MODELING OF AUTONOMOUS EV VECTOR-CONTROLLED POWER CONVERSION SYSTEM FOR BATTERY MANAGEMENT SYSTEM DESIGN

GÁBOR KOHLRUSZ*[1] AND DÉNES FODOR[1]

[1]Research Institute of Automotive Mechatronics and Automation, University of Pannonia, Egyetem u. 10, Veszprém, 8200, HUNGARY

Abstract - Since the energy consumption of electric drivetrains can be optimized in an automatically controlled system, the driving ranges and efficiency of driverless electric cars can be enhanced. The analysis of a model of the electric power conversion system provides the opportunity to consider different driving circumstances, moreover, it is possible to evaluate the performance of a power conversion system when the vehicle is driven along different routes. The results provide detailed information on the transient operation of all the power modules as well as their components, and on the overall performance of the power conversion system. In this study, a permanent-magnet synchronous machine (PMSM) acts as the traction motor in the autonomously driven electric car. Besides the PMSM, the power electronics and battery are also modeled in an OrCAD PSpice circuit-simulation environment that serves as a model of an electric power conversion system for the simulation and testing of a battery management system algorithm. The battery management system and control algorithm are modeled in Simulink and can be tested together with the PSpice-modeled circuits utilizing an interconnected simulation environment. The input of the power conversion system model was a driving scenario that included uphill and downhill sections. The performance of the implemented battery management system algorithm was analyzed and evaluated.

Keywords: electric power conversion system, autonomous vehicle, battery management, interconnected simulation

1. Introduction

A driverless car is an automated system where the driving task is a self-acting function. As a result, it is possible to optimize the energy efficiency of the car in order to increase its driving range and reduce its energy consumption. The optimal operation can be achieved by applying well-designed control algorithms and battery management functions.

The electric power conversion system of an autonomous vehicle is a very complex system, especially when the electronic control unit is also considered to be part of the system. During the test procedure of such a complexly controlled electric power conversion system, several unforeseen or unexpected behaviors can occur which may lead to damage. In order to avoid damage, it is essential to identify unexpected system behaviors as early as possible.

In the early phase of development, the testing and design of these systems should be performed simultaneously to take advantage of the interconnected simulation environment. A mixed-signal simulation environment supports the analysis of controlled electric power conversion systems where digital and analogue domains can be connected and run simultaneously [1]. Simulations can be performed by taking into account practical aspects that introduce cycle times and delays into the model. In such a realistic simulation environment, the reliable design of a battery management system can be achieved.

2. Electric Power Conversion System Model

A system model, including a model of the electronic control unit, is a prerequisite to the simulation. The electric part of the system can be accurately modelled in a circuit design environment, while the digital environment and control algorithm should be implemented in an environment that is capable of running numerical computations.

The electronic components were modeled in OrCAD PSpice environment while the control algorithm was realized in MATLAB-based Simulink. The electric parts of the modeled power conversion system can be seen in Fig. 1. The system consists of a Nickel Manganese Cobalt Oxide (NMC)-type Li-ion battery, a three-phase inverter and a permanent-magnet synchronous machine as the traction motor. The modeled system is a low-power counterpart of a vehicle power conversion system assum-
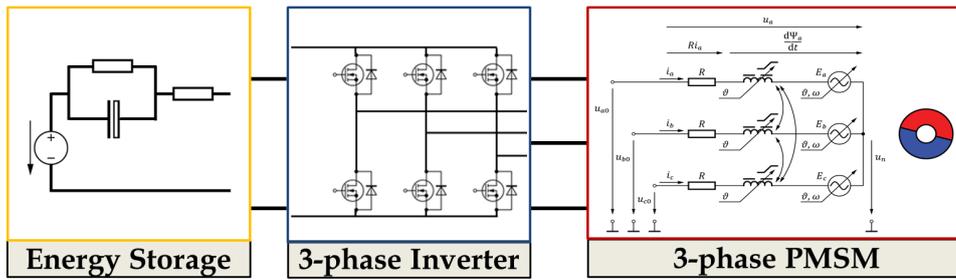
*Correspondence: kohlrusz.gabor@mk.uni-pannon.hu

Figure 1: Structure of the Modeled Power Conversion System

ing that the system behavior is independent of the power rating.

2.1 Modeling of Energy Storage

The energy storage element of the modeled power conversion system is a Li-ion battery. The modeled battery cell consists of 4 individual NMC-type Sony VTC4A 2100 mAh battery cells connected in series. As Fig.2 shows, the model of the battery contains a State-of-Charge (SoC)-dependent voltage source and three passive components.

The model of the battery cell was developed and validated by using measurement data. A fully charged Sony VTC4A 2100 mAh battery cell was discharged slowly by setting the discharge current to a C/10 value until its voltage reached the recommended minimum voltage level. The time, battery voltage and discharge current were recorded and the SoC values were determined by time-integration of the current. Using the voltage and SoC time series, the $U(\mathrm{SoC})$ function of the battery cell could be obtained by curve fitting. The best fit was achieved by

$$
\begin{aligned}
U & = 0.227 \log_{10}\left(0.1\left(\mathrm{SoC} + 3.35 \cdot 10^{-5}\right)\right) \\
& + 0.535\,\mathrm{SoC}^3 + 3.8926
\end{aligned} \tag{1}
$$

The fitted curve and the measured voltages are shown together in Fig. 3 as a function of the SoC.

In order to complete the battery model, it is also necessary to obtain the electrical parameters of the battery cell. The values of the passive components were provided by measurements which put the transient behavior of the battery cell into focus. In Table 1, the values of all three parameters can be seen.
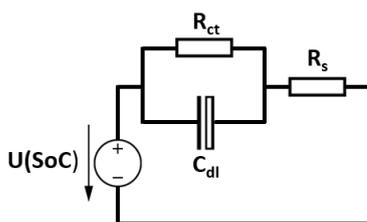

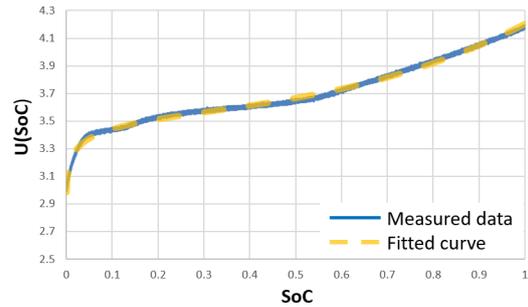*Figure 2:* Equivalent Circuit Model of the Battery

Figure 3: Approximation of the Voltage Response of the Battery

2.2 Modeling of the Three-phase Inverter

The energy conversion between the electric motor and the battery was provided by a three-phase full-bridge inverter with 2 Metal–Oxide–Semiconductor Field-Effect Transistor (MOSFET) semiconductors in each leg. The drive circuit was connected through a gate resistor to avoid an inrush current at the gate of the mosfets. The pulse-width modulated (PWM) gate-source voltages were provided by the digital controller which was implemented in the MATLAB-based Simulink environment.

A schematic diagram of the inverter can be seen in Fig. 4. As the figure shows, a shunt resistor was added to each phase wire for current sensing.

2.3 Modeling of the Three-phase PMSM

In the modeled system, the traction motor was a permanent-magnet synchronous motor. The implemented model is valid for constructions with surface-mounted magnets on the rotor and distributed winding in the wye-wound stator.

The electrical model was created by determining the voltages across the stator windings which consisted of

Table 1: Equivalent Circuit Model Parameters of the Battery

Circuit parameter	Notation	Value
Series Resistance	R_{s}	$0.1\,\Omega$
Charge Transfer Resistance	R_{ct}	$0.0052\,\Omega$
Double-layer Capacitance	C_{dl}	$0.9\,\mathrm{F}$
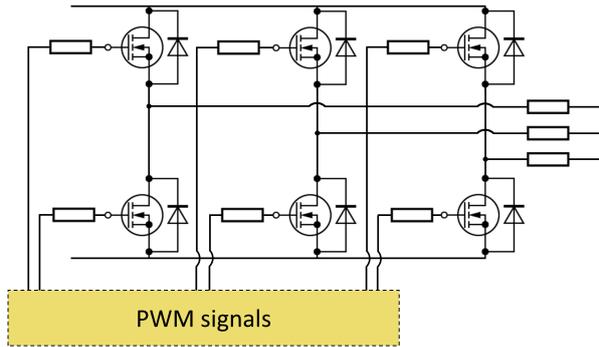
Figure 4: Three-phase Full-bridge Inverter

Table 2: PMSM Model Parameters

MOTOR parameter	Notation	Value
Phase resistance	R	$0.49\ \Omega$
Leakage inductance	L_{sl}	$33\ \mu H$
Mean value of the mutual inductances	L_{so}	$92\ \mu H$
Amplitude of inductance fluctuation	L_x	$10\ \mu H$
Amplitude of permanent magnet-created flux linkage	Ψ_{PM}	22.4 mWb
Friction coefficient	B	5.25×10^{-5} Nms
Rotor inertia	J	220 gcm^2
Number of pole pairs	z_P	2

the resistive drop, the voltage across the inductor and the back electromotive force (back EMF). In the OrCAD environment, it was quite easy to create the electric part of the motor model due to the fact that the voltage across the inductor in each phase could be represented by defining time-dependent self- and mutual inductances as passive components of the circuit, while their analytical modeling would have been complicated.

The simulation of the motor requires the implementation of the mechanical submodel as well. Papers dealing with the model of the three-phase PMSM also included the electromagnetic torque [2, 3], however, it was challenging to determine the expanded form of an expression which could be easily implemented in an OrCAD environment. In Ref. [4], a suitable form is presented for implementation in an OrCAD environment, hence this model was used in this work to obtain the mechanical model.

In the simulation, a MAXON EC-4pole 252463 type motor was considered. The parameters of the motor can be seen in Table 2.

3. Modeling of Electronic Control Unit

The Electronic Control Unit was modeled in a MATLAB-based Simulink environment which is suitable for quick algorithm and was equipped with an interface to OrCAD PSpice circuit simulation software.

A realistic electronic control unit could be modeled using a trigger-activated Simulink function which run the control algorithm when a trigger signal is received. The trigger signals were generated to fit the operating frequency of the digitally controlled discrete-time system. This solution ensured that the control algorithm only ran once during each sampling period, consequently, the Pulse Width Modulation (PWM) signals of each MOS-FET were generated only once during each sampling period. This solution is able to accurately simulate the operation of a microcontroller or even an electronic control unit.

The discrete-time control algorithm was implemented in the trigger-activated block. The structure of the algorithm can vary depending on how the difference equation is determined.

3.1 Discrete-Time Control Algorithm

The frequently used Proportional Integral (PI) algorithm was implemented to control the power conversion system. The structure of the control system is shown in Fig. 5. As the figure shows, a speed-cascaded torque control loop and a magnetic field control loop were created. This strategy is the widespread field-oriented control scheme of three-phase alternating current (AC) motors.

In the case of a PI controller, the only dynamic component of the controller is the integrator since the proportional component simply multiplies the error signal. Consequently, the proportional component can be implemented in the same form as in a continuous-time algorithm, however, the discrete-time form of the integrator has to be determined by using a Z-transform. The transformation can be conducted using different formulae to approximate the continuous-time system.

An easy way to determine the discrete-time form of the integrator is to use the backward Euler method. Using this method, the value of the integral at time j is

$$Y_j = Y_{j-1} + e_j T T_I \tag{2}$$

where Y_j denotes the result of the integral at time j, Y_{j-1} represents the result of the integral at time $j-1$, e_j stands for the error signal at time j, T is the sampling time and T_I denotes the integral gain.

A drawback of this method is that a continuous-time system could be unstable in its discrete-time form using the backward Euler method [5].

The bilinear transform (also known as Tustin's method) could be a more reasonable technique to obtain the discrete-time representation of a system since it preserves stability. The value of an integral at time j is determined by this method as follows:

$$Y_j = \frac{Y_{j-1} + \left(e_j + e_{j-1}\right) T T_I}{2} \tag{3}$$

where Y_j denotes the result of the integral at time j, Y_{j-1} stands for the result of the integral at time $j - 1$, e_j represents the error signal at time j, e_{j-1} is the error signal
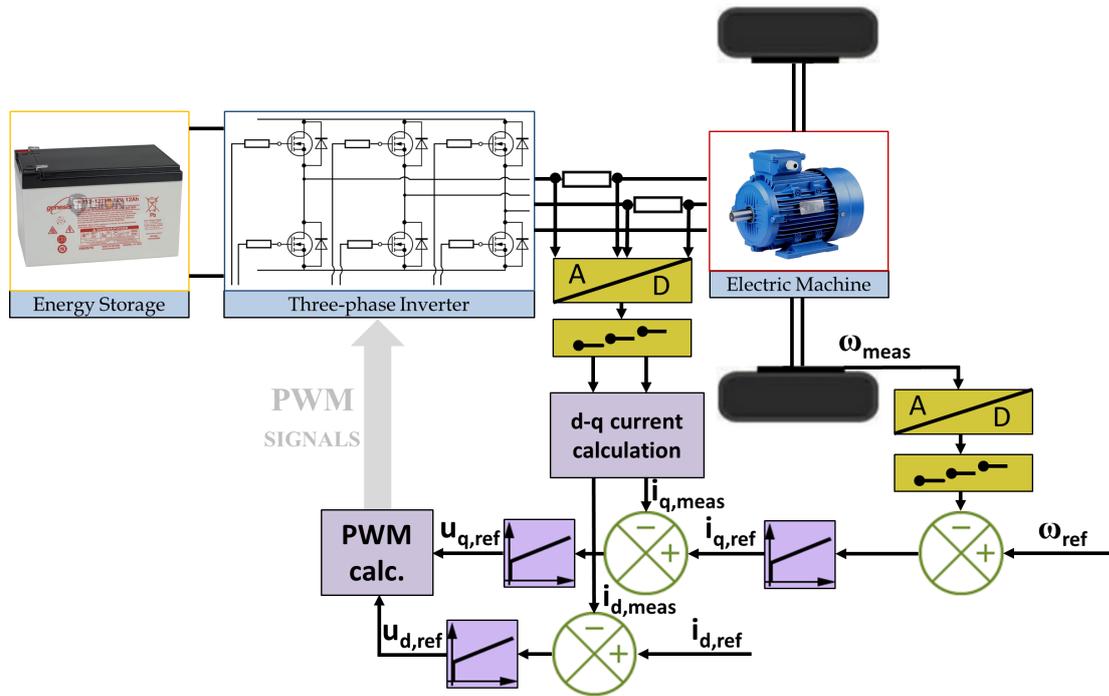
Figure 5: Vector-Controlled Power Conversion System of an Electric Vehicle

at time $j - 1$, T denotes the sampling time and T_{I} stands for the integral gain.

The realization of a controller that applies the backward Euler method is easier and this form generally provides a stable operation, thus in the implemented model of a controller, the integrator was approximated by Eq. 2.

3.2 Design of the Controller Parameters

The implemented controllers were PI controllers in parallel form, that is the integral and proportional terms were independent of each other.

One of the simplest controller design procedures is the pole-zero cancellation but this method provides a frequency bandwidth for the controller that is identical with that of the controlled system resulting in poor performance occasionally. This problem is particularly severe when a system with a large time constant is controlled by a slow integrator with the same time constant as the system [6].

A better controller performance can be achieved by another analytical tuning method where the closed-loop poles are chosen to obtain an arbitrarily determined closed-loop Transfer function. The closed-loop is shaped by utilizing some predefined parameters such as the desired damping factor and bandwidth of frequencies.

The parameters of the PI controller can be computed as

$$K_{\mathrm{c}} = \frac{2\xi\omega_0\tau - 1}{K} \qquad (4)$$

where K_{c} denotes the proportional gain of the controller, ω_0 stands for the natural frequency of the desired closed-loop system, ξ represents the desired damping factor of

the second-order closed-loop system, τ is the time constant of the controlled first-order subsystem, and K denotes the gain of the controlled first-order subsystem.

The value of the integral gain is determined as

$$T_{\mathrm{I}} = \frac{\omega_0^2 \tau}{K} \qquad (5)$$

where ω_0 denotes the natural frequency of the desired closed-loop system, τ stands for the time constant of the controlled first-order subsystem, and K represents the gain of the controlled first-order subsystem.

The damping ratio was chosen to ensure an overshoot of approximately 1%, therefore, for the computations, $\xi = 0.85$ was applied. The desired natural frequency of the closed-loop system (ω_0) was obtained by determining the desired settling time of the controlled variable by considering a desired error band:

$$\omega_0 = -\frac{\ln p}{\xi t_{\mathrm{b}}} \qquad (6)$$

where p denotes the difference between the reference signal and the value of the controlled variable as a percentage of the reference value, ξ stands for the desired damping factor of the second-order closed-loop system, and t_{b} represents the period of time until the controlled variable is expected to reach the desired error band [7–9].

3.3 Implementation of the Smith predictor

When a discrete-time unit controls a system, a time delay is always present. During each time period, some measurements serve as the basis of control signal generation.

However, the calculated control signals are only realized as controller outputs during the following time period, which leads to a delay of one period in duration.

Compensation for this one-period delay can be solved by implementing a Smith predictor. The predictor is an algorithm which modifies the actual measurement signal by extrapolation based on former measurements. The implementation of a Smith predictor requires the difference equation of the controlled variable as a function of the controller output to be obtained. Based on the difference equation, the measurement signal was compensated by predicting the measurement value of the following time period:

1. At time T_1, measurement signal $\hat{u}_1$ is considered for the computation of controller output y_2 which will only take effect at time T_2.

2. From y_2, it is possible to compute the predicted measurement value $\hat{u}_2$ that can be used to obtain $\hat{u}_1 - \hat{u}_0$

3. $\hat{u}_1 = u_1 + (\hat{u}_1 - \hat{u}_0)$ is the corrected measurement signal, which is used to compute y_2 as has already been stated in Step 1:

where u_1 denotes the measured value of the controlled variable at time T_1 modified by the correction term, y_2 stands for the controller output at time T_2, $\hat{u}_2$ represents the predicted value of the measurement at time T_2, and $\hat{u}_1$ and $\hat{u}_1$ are the computed values of the controlled variable at times T_1 and T_0, respectively.

In Step 2, the difference equation of the controlled subsystem is used for the computation of the predicted $\hat{u}_2$ at time T_2. In the case of the current controllers, the prediction algorithm is

$$\hat{i}_2 = (T_2 - T_1)\frac{R}{L}\left(\frac{U_2}{R} - \hat{i}_1\right) + \hat{i}_1 \qquad (7)$$

where the current $\hat{i}_2$ is the predicted value of the measurement signal at time T_2, which was the basis of the calculation of the correction term $\hat{i}_1 - \hat{i}_0$. The measured current i_1 could be corrected as

$$\hat{i}_1 = i_1 + (\hat{i}_1 - \hat{i}_0). \qquad (8)$$

The predictor was the same in both the i_d and i_q current controllers, only the parameters were different as L_d was substituted for Eq. 7 in case of the i_d control loop and L_q was substituted for Eq. 7 in case of the i_q control loop. The Smith predictor in the cascade control loop requires the difference equation of the angular velocity as a function of the current i_q:

$$\hat{\omega}_2 = (T_2 - T_1)\frac{1.5z_{\mathrm{P}}\Psi_{\mathrm{PM}}i_{q,1} - B\hat{\omega}_1 - M_{\mathrm{T}}}{J} + \hat{\omega}_1 \quad (9)$$

The measured speed ω_1 could also be corrected BY calculating the values $\hat{\omega}_1$ and $\hat{\omega}_0$ from $\hat{\omega}_2$:

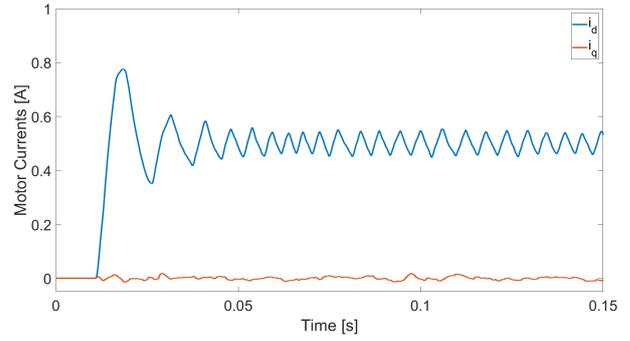$$\hat{\omega}_1 = \omega_1 + (\hat{\omega}_1 - \hat{\omega}_0) \qquad (10)$$

Figure 6: Testing of the i_d and i_q current controllers

4. Testing of the Controlled Power Conversion System Model

The testing was performed with an operating frequency of 25 kHz in the control loop. The current controllers were designed to provide a settling time of $t_{\mathrm{b}} = 2$ ms with $p = 0.02$ that equated to an error of 2% which resulted in the desired frequency bandwidth of $\omega_0 = 2301$ rad/s. By taking into consideration the frequency bandwidth, the calculated controller parameters were

$$K_{\mathrm{c},i_d} = 0.12 \qquad (11)$$

$$T_{\mathrm{I},i_d} = 824 \qquad (12)$$

for the d-axis current controller and

$$K_{\mathrm{c},i_q} = 0.238 \qquad (13)$$

$$T_{\mathrm{I},i_q} = 985 \qquad (14)$$

for the q-axis current controller.

The testing of the controllers was performed without the implementation of the cascaded speed control loop. A reference current was set up for both current loops, d and q. The results can be seen in Figs. 6 and 7.

It is shown in Fig. 6 that the current i_d overshot the reference value of 0.5 A and started to oscillate around the setpoint with an amplitude of 50 mA, which is 10% of the reference value. The current i_q tracked its setpoint of 0 A with an inappreciable error. The results show that the integral gain of the current i_d should be decreased, therefore, the controllers were tested with redesigned i_d
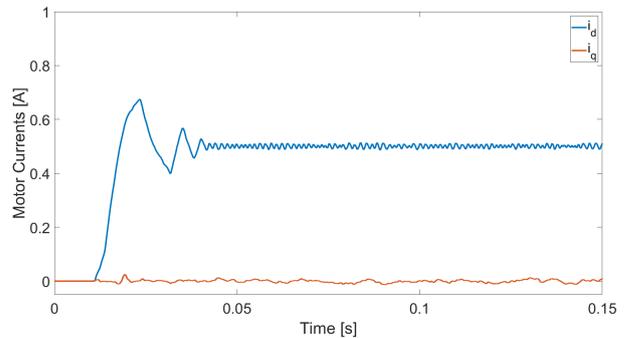

Figure 7: Results of the redesigned current controllers

(a) Speed of the Electric motor

(b) Performance of the d-axis current controller with and without the Smith predictor
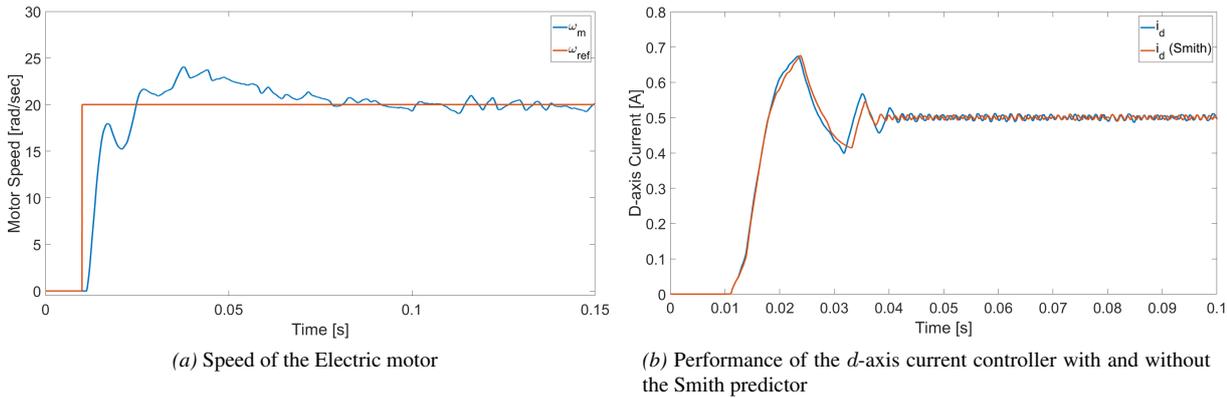
Figure 8: Simulation results of the controlled system using the Smith predictor in the current controllers

controller while the i_q controller parameters remain unchanged.

$$K_{c,i_d} = 0.12 \quad (15)$$

$$T_{I,i_d} = 275 \quad (16)$$

for the current controller d.

In Fig. 7, the results for the d-axis controller testing are presented. The performance of the redesigned d-axis current controller i_d was enhanced. The controlled variable reached the setpoint and stabilized after approximately $2\,\mathrm{ms}$.

The angular velocity ω was controlled by a cascaded PI controller in the q-axis current loop. The controller parameters were obtained by determining the desired frequency bandwidth which provided a settling time of $t_{\mathrm{b}} = 0.05\,\mathrm{s}$ and $p = 0.05$, which equated to an error of 5%:

$$K_{c,\omega} = 0.038 \quad (17)$$

$$T_{I,\omega} = 1.627 \quad (18)$$

Fig. 8a shows the controlled motor-speed signal which has a reference value of $20\,\mathrm{rad/s}$. It can be seen that the speed signal overshot slightly and stabilized after $0.065\,\mathrm{s}$ which was acceptable since the designed settling time was defined as $t_{\mathrm{b}} = 0.05\,\mathrm{s}$.

4.1 Testing of the Smith predictor

The predictor was only implemented for the current controllers since the speed tracking performance of the system was satisfactory without the application of a predictor. The implemented predictor was tested under the same conditions as the current controllers, that is the current references i_d and i_q were set at $0.5\,\mathrm{A}$ and $0\,\mathrm{A}$, respectively. Fig. 8b shows the controller performance with and without the Smith predictor.

In Fig. 8b, it can be seen that the controller performed better when the Smith predictor modified the error in the input signal of the controller by correcting the measured current signal. The controlled current i_d stabilized faster and in a steady state it could be observed that the PWM-related high-frequency oscillation of the current occurred

with a reduced amplitude compared to when the Smith predictor was not applied. The behavior of the current signal i_q did not change significantly since its reference value was set at $0\,\mathrm{A}$ and no dynamic changes in the reference signal needed to be handled.

5. Battery Management System Design

The controlled power conversion system was validated by tests. The created model of the complex system is suitable for simulations that are intended to analyze the operation of the system. By analyzing the operation of the power conversion system, the design aspects of a battery management system can be summarized. Simulation of the power conversion system can be a suitable procedure to validate the operation of the designed battery management system under realistic conditions.

5.1 Implementation of a Battery Management System

When a battery management strategy is implemented, it influences both the hardware and software components of the supervised system. Battery management solutions actuate in order to ensure efficient system operation as well as keep the system safe. If the management algorithm is used to ensure the system operates safely, all the avoided situations should be summarized during the design phase. Moreover, for each undesirable situation, a hardware solution has to be designed, which can be activated when necessary.

In this study, a battery overcharge protection system was analyzed as a battery management system. For this purpose, the power electronics had to be extended by additional power switches that were capable of changing operation mode when overcharging occurred (see Fig. 9). The implementation of the battery management system also included augmentation of the software.

In Fig. 9, a power semiconductor (M_1) can be seen that ensures an alternative path for the current when the battery cannot be charged. In this case, the recuperated energy is dissipated into a resistor. When switch M_1 is
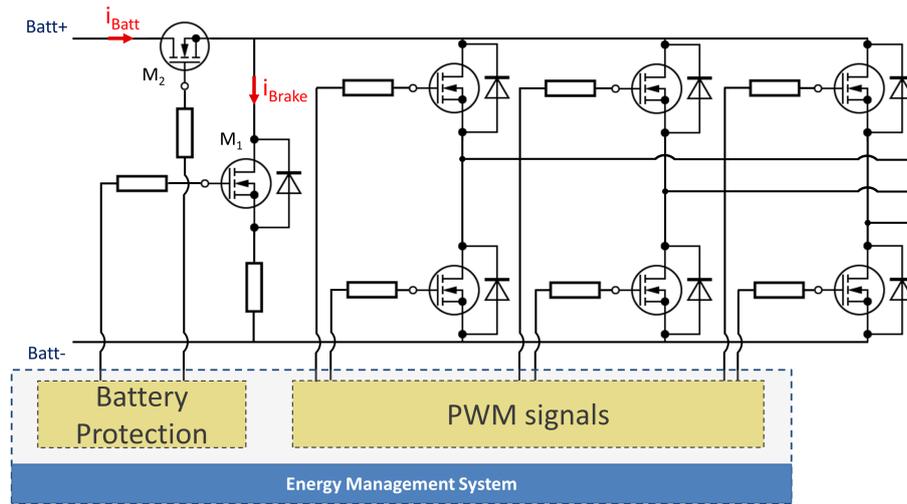
Figure 9: The analyzed battery management system

turned on, another switch (M_2) becomes responsible for decoupling the battery, otherwise the battery would be discharged through the resistor. In certain situations, generation of the PWM signal is also affected by the battery management algorithm. In this case, the PWM signals are generated in the same way whether the battery protection circuit operates or not.

5.2 Simulation of a Driving Cycle of a Vehicle

The simulation of a driving cycle of a vehicle provides the opportunity to observe the efficiency of the battery management strategy of the power conversion system and its interactions with the components of the system. The results serve as a basis for implementing different battery management strategies or modifying the implemented one.

A driving cycle can be simulated by defining different load torques of the traction motor. When an uphill or downhill stretch of road should be represented, the motor is loaded with a positive or negative load torque, respectively. If the gradients of the uphill and downhill sections of road are unchanged, the load torque remains constant.

A simulation of $5.5\,\text{s}$ in duration was performed. The simulated driving cycle consisted of uphill, downhill and flat sections (see Fig. 10). In the picture, the load torques of the PMSM can also be seen below the corresponding sections of road. During the driving cycle, the speed of the vehicle was expected to be constant, hence the angular velocity was assumed to also be constant at $20\,\text{rad/s}$. At the beginning of the simulation, it was assumed that the battery was fully charged.

The results of the simulation can be seen in Figs. 11 and 12. In Fig.11, the speed of the traction motor can be seen throughout the driving cycle. It can be observed that the changes in load torque influenced the speed of the motor, however, the controller promptly compensated for the transient overshoots.
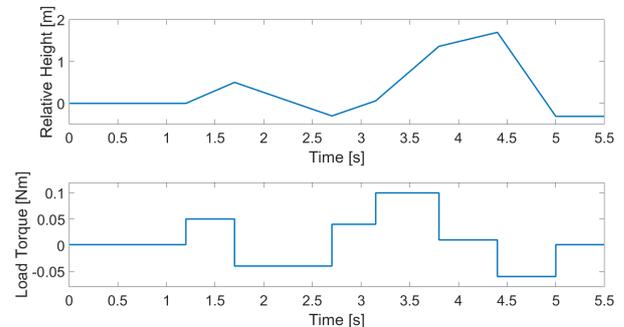

Figure 10: Simulated vehicle path and related load torques

The currents of the power conversion system are presented in Fig. 12. The braking current and the current of the battery are also shown in Fig. 9. In Fig. 12, it can be noted that the q-axis current controller achieved precise and fast reference tracking when the system was perturbed by changes in load torque. If the reference current i_q was negative, the energy was recuperated, that is the traction motor was in generator operation mode. Since the battery was fully charged, energy must have been dissipated into the braking resistor. In the simulation, two cases when the battery management SYSTEM activated the braking circuit can be seen in the figure after $T_1 = 1.7\,\text{s}$ and $T_2 = 4.4\,\text{s}$. When the current flowed through the resistor the battery current was zero due to decoupling performed by switch M_2. In the figure, a current spike can be easily observed when the battery was connected to the circuit after reference current i_q became positive.

6. Conclusions

This paper presents models of the power conversion system of an autonomous electric vehicle as well as the electronic control unit. An interconnected simulation environment enables an integrated model to be simulated. This kind of simulation is useful for the comparative performance analysis of different control algorithms as well
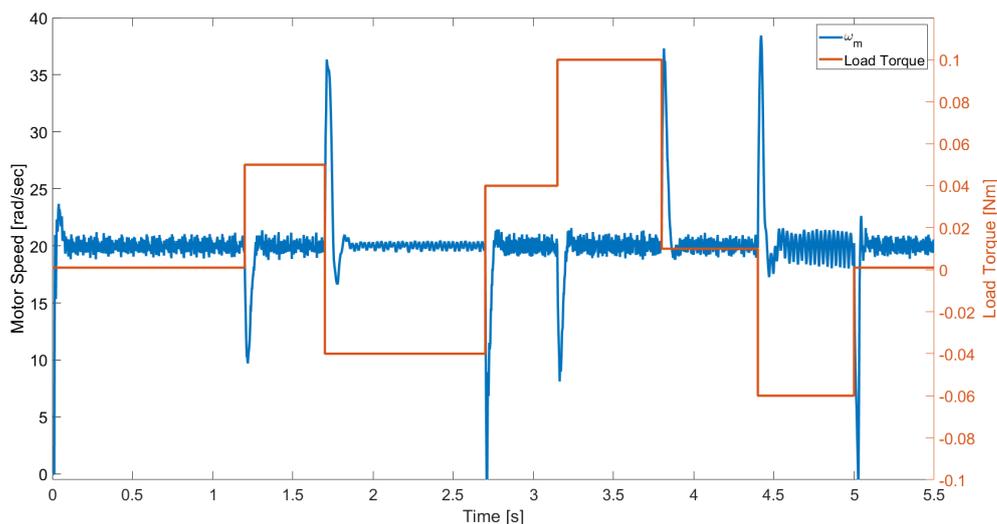
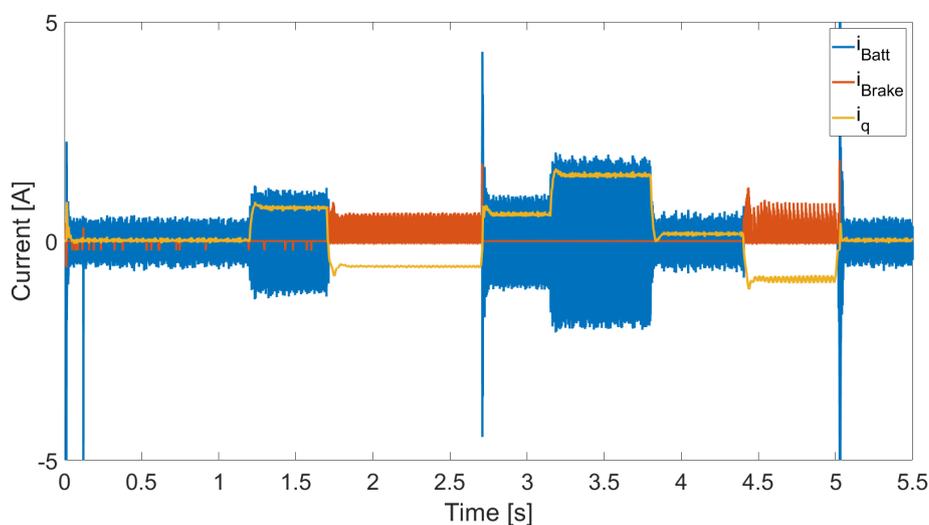*Figure 11:* Simulated motor speed during the driving cycle of a vehicle

Figure 12: The simulated currents of the power conversion system with BMS

as system level analysis of interactions between different system modules and components.

A power conversion system model is very complex and it is hard to predict interactions between components of the system following dynamic changes at system inputs. Furthermore, the validation of the controllers is only plausible if the simulation model includes all relevant components of the system and all significant factors such as sampling of the measured data or time delays in the control loop. The simulation is very useful if the operation of the hardware level system is the focus of observations and analyses.

However, the simulation of the top-level management system of an autonomous electric vehicle is not feasible due to lengthy simulation times. In order to analyze the performance of different controllers, the simulation must be run with a sampling time of $2\ \mu$s or less, while the effectiveness of a higher-level management system can be measured after a few hours of driving. In cases when the analysis of the efficiency of the management system takes into consideration the performance of the digital controller, it is essential to identify a modeling procedure which provides a model of the system that is suitable for long-time simulation.

Acknowledgements

REFERENCES

[1] Kohlrusz, G.; Csomós, B.; Enisz, K.; Fodor, D.: Electric energy converter development and diagnostics in mixed-signal simulation environment, *ACTA Imeko*, 2018, **7**(1), 20–26 DOI: 10.21014/acta_imeko.v7i1.512

[2] Hang, J.; Zhang, J.; Cheng, M.; Huang, J.: On-line Interturn Fault Diagnosis of Permanent Magnet

Synchronous Machine Using Zero-Sequence Components, *IEEE Transactions on Power Electronics*, 2015, **30**(12), 6731–6741 DOI: 10.1109/tpel.2015.2388493

[3] Qi, Y.; Bostanci, E.; Gurusamy, V.; Akin, B.: A Comprehensive Analysis of Short-Circuit Current Behavior in PMSM Interturn Short-Circuit Faults, *IEEE Transactions on Power Electronics*, 2018, **33**(12), 10784–10793 DOI: 10.1109/tpel.2018.2809668

[4] Kohlrusz, G.; Szalay, I.; Fodor, D.: OrCAD PSpice Implementation of a Realistic Three-Phase PMSM Model for Diagnostic Purposes, San Diego, CA, USA, 2019, 372–376 DOI: 10.1109/IEMDC.2019.8785371

[5] LeVeque, R. J.: Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems (Society for Industrial and Applied Mathematics, Philadelphia, PA, USA), 2007 ISBN: 978-0-898716-29-0

[6] Åström, K. J.; Hägglund, T.: PID controllers : theory, design, and tuning (2nd edition) (International Society for Measurement and Control, Research Triangle Park, N.C., USA), 1995 ISBN: 1556175167 9781556175169

[7] Kuo, B. C.; Golnaraghi, F.: Automatic Control Systems (John Wiley and Sons Inc., New York, NY, USA), 2002 ISBN: 0470048964

[8] Katsuhiko, O.: Modern Control Engineering, (5th Edition) (Pearson), 2009 ISBN: 0136156738

[9] Guzmán, J. L.; Moreno, J. C.; Berenguel, M.; Moscoso, J.: Inverse pole placement method for PI control in the tracking problem, in *IFAC-PapersOnLine*, **51**(4), 406–411 DOI: 10.1016/j.ifacol.2018.06.128