentioned before, in a battery cell the capacity and resistance vary with temperature, current, SOC and age. Various tests are made by the manufacturer to determine these relations. Some tests have been used to form lookup tables for more accurate SOC and power limits calculation.

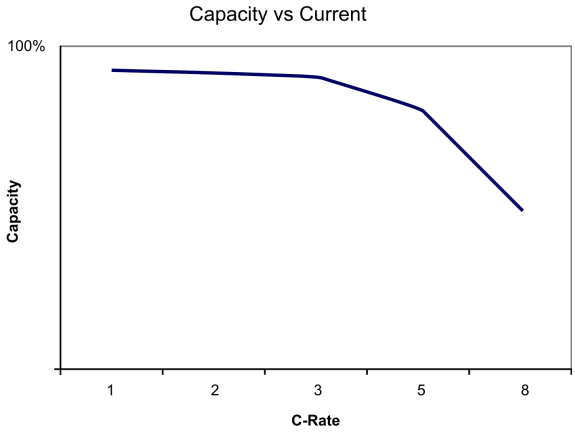
Parameter behavior The rate of charge and discharge of current as well as the operating temperature have a big impact on a battery's capacity. An example for a 30Ah Li-Ion battery is shown in Figures 9a and 9b. These figures show that higher C-Rates or lower temperatures reduce the available capacity of the cell.

It can be seen from figure 9c that the biggest influence of the cell's internal resistance is the operating temperature, with higher resistance at lower temperatures.

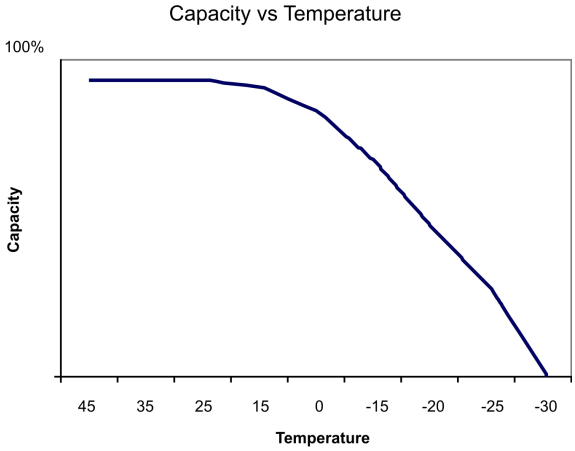
In time, the battery cell ages and both the resistance and the capacity are affected. The aging process depends on how the battery is used; mainly the operating temperature, SOC, charge and discharge rate limits as well as number of cycles. For this reason, it is very hard to predict the aging process of a battery cell.

A solution is to perform laboratory tests and put the results into lookup tables, but this makes them unreliable; laboratory tests imply controlled conditions, which is not the case for a normal driving cycle. A lookup table generated in this way is good but only as a reference point, and a follow-up of these values should be carried out.

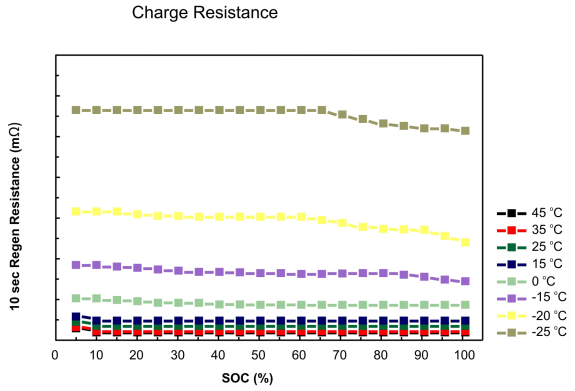
For the problem of updating the lookup table data, several solutions to calculate these parameters are taken into consideration. Solutions can be implemented either while the hybrid components are in use (online) or while their operation is suspended (offline). The offline solutions estimate the parameters at certain intervals with some tests which will be done on the battery. The online solutions estimate the parameters continuoulsy, but at the cost of more requirements on processing power.



(a) Rated capacity [Ah] values for different current discharge rates.



(b) Rated capacity [Ah] values at different temperatures.



(c) Battery's internal resistance values at different temperatures.

Figure 9: Battery parameters and their variations.

Offline Methods It is possible to obtain parameter information by performing standardized tests, in this work the tests are performed at the cell level. A current pulse test is performed to read the cell's internal resistance, and a current charge/discharge test is performed to read the cell's rated capacity. An example of such a test can be seen in [19].

- Pulse Test

  A current pulse is applied to the battery. Since the dynamics of the open-circuit voltage are too slow to change with this pulse, the resistance can be calculated using Ohm's Law, solved for R:

  $$R = \frac{\Delta V}{\Delta I} \tag{30}$$

  The test is made at a certain temperature and state of charge. The result is used to update the "Resistance vs Temperature" lookup table with the new values. A problem is that the resistance at only one temperature is calculated and that it would be inefficient to make the test for all temperatures. Therefore, when updating the table, it is assumed that all the values in the resistance vs temperature curve change with the same ratio.

- Capacity Test

  In order to estimate the capacity the battery cell is charged to 100%, and then it is fully discharged, from 100% SOC to 0%. From the state of charge definition it can be seen that the battery capacity will be equal to the integrated current during the full discharge.

  $$C = \int i\, dt. \tag{31}$$

  Like the resistance, the calculated capacity corresponds to a certain temperature. All the values in the capacity vs temperature lookup table are updated with the same ratio. The main disadvantage of this test is the large time requirements of the test.

Online Methods The online approach considered in this work is the dual Extended Kalman Filter from [7]. While there are alternative ones, such as fuzzy logic control logic and artificial neural network design, the Extended Kalman Filter is adequate enough for a battery cell.

As explained in section 4.3.2, a model of the process is needed, the model proposed in [17] is the one used. It does not take into consideration the losses in a battery cell, only the variation of the capacity due to the error covariance.

$$C_k = C_{k-1} + w_k \tag{32}$$

Furthermore, the measurement model is as follows:

$$D_k = \mathrm{SOC}_k - \mathrm{SOC}_{k-1} + \frac{1}{C_k} i_k \Delta t \tag{33}$$

Notice the difference between equations 33 and 26, this construction is intended to drive the value of D_k to zero, so it is compared to zero on the calculations. The error of this comparison is used to update the capacity estimate.

The EKF for capacity estimation was implemented but the results were inadequate (depending on the initial value and covariances, the fastest convergence was achieved in little less than an hour). This was due to several reasons: the model of the capacity is perhaps too simple and more dynamics need to be introduced; in addition, a real battery would be needed to obtain actual parameters and have a more reliable algorithm. These setbacks can be overcome with more time, and so this is left as future work.

4.4 Other Subsystems

In this section the lower-level subsystems will be presented one by one. A short description of their functions is given and a short discussion on their relevance is also included.

4.4.1 Air Conditioning System

The air-conditioning subsystem is in charge of regulating the temperature in the vehicle's inside environment, and the power consumed by it is a function of the difference between the user's desired temperature and the external temperature, among other factors. The power requirements of this subsystem are low when compared to those on the other subsystems, such as the electric machine. This is also true in *worst case scenario* conditions, which involve temperature differences of up to 30 °C. In view of this, it was decided that this subsystem could be neglected.

4.4.2 DC-DC Converter

The DC-DC converter takes high voltage power from the battery cell and moves it to low-voltage, in order to feed the components that work in this range of voltage. Power consumption and losses on this component are relatively low, and so it was discarded.

4.4.3 Internal Combustion Engine (ICE)

The equipment will be installed in a normal car. This means that the ICE will behave differently compared to how it would behave on a hybrid car. Therefore, the torque cannot be read directly from the CAN bus and a model of the ICE needs to be introduced. For simplicity reasons, the model is made to only account for the main behavior, which is the on-off switching according to the hybrid strategy rules. The dynamic part of the system is left out by assuming that the ICE torque is equal to the requested torque when the engine is on and that the engine speed instantly reaches the desired speed when the ICE changes operating modes, see figure 10.

The crankshaft speed is calculated using a gear ratio map that has the current gear and the vehicle speed as inputs. This solution is implemented to avoid errors because of differences between the model and the vehicle and to make the model usable in any vehicle without having to modify the parameters.

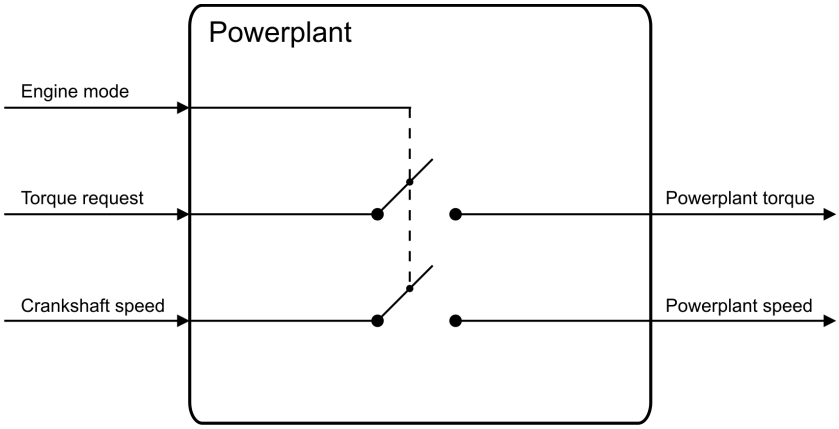


Figure 10: The ICE model.

4.4.4 Battery power limits

The lifetime of a battery greatly depends on the charge and discharge currents. In order to expand its life, various experiments have been made to clearly define the battery's operating range, as well as the maximum rated values. In this work, some of these limits were included to limit the output to the battery cell, and the equipment as well. Beside the maximum current limitations, the current is also brought down to allowable values as the open circuit voltage approaches it's marginal values (both maximum and minimum). Additionally, temperature and SOC variations are also taken into account by two lookup tables. The lookup tables limit the discharge and charge power (hence the current) accordingly. Since the RMS currents also have a significant impact on a battery's life, and related power restrictions are added due to that.

4.4.5 Adaptive Lookup Tables

The battery parameters such as rated capaticy C_{max} and internal resistance R_i depends and changes when the battery ages. To be able to report the state of charge to the hybrid strategy, this changes need to be taken into account.

The usual approach is to perform laboratory tests and record the obtained values of the parameters in a lookup table. The problem, however, is that there are other relationships that may change the parameters further, such as driver behavior (which translates into a current charge/discharge regime) and vehicle mode (strategy modified to optimize either performance or fuel consumption). These relationships are vehicle specific, so a single lookup table cannot suit all behaviors and so the lifetime cannot be expected to be monitored correctly.

To accomodate for this, an adaptive lookup table was introduced which can update the data on the table based on offline parameter testing. This means that use of the battery will be suspended for a certain amount of time, to perform the pulse and charge/discharge tests discussed in section 4.3.4. The MATLAB user-defined functions blockset was used to generate the C-code that will

perform the table update, while the tests were implemented in Simulink.

5 The Hybrid Strategy

5.1 Component Description

The hybrid strategy is the main control unit of a hybrid vehicle. It takes information from the vehicle and controls every component, see figure 11. From the driver's pedals and the vehicle speed, a requested vehicle torque is calculated through a lookup table. The main role of the hybrid strategy is to split this torque between the electric motor and the engine in an intelligent way, the strategy could be made to give priority to low fuel consumption, or to increase vehicle performance. Since the engine has high consumption and emissions rates at low powers, the electrical motor is used until a certain power treshold or when the engine requires assistance. Each engine has a specific treshold which can be chosen according to it's powermap. The hybrid strategy also charges the batter, either by requesting regenerating torque to the ERAD, or by requesting negative torque from the generator when the SOC from the battery cell is too low.

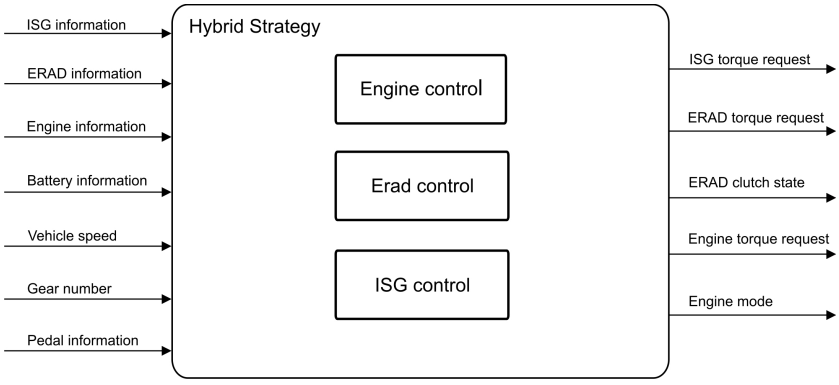


Figure 11: The hybrid strategy.

5.2 Model implementation

The major issue when implementing the hybrid strategy is that it also controls the engine, among other components. Because the actual engine in the car is not controlled, the ERAD torque request cannot be properly calculated since it depends on it. Also, because this engine torque should be controlled by the hybrid strategy it cannot be read from the normal vehicle. Therefore, a simple model of the engine was implemented. This model assumes that the requested torque is the same as the actual torque when the engine mode signal requires the engine to be on and 0 otherwise. The speed of the engine is calculated from the chassis speed by multiplying it with the gear ratio at the current gear number. This speed is also turned to 0 when the engine is supposed to be off. All this is discussed in more detail in the powerplant section.

This solution has unfortunately some drawbacks since it doesn't take the dynamics of the engine into consideration. It assumes that the engine has fast torque response. In normal operation the engine is slower and the ERAD has to account for the torque the engine cannot handle. Therefore, during more agressive driving, the ERAD torque request will be less accurate.

Another drawback of this solution is the missing boost that the generator gives to the engine at startup. This happens because the speed is not continuously calculated in the powerplant and it performs a jump every time the engine is turned on. A simple way to fix this problem is to change the engine mode block and request a constant cranking of 0.2 seconds (obtained by trial and error) every time the engine is turned on.

6 System Integration

Figure 12 shows the final connections between components, as well as input and output signals. The program takes vehicle information (from CAN messages) such as vehicle speed, pedals position, and current gear in consideration; additional information comes from the sensors connected to the battery (voltage, current and temperature). This information is sent to a number of software modules inside the global system (hybrid strategy, components, and limits), the end result being the current command to the battery.

An important feature of this implementation is the fact that the crankshaft speed and the chassis speed are derived from the vehicle speed (the crankshaft speed indirectly, through the chassis speed), rather than taking the sensor information from CAN. This is a necessary action, the main reason is to avoid contradictions in the system behavior, which arise due to differences in real-world and software dynamics. For example, the engine is controlled with an on-off scheme, and in the software the engine starts and stops almost immediately, but this is not the case in the real car, and so using information from a real car in this case could generate false responses from the software. Another reason is the design differences in hybrid strategies, which often handle each case differently.

The measures discussed previously offer an advantage: not reading this information directly from the car makes the model suitable for use in any kind of car (hybrid, plug-in hybrid, or normal car with only an ICE) and of any category (SUV, compact, etc.) without changing system behavior or causing erratic commands on the battery.

Once the components are integrated and the system sending commands to the battery, the final step is to design the logic that will enable either the use of the test equipment (ECU with the model inside), or enable the test procedures presented in section 4.3.4. See figure 13.

The test procedures are initiated to update the data in the lookup tables and to monitor the degradation of the cell. The tests are periodical, with an adjustable period, which is usually as long as two months between tests. While the tests are being performed the system in figure 12 is shut down.

An important aspect of the test procedures is that these must be performed within a specific temperature range, namely temperatures greater than 20 °C, this is because above this value, the battery's capacity and resistance remain constant, see figure 9b. A test usually lasts a few hours in which the temperature of the battery increases. Due to this, an increase in the temperature during

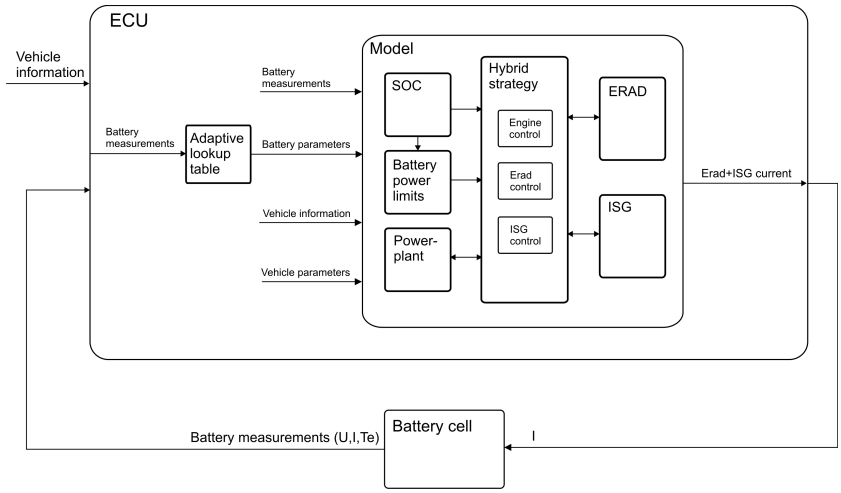


Figure 12: System integration

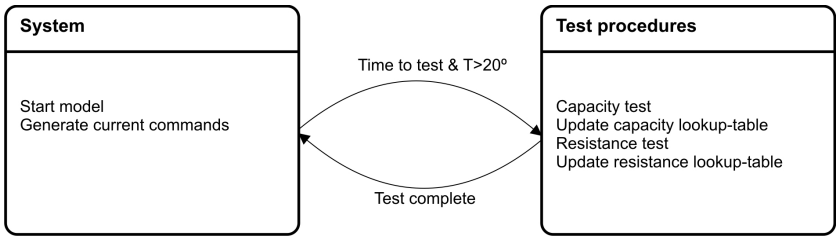


Figure 13: Global logic for the test procedures.

the test can cause errors in the calculated parameters if the starting temperature is below 20 °C.

6.1 Software Tools

All the components of the hybrid architecture described in the previous subsection as well as the hybrid strategy itself are modelled with the Matlab/Simulink software. Data for validation was extracted using the Canalyzer software solution. Information about the models is obtained from the VSIM software's model repository (provided by VCC). Finally, the global integration is compiled and programmed using the DSpace software.

7 Results and Discussion

In this section results are presented which were obtained when testing the software against data from real-world driving cycles. Furthermore, some observations are made on the differences between the software's output and what is reported by the car.

7.1 State of Charge Algorithm Validation

The Extended Kalman Filter discussed in section 4.3.3 is implemented, and tested against real world data (actual driving cycles). The data is extracted from the car's memory using the CANalyzer software. Since the algorithm is generic, there are no specific criteria with which the data is chosen, the only requirement being that the driving behavior should look "normal" (no anomalies in any of the components).

Figure 14 shows the measured SOC in the car against the SOC predicted by the algorithm, the data comes from a normal driving cycle, performed in winter, and with the battery on depletion mode at the beginning and charge sustaining after some time. The initial difference is due to differences in initial conditions (it could be assumed that the initial SOC is known, in which case the initial error would not take place), and after convergence is reached, errors get no bigger than 3% (measured SOC).

Figure 15 shows another performed test (with a different driving cycle), showing the comparison of the SOC reported by the car against the SOC calculated by the algorithm. This time the data comes from a driving cycle that consisted of a regular, long distance trip from a nearby town; the battery is kept in charge sustaining mode. The error in this case is much smaller.

Finally, figure 16 shows the alternate case to the one depicted in figure 15, which means that it is the return trip, so the behavior is quite similar, again long distance driving, with the battery on charge sustaining mode. It is not known with certainty the reasons for the observed differences between the software output and what is read from CANadlyzer. It is not known, for example, at what time does the driver start to measure data, or how the driver is using the vehicle. Regardless, errors are still reasonably small.

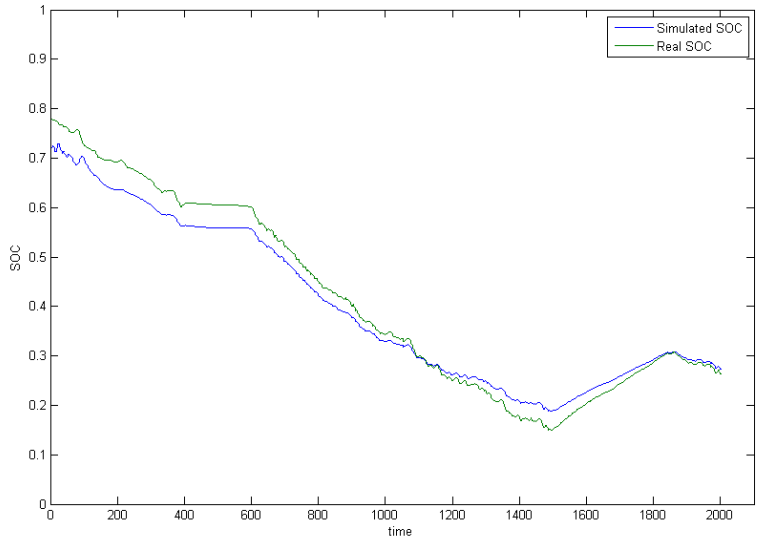


Figure 14: SOC validation: First driving cycle.

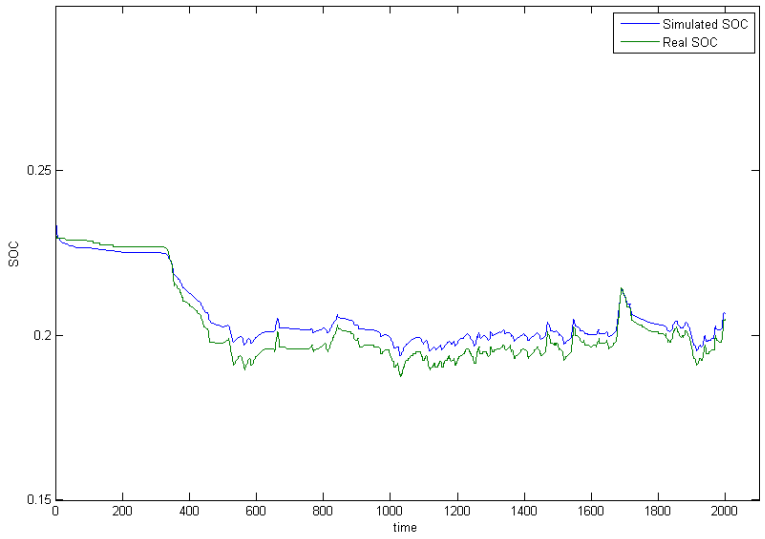


Figure 15: SOC validation: Second driving cycle

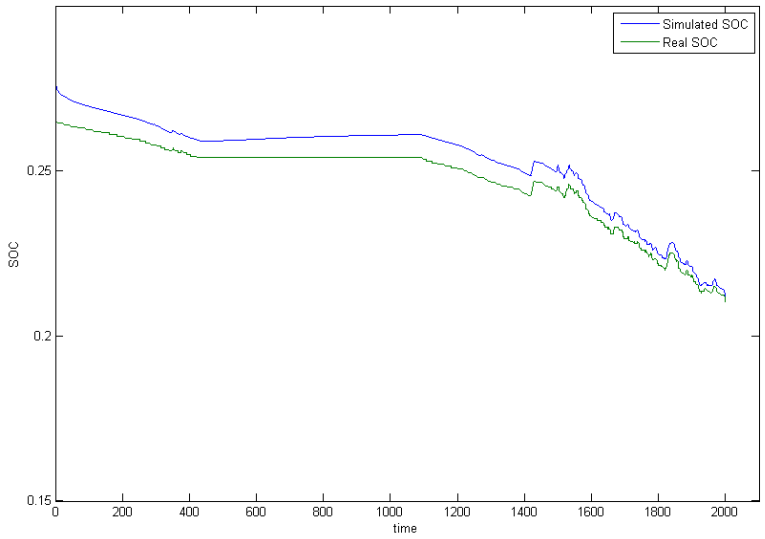


Figure 16: SOC validation: Third driving cycle

Additionally, some differences in the results may come also from the battery power limits, which may not be the same for the car's working software and the one implemented in this work.

7.2 Vehicle Hybrid Strategy Behavior

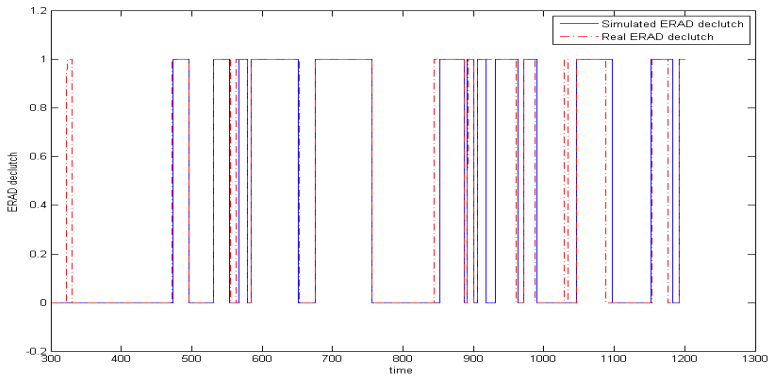
For the hybrid strategy, a working model is obtained. However the model is designed with a specific vehicle architecture in mind (along with specific components), and so this strategy is modified to fit this project's purposes.

The model is tested against real data which comes from a normal driving cycle. Vehicle speed, acceleration pedal, brake pedal and gear number are given to the model as inputs and the strategy's requests are compared against the ones generated by the model to ensure correct behavior.

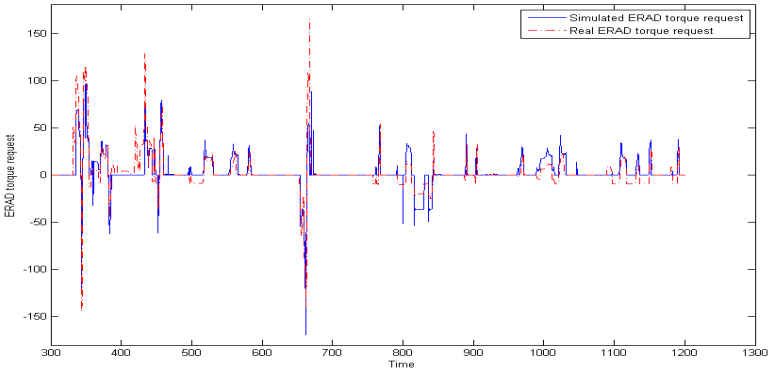
Figures 17 show some of the hybrid strategy's outputs, namely the requested torques which are important to understand how the vehicle is being propelled. Overall, the real car's hybrid strategy and the software hybrid strategy are taking similar decisions as to when to disengage the motor, as shown in figure 17a.

Figure 17b shows the torques requested from the ERAD; as can be seen, there are some differences, which are attributed to several reasons:

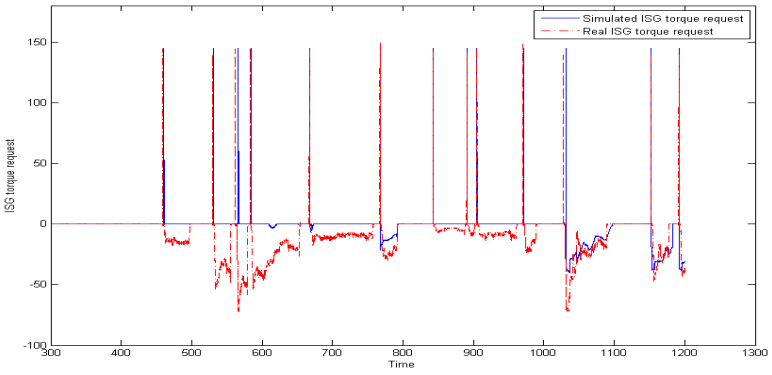
- There are differences in the machine's limits, this differences arise every time the software is modified or upgraded, and it is unknown which is the working version in the car.



(a) ERAD engagement (declutching)



(b) Requested electric machine (ERAD) torque



(c) Requested generator (ISG) torque

Figure 17: Software Hybrid Stragety Outputs

- A negative friction torque appears when the engine is shut down after operation and there is no pedal command from the user; the cause for this is not known, but it is suspected that, rather than braking the engine, the friction torque is used to charge the battery (regenerative braking), the end result is that the battery is charged, and the driver still gets a feeling of slowing down if it lets go of the pedal. Unfortunately, insufficient data means that the dynamics could not be modeled.
- The established engine control logic is a simple *on-off* control, which shuts off the engine when the ERAD is working, and viceversa; the logic is based solely on the requested wheel torques, and the specific values which trigger either the ERAD or the engine were not given, and had to be estimated by trial and error.
- There is no usable model of the internal combustion engine, so the system does not have this component; what is done instead is that the hybrid strategy's requested torque is considered to be the engine torque, and the engine is turned on or off according to the logic described in the previous item. Fortunately, the absence of an engine model does not introduce significant errors on normal driving, differences would only be observed on more aggresive driving and at high speeds.

Figure 17c shows the requested torque from the generator (ISG), this component's behavior depends only on the state of charge, turning on when there is the need to charge, giving an impulse of very large torque when a boost is required to move the car, and turning off when there is enough charge in the battery. From this information it can be deduced that the differences observed in the figure are due to differences between the SOC reported by the car, and the SOC calculated in the software model; this in turn can be attributed to the fact that there are unknown loads on the real car, and for that reason they cannot be taken into account in the software.

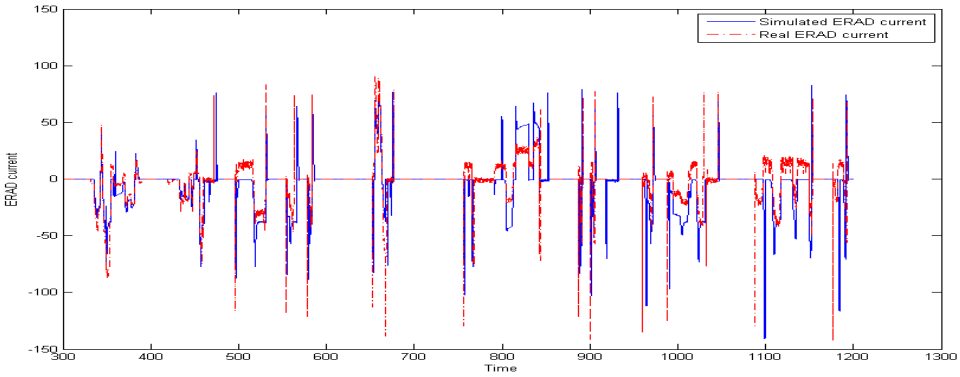
7.3 Component Behavior

Figures 18 show the output information from the main components, namely the ERAD and generator (ISG) currents, as well as the crankshaft torque. Again, the overall behavior is quite similar, and the observed differences are due to the decisions taken in the hybrid strategy (which, as discussed before, presents some differences when compared to the existing strategy in the car).

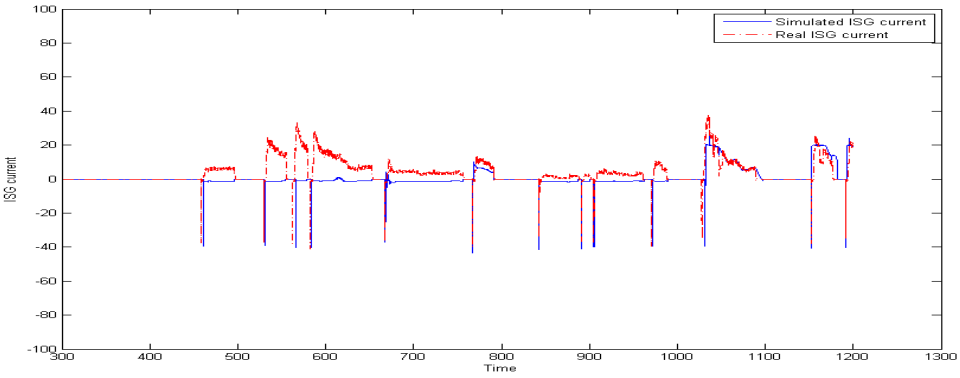
7.4 Battery Behavior

Figures 19 show the final behavior of the battery, figure 19a shows the current that would go to and from the battery cell, and figure 19b shows the comparison between the state of charge reported by the car and the one generated by the model.

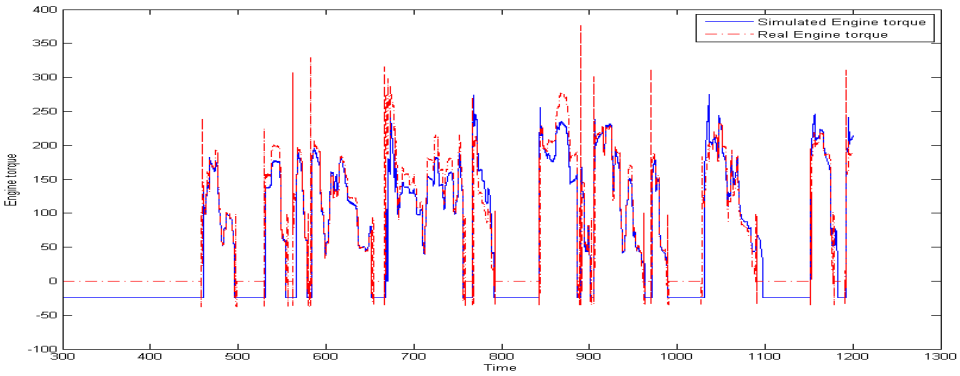
In both figures, the observed differences are due to unknown loads on the real car and for this reason, as mentioned before, they cannot be taken into account in the model. Identification of these loads is left to future work.



(a) ERAD currents.

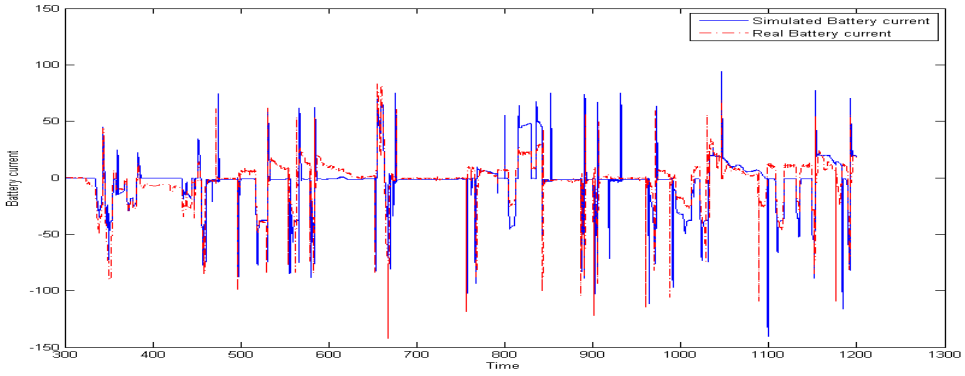


(b) ISG currents.

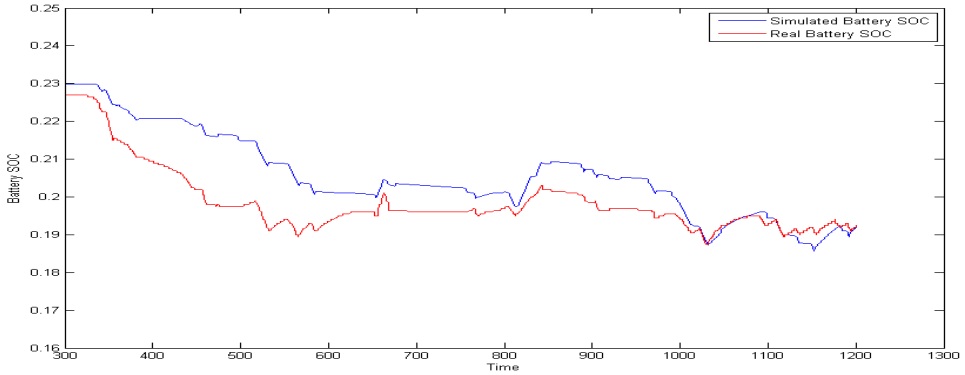


(c) Crankshaft torque.

Figure 18: Component Outputs



(a) Battery currents.



(b) Battery state of charge.

Figure 19: Battery cell's current command and SOC.

8 Conclusion

The end result of this work has been the succesful modeling and integration of the components of a HEV into a single MATLAB program, which can be compiled into an on-board computer to perform tests on a battery cell and to study its behavior. Thanks to some changes in the choice of inputs (discussed in section 6), any kind of car can be used for this purpose.

There are, however, some important differences between what is reported in the car and what the software model predicts. In general, these differences can be attributed to behavior that, due to unavailable data, could not be accounted for. These behaviors can be traced to the engine dynamics, the generator dynamics (specifically, the lack thereof) and finally on the hybrid strategy's decision-making.

A compromise had to be made between model complexity and feasibility: the complexity of any component is limited by the computing power and memory size of the ECU. Furthermore, the priority has not been to obtain accurate behavior on the components, but rather a functional integrated system that can account for the main behaviors and is easy to undersand and modify as necessary. The use of look-up tables (instead of mathematical models) was prevalent on this work.

Despite the simplifications, the model has similar behavior and it is suitable for testing. Future work, in identifying the car's dynamics and obtaining documentation of the implemented software in the real car, should result in a model that matches the behavior of the real car more accurately.

An additional conclusion reached by this work is that care must be taken when the equipment is set to read information from an actual vehicle. For example, if the equipment were to be installed on a normal, fuel-propelled car, reading the engine torque directly (to calculate the requested ERAD torque) would not be appropriate; the absence of a hybrid strategy in the car means that the engine is not controlled (in this particular case, by the *on-off* control between the engine and the ERAD) and it can lead to strange decision-making on the equipment's hybrid strategy.

The test procedures are performed to update values in the lookup tables used for SOC calculation, as well as monitor the degradation of the cell's capacity and the available power. However it is possible, through future work, to do away with the test procedures if the adaptive SOC estimation algorithm is further extended to include EKFs for capacity and internal resistance behavior over time. These extensions require information (such as hysteresis curves, for example) about the physical components.

Disturbance covariances can be formalized if information on the sensors is obtained, however it is very likely that,the covariances, will still require manual tuning in order to get acceptable output from the filter; it is often the case that the filter makes a correct but rather noisy estimation, or the reverse which is a smooth but inaccurate output, a *quick and dirty* fix is manual tuning of the covariances.

Finally, another conclusion arose from the work presented in this thesis. Each individual car is driven differently by its users, this means that the lifetime of the battery (and as important, the battery parameters) may vary from vehicle to vehicle, and since performing pulse and charge/discharge tests on each car is not feasible, the adaptive techniques for SOC estimation become more attractive.

References

[1] Pontus Svens. Development of a novel method for lithium-ion battery testing on heavy duty vehicles. Licentiate thesis, KTH Stockholm, 2011.

[2] Olle Gelin Marten Behm Pontus Svens, Johan Lindstrom and Goran Lindbergh. Novel field test equipment for lithium-ion batteries in hybrid electrical vehicle applications. *Energies*, pages 741–757, 2011.

[3] Olle Gelin. Heavy hybrid vehicle emulator for li-ion cell test. Master thesis, Linköping University, 2009.

[4] James Francfort Donald Karner. Hybrid and plug-in hybrid electric vehicle performance testing by the us department of energy advanced vehicle testing activity. *Journal of Power Sources*, pages 69–75, 2007.

[5] James Edward Francfort. Hybrid electric vehicle and lithium polymer nev testing. 2007.

[6] Lennart Ljung Torkel Glad. *Control Theory: Multivariable and Nonlinear Methods.* CRC Press, 2000.

[7] Michael Wahlstrom. Design of a battery state estimator using a dual extended kalman filter. Master thesis, University of Waterloo, Canada, 2010.

[8] Bor Yann Liaw and Matthieu Dubarry. A roadmap to understand battery performance in electric and hybrid vehicle operation. *Journal of Power Sources*, pages 366–372, 2007.

[9] Ulf Soderberg. Scania bmu software design description, 2010.

[10] Eckart Nipp. *Permanent Magnet Motor Drives with Switched Stator Windings.* PhD thesis, Royal Intitute of Technology Stockholm, 1999.

[11] Enrique L. Carrillo Arroyo. Modeling and simulation of permanent magnet synchronous motor drive system. Master thesis, University of Puerto Rico, 2006.

[12] Kwanghee Nam Hyung-Bin Ihm Gubae Kang, Jaesang Lim and Ho-Gi Kim. A mtpa control scheme for an ipm synchronous motor considering magnet flux variation caused by temperature. *Applied Power Electronics Conference and Exposition*, pages 1617–1621, 2004.

[13] Quanshi Chenb Junping Wanga, Binggang Caoa and Feng Wanga. Combined state of charge estimator for electric vehicle battery pack. *Control Engineering Practice*, page 1569–1576, 2007.

[14] Myoungho Sunwoob Jaehyun Hana, Dongchul Kima. State-of-charge estimation of lead-acid batteries using an adaptive extended kalman filter. *Journal of Power Sources*, page 606–612, 2008.

[15] Gregory L. Plett. Extended kalman filtering for battery management systems of lipb-based hev battery packs part 1. background. *Journal of Power Sources*, pages 255–261, 2004.

[16] Gregory L. Plett. Extended kalman filtering for battery management systems of lipb-based hev battery packs part 2. modeling and identification. *Journal of Power Sources*, pages 262–276, 2004.

[17] Gregory L. Plett. Extended kalman filtering for battery management systems of lipb-based hev battery packs part 3. state and parameter estimation. *Journal of Power Sources*, pages 277–292, 2004.

[18] Rui Xiong Yongli Xu Hongqiang Guo Hongwen He, Xiaowei Zhang. Online model-based estimation of state-of-charge and open-circuit voltage of lithium-ion batteries in electric vehicles. *Energy*, pages 310–318, 2012.

[19] U.S. Department of Energy Vehicle Technologies Program. Battery test manual for plug-in hybrid electric vehicles, 2010.