Preprints, 10th IFAC International Symposium on
Advances in Automotive Control
Columbus, Ohio, USA, August 28-30, 2022

Processor-In-the-Loop Demonstration of MPC for HEVs Energy Management System

Luca Cavanini*, Pawel Majecki*, Mike J. Grimble*,

Lakshmy Vazhayil Sasikumar, Richard Li***, Curt Hillier*****

**Industrial Systems and Control Ltd, Glasgow, UK*
(e-mail: {l.cavanini, pawel, m.grimble}@isc-ltd.com)

*** NXP Semiconductors, East Kilbride, UK*
(e-mail: lakshmy.vazhayilsasikumar@nxp.com)

**** NXP Semiconductors, Austin, USA*
(e-mail: { richard.li_4, curt.hillier}@nxp.com)

Abstract: An Energy Management System for Hybrid Electric Vehicles is described based on a Model Predictive Control solution. This is implemented using a Processor-In-the-Loop simulation running on a GreenBox II development board from NXP Semiconductors. The hybrid vehicle considered contains an internal combustion engine and an electric motor in a parallel configuration. The MPC design involves a Linear Parameter-Varying model to approximate the nonlinear vehicle model and provide a simpler algorithm for implementation. The control policy was integrated into the GBII control board and assessed in simulation with the processor in the loop. The very promising performance of the proposed predictive controller in terms of mileage and battery degradation is compared with the well-established Equivalent Consumption Minimization Strategies.

Keywords: Energy management, hybrid electric vehicle, model predictive control, processor-in-the-loop.

1. INTRODUCTION

The control of Hybrid Electric Vehicles (HEVs) is a topic attracting the attention of governments and industry for minimizing energy and reducing pollution (Horrein, L., Bouscayrol, A., Cheng, Y., & Dumand, C., 2015). This has been an active area of research and a number of Energy Management Systems (EMSs) have been proposed. These have involved different optimal control methods which follow naturally from the formulation of the HEVs energy control problem.

The early EMS methods belong to the class of rule-based control strategies (Johnson, Valerie H., Keith B. Wipke, and David J. Rausen, 2000). These policies were considered due to their low memory requirements and computational burden that enabled implementation on common automotive powertrain control units. However, they have been affected by many issues, including the architecture dependent structure of the algorithm which does not enable designs to be easily modified for different vehicle models. The effort required for calibration is therefore large since each system needs careful tuning to achieve fuel economy improvement.

Due to these problems, optimal methods were soon applied for EMS control. The so-called Equivalent Consumption Minimization Strategy (ECMS), that is equivalent to invoking Pontryagin's Minimum Principle (PMP), and its Adaptive version (A-ECMS), soon became a standard for HEV energy management systems control (Musardo, C., Rizzoni, G., Guezennec, Y., & Staccia, B, 2005). These methods involve limited computational and calibration complexity. They also use a well-established optimization approach which guarantees a certain level of performance if the modelling assumptions are met (Wu, J., Zou, Y., Zhang, X., Liu, T., Kong, Z., & He, D, 2019).

Recently different automotive companies and research centers have focused their attention on a more efficient version of optimal EMS based on Model Predictive Control (MPC) (Yan, F., Wang, J., & Huang, K., 2012). An optimal MPC control design for multivariable systems uses a model to predict the states of the plant over a moving horizon and can explicitly include input and output constraints in the controller design (Camacho, Eduardo F., and Carlos Bordons Alba, 2013). The MPC paradigm has been successfully applied to many different problems in several control domains (Cavanini, L., Ippoliti, G., 2018). However, one of the major obstacles to the use of MPC in embedded systems is its relatively high computational cost, mainly due to the need to solve a constrained optimization problem online. This is also the case in the MPC-based EMS solutions for hybrid vehicles. The problem is explored in the following using Processor-In-the-Loop (PIL) implementation of an MPC running on the new GreenBox II (GBII) prototyping control board.

The computational complexity of the MPC increases with the number of states and the nonlinearities in the model used for design. This results in a more complex and larger optimization problem to be solved on-line within the sampling time (Cavanini, Luca, Gionata Cimini, and Gianluca Ippoliti, 2018). In the proposed approach, the nonlinear model describing the controlled system is approximated by a Linear Parameter-Varying (LPV) model (Grimble, M. J. and Majecki, P., 2020)..This results in the MPC optimization problem having a Quadratic Programming (QP) form. It requires less memory and results in less computational complexity than the full nonlinear MPC solution (Cavanini, L., Cimini, G., & Ippoliti, G, 2017). The LPV model is updated at each time instant given the time-varying parameters representing the operating conditions. The solution of the resulting LPV-MPC optimization problem provides the optimal control policy designed to improve the HEV fuel economy and reduce battery degradation and aging. To verify the effectiveness of the proposed MPC solution running on the GBII control board, it has been tested using a set of standard driving cycles.

The paper is structured as follows. Section II describes the HEV model, Section III presents the proposed MPC-EMS approach, Section IV reports test results, and Section V concludes the paper.

2. HYBRID ELECTRIC VEHICLE MODEL

In this section, the nonlinear simulation model of the HEV is presented. This model is based on the Mathworks® HEV P4 Reference Application (The MathWorks, 2019). The vehicle powertrain has a parallel architecture, to control the Internal Combustion Engine (ICE) and the Electric Motor (EM) independently (Serrao, L., Onori, S., & Rizzoni, G., 2011). The hybrid powertrain structure is illustrated in Figure 1, showing also the EMS. The EMS defines the torque split between the engine (T_{ICE}) and the electric machine (T_{EM}), given the reference torque (T^{ref}) demanded by the driver to control the vehicle speed (v_{veh}). These signals, representing information flow, are shown in red in Figure 1.

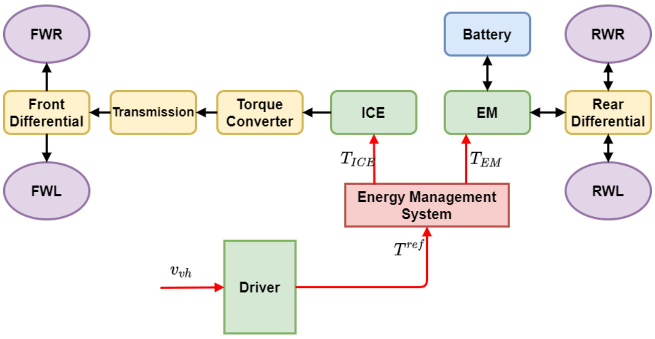

Fig. 1. Hybrid Electric Vehicle Architecture

The ICE generates the torque driving the Forward Right (FWR) and Left (FWL) wheels. The engine and the wheels are connected by the front differential, transmission and torque converter. The black arrows connecting these components represent the energy flow from the power generator (ICE) to the wheels. At the back, the rear right (RWR) and left (RWL) wheels are driven by the Electric Motor (EM) via the rear differential. The EM is connected to the battery that is the electric energy storage system. Because of the possibility to recharge the battery through regenerative braking, the arrows connecting these components are bidirectional. The EM generates/absorbs electrical energy, as demanded by the control torque signal T_{EM} defined by the EMS. The different components of the HEV are described in the following sections.

2.1 Vehicle Dynamics

The longitudinal motion of the vehicle is described by a differential equation that involves the traction force (F_{trac}), given by the powertrain and brake forces (F_{pwt}, F_{brake}), and the disturbance forces ($F_{roll},\ F_{aero},\ F_{grade}$), as follows:

$$M_{veh}\frac{dv_{veh}}{dt} = F_{trac} - F_{roll} - F_{aero} - F_{grade} \tag{1}$$

with

$$F_{trac} = F_{pwt} - F_{brake} \tag{2}$$

$$F_{roll} = (c_{r0} + c_{r1}v_{veh})M_{veh}g\cos(\delta) \tag{3}$$

$$F_{aero} = \frac{1}{2}\rho_{air}A_f C_d v_{veh}^2 \tag{4}$$

$$F_{grade} = M_{veh}g\sin(\delta) \tag{5}$$

The parameters of the dynamic vehicle model are obtained from (The MathWorks, 2019). Their main values are collected in Table 1.

Table 1. Vehicle Dynamics Parameters

Symbol	***Parameter***	***Value***	***Unit***
M_{veh}	Vehicle Mass	1623	kg
g	Gravity acceleration	9.80665	m/s^2
ρ_{air}	Air density	1.25	kg/m^3
C_d	Aerodynamic drag coeff.	0.25	
A_f	Front side vehicle area	2.46	m^2
$c_{r0} = c_{r1}$	Rolling resistance params.	0.00001	

2.2 Traction and Brake Forces

The traction force that moves the vehicle is given by the sum of forces (traction and brake) provided by each wheel. Since the front wheels are driven by the ICE and the rear wheels by the EM, the total traction force is given by the sum of both components, such that:

$$F_{trac} = F_{tracEM} + F_{tracICE} \tag{6}$$

where F_{tracEM} is the force provided by the EM and $F_{tracICE}$ it the force given by the ICE. The torques provided to the front and rear axles are assumed to be equally split between each wheel, while the brake torque generated by each brake is considered 25% of the overall brake torque required. The action of the EM and ICE on each wheel can be expressed in terms of the torque provided (this is useful for combining the model with other subsystems presented in the next sections):

$$F_{trac} = \frac{\sum_{i=1}^{4}(T_{W,i} - T_{brake,i})}{R_{wh}} \tag{7}$$

with $T_{W,i}$ the torque provided by the i-th wheel, $T_{brake,i}$ the braking torque provided by the related i-th brake (defined as the difference between the EM negative torque mapped on the i-th wheel and the reference negative torque) and R_{wh} the radius of the wheel. The rotational speed of wheels (ω_{wh}) is computed with respect to the vehicle speed as:

$$\omega_{wh} = {}^{v_{veh}}/_{R_{wh}}. \tag{8}$$

The traction force driving the vehicle is subject to max/min constraints, defined according to the road/tire friction coefficient k_{Fric} and the vertical weight force acting on the vehicle F_z, such that:

$$-F_z k_{Fric} \leq F_{trac} \leq F_z k_{Fric}. \tag{9}$$

The set of parameters describing this subsystem is collected in Table 2.

Table 2. Tire and Brake Parameters

Vehicle Dynamics Parameters			
Symbol	*Parameter*	*Value*	*Unit*
R_{wh}	Loaded wheel radius	0.327	m
k_{Fric}	Road/tire friction coefficient	0.85	
F_z	Weight Force	$M_{veh} \cdot g$	N

2.3 Differentials

The differential performs a torque split action between the two wheels connected to each axle with respect to the transmission input torque. There are two differentials in the vehicle, serving the front and the rear axle, and driven by the ICE and EM, respectively. The front differential has the differential ratio $N_F = 3.32$ and the efficiency factor $\eta_{Fdiff} = 0.98$. The rear differential has the differential ratio $N_R = 4.1$ and the efficiency $\eta_{Rdiff} = 0.98$.

2.4 Transmission

The transmission is positioned between the torque converter and the differential of the front axle, to enable axle rotational speed and torque transmitted to be changed. The efficiency of the transmission is computed according to a 4-dimensional look-up table having as inputs the axle torque T_{in}, the axle speed ω_{in}, the temperature T_{Temp} (assumed constant) and the $Gear$. The transmission ratio is considered time-varying with respect to the gear which represents a control input from an external transmission control system. The model of the transmission system is given by the following equation:

$$T_{out} = N_{trans}(Gear)\eta_{tras}(k)T_{in} \tag{10}$$

with T_{out} the output torque, T_{in} the input torque, $\eta_{trans}(k)$ the transmission efficiency and $N_{trans}(Gear)$ the transmission ratio given by a pre-defined look-up table.

2.5 Torque Converter

The torque converter changes the input torque provided by the ICE to a scaled output torque according to the Torque Ratio TR and the Speed Ratio SR:

$$TR \cdot T_{in} = T_{out} \tag{11}$$

$$SR \cdot \omega_{in} = \omega_{out} \tag{12}$$

where ω_{in} and ω_{out} are the torque converter input and output speeds, respectively. Both the torque and speed ratios are time-varying parameters that depend on the instantaneous values of both inputs and outputs. In the simulation model, such relationships are represented by different look-up tables.

2.6 Battery

The battery model represents a lithium-ion battery system parameterized according to manufacturer's data. It includes the open-circuit voltage and internal resistance parameters. The manufacturer discharge characteristic by temperature data is considered. The battery model is based on a look-up table and equivalent OCV-R model as detailed in (The MathWorks, 2019).

2.7 Electric Motor

The electric machine was modelled by a look-up table representing the behavior of the system. The inputs for the system model are the battery voltage, the EM speed ω_{EM} computed from the vehicle speed considering the effect of the rear axle differential:

$$\omega_{EM} = \frac{v_{veh}}{R_{wh}} N_R \tag{13}$$

and the EM torque demand provided by the control system. The model outputs are the battery current and the mechanical torque provided by the EM.

2.8 Internal Combustion Engine

The ICE is modelled by a set of look-up tables describing the engine performance over different operating conditions. The input to the model is the required torque signal, filtered by a low-pass filter representing the throttle dynamics. The torque output is combined with the accessory components load, also powered by the engine. The ICE speed is computed from the vehicle's speed feedback, scaled according to the wheel radius and the transmission system ratio given by the front axle differential ratio N_F, the gear ratio $N_{trans}(Gear)$ and the conversion factor between rad/s to rpm rotation speed $k_{rad/s2rpm} = 9.5493$

$$\omega_{ICE} = \frac{v_{veh}}{R_{wh}} N_F k_{rad/s2rpm} N_{trans}(Gear). \tag{14}$$

3. MODEL PREDICTIVE CONTROL APPROACH TO EMS PROBLEM

In this section, the MPC policy representing the EMS for the HEV is presented. Initially, the design model in LPV form is presented, then the system constraints are defined that represent physical limits for the powertrain. Finally, the MPC control law is described.

3.1 Linear Parameter-Varying Model

The control-oriented design model of the powertrain has the following discrete-time state-space form:

$$\begin{bmatrix} SOC(k+1) \\ T_{ICE}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} SOC(k) \\ T_{ICE}(k) \end{bmatrix} + \begin{bmatrix} 0 & -K_b \\ \frac{T_s}{\tau_{etc}} & 0 \end{bmatrix} \begin{bmatrix} T_{ICE,c}(k) \\ T_{EM,c}(k) \end{bmatrix} \tag{15}$$

$$\begin{bmatrix} SOC_m(k) \\ \dot{m}_f(k) \end{bmatrix} = \begin{bmatrix} 100/SOC_{MAX} & 0 \\ 0 & K_f \end{bmatrix} \begin{bmatrix} SOC(k) \\ T_{ICE}(k) \end{bmatrix} \tag{16}$$

where the input vector $u(k) = [T_{ICE,c}(k), T_{EM,c}(k)]'$ represents the torque commands for the ICE and the EM, the state vector $x(k) = [SOC(k), T_{ICE}(k)]'$ comprises the battery State of Charge (SoC) and the ICE torque accounting for the electronic throttle dynamics, and the output $y(k) = [SOC_m(k), \dot{m}_f(k)]'$ involves the state of charge represented as a percentage and the fuel mass flow. The input-matrix involves a time-varying coefficient defined as:

$$K_b(k) = \frac{T_s \omega_{em}(k)}{V_{batt}(k) Q_{nom}} \tag{17}$$

where V_{batt} is the battery voltage, Q_{nom} is the battery nominal (maximum) charge, and T_s is the MPC sample time. The motor speed ω_{em} is derived from the measured vehicle speed and the known wheel radius as $\omega_{em} = v_{veh}/R_{wh}$. The LPV model was validated against the nonlinear simulation and generally a good fit was observed between the 'real' system outputs and their one-step-ahead predictions.

To use the look-up table $f_{fuel}(T_{ICE}, n_{eng})$, with n_{eng} the engine speed in rpm, as part of the LPV model, the mass flow characteristic is linearized around the current operating point $(T_{ICE,0}, n_{eng,0})$:

$$\dot{m}_f(k) = \dot{m}_{f,0} + \left.\frac{df_{fuel}}{dT_{ICE}}\right|_0 \left(T_{ICE}(k) - T_{ICE,0}\right) + \left.\frac{df_{fuel}}{dn_{eng}}\right|_0 \left(n_{eng}(k) - n_{eng,0}\right) \tag{18}$$

Defining the Jacobians $K_{ft,0} = \left.\frac{df_{fuel}}{dT_{ICE}}\right|_0$ and $K_{fn,0} = \left.\frac{df_{fuel}}{dn_{eng}}\right|_0$, the model can be rewritten as:

$$\dot{m}_f(k) = K_{ft,0}T_{ICE}(k) + \tilde{G}(k) \tag{19}$$

where

$$\tilde{G}(k) = \dot{m}_{f,0} + \left.\frac{df_{fuel}}{dT_{ICE}}\right|_0 (-T_{ICE,0}) + \left.\frac{df_{fuel}}{dn_{eng}}\right|_0 (n_{eng}(k) - n_{eng,0}) \tag{20}$$

can be treated as a time-varying disturbance. In the MPC problem formulation, this affine term can be incorporated into the fuel flow reference signal. The Jacobians K_{ft} and K_{fn} can therefore be approximated as finite differences along each direction, as:

$$K_{ft,0} = \frac{f_{fuel}(T_{ICE,0}+\Delta T_{ICE}, n_{eng,0}) - f_{fuel}(T_{ICE,0}-\Delta T_{ICE}, n_{eng,0})}{2\Delta T_{ICE}} \tag{21}$$

$$K_{fn,0} = \frac{f_{fuel}(T_{ICE,0}, n_{eng,0}+\Delta n_{eng}) - f_{fuel}(T_{ICE,0}, n_{eng,0}-\Delta n_{eng})}{2\Delta n_{eng}} \tag{22}$$

where ΔT_{ICE} and Δn_{eng} are arbitrarily small deviations of the engine torque and speed, respectively.

3.2 Hybrid Powertrain Constraints

The HEV powertrain involves various physical limits. These must be satisfied while computing the energy management strategy for enforcing the safety of both vehicle and driver. Such constraints may be listed as:

- In a HEV the battery is subject to a set of physical and control-design constraints that must be satisfied to ensure a nominal lifetime and good performance. For example, the SoC of the battery is usually constrained to vary over a prescribed range:

  $$100 \geq SOC_{MAX} > SOC_m(k) > SOC_{min} \geq 0 \tag{23}$$

  where $SOC_{MAX} = 80\%$ and $SOC_{min} = 40\%$ are the limits used in this study given in (The MathWorks, 2019).
- The current of the battery can be limited to ensure the performance of the battery is maintained. The battery current can be approximated by a linear function of the EM speed, neglecting electric power losses, and approximating the battery voltage by a nominal value. By writing the mechanical power generated by the electric system in terms of the vehicle speed, the battery current can be constrained as:

  $$I_{MAX} \geq I(k) = \frac{v_{veh}(k)N_R T_{EM}(k)}{R_{wh}V_{OC}} \geq I_{min} \tag{24}$$

  with $I_{MAX} = 150\,A$ and $I_{min} = -150\,A$ denoting the upper and lower limits.
- The power the battery can provide can be approximated by the instantaneous EM rotational speed, assuming a linear dependency on the EM torque, such that:

  $$P_{MAX} \geq P(k) = \frac{v_{veh}(k)N_R T_{EM}(k)}{R_{wh}} \geq P_{min} \tag{25}$$

  with $P_{MAX} = 46 \times 10^3 W$ and $P_{min} = -30 \times 10^3 W$ representing the maximum and minimum power of the battery, respectively.
- The torque provided by the EM should be constrained by the nominal maximum $T_{EM\,MAX} = 200$ Nm and minimum $T_{EM\,min} = -200$ Nm achievable torques, defined from the EM specifications, such that:

  $$T_{EM\,MAX} \geq T_{EM}(k) \geq T_{EM\,min}. \tag{26}$$

In a similar way, the torque provided by the ICE is constrained to be:

$$T_{ICE\,MAX} \geq T_{ICE}(k) \geq T_{ICE\,min}. \tag{27}$$

with $T_{ICE\,MAX} = 175$ Nm and $T_{ICE\,min} = 0$ Nm.

3.3 Linear Parameter-Varying Model Predictive Control

Given the discrete-time linear time-varying model of the hybrid powertrain in a state-space LPV form, and with a quadratic cost-function subject to affine constraints, the LPV-MPC optimization problem for the EMS can be expressed as:

$$\min_{\Delta u_j} \sum_{i=1}^{N_p} ||Q(y_{k+i|k} - r_{k+i|k})||_2^2 + \sum_{j=1}^{N_u} ||R(\Delta u_{k+j-1})||_2^2 + \sum_{j=1}^{N_u} ||W(u_{k+j-1})||_2^2 + \sum_{i=1}^{N_p} ||W_t(T_{k+i|k}^{ref} - T_{k+i|k})||_2^2 \tag{28}$$

$$\begin{aligned}
s.t.\quad & x_{k+i+1|k} = Ax_{k+i|k} + B_k u_{k+i} && (29)\\
& y_{k+i|k} = Cx_{k+i|k} && (30)\\
& T_{k+i} = Dx_{k+i|k} + Eu_{k+i} && (31)\\
& x_{k|k} = \hat{x}_k && (32)\\
& u_{k+i+1} = u_{k+i} + \Delta u_{k+i+1} \qquad for\, i \leq N_u && (33)\\
& u_{k+i+1} = u_{k+i} \qquad for\, i > N_u && (34)\\
& \Delta u_{k+i} \in \mathbb{D}\, , u_{k+i} \in \mathbb{U}\, , y_{k+i} \in \mathbb{Y}\, , N_p \geq N_u && (35)
\end{aligned}$$

where N_p denotes the prediction horizon, N_u is the control horizon, Q, R, W and W_t are the weights on output, input rate, input magnitude and torque reference tracking error respectively, $\Delta u_{k+j|k}$ is the vector of control increments, and $x_{k+i|k}$ is the prediction of the state at time $k+i$ based on information available at time k.

The vector $r_{k+i|k}$ represents the reference signal: $r_{k+i|k} = [SOC_{ref}(k+i|k) \quad \dot{m}_{f,ref}(k+i|k)]'$, and $SOC_{ref}(k) = SOC_{ref}(k+i|k)$ for $i = 0, .., N_p$ has the constant reference for the state of charge of the battery defined in the system set-up, and $\dot{m}_{f\,ref}(k+i|k) = 0$ is the fuel consumption reference level. The sets $\mathbb{D}$, $\mathbb{U}$, and $\mathbb{Y}$ are polyhedral sets of constraints of input rate, input, and output, respectively, defined according to the inequality constraints presented in Section 3.2. These constraints are defined given the hybrid powertrain constraints. According to the LPV modelling approach, the MPC design model in Eqs. (28)-(35) can be represented by the following matrices:

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};\; B_k = \begin{bmatrix} 0 & -K_b \\ \frac{T_s}{\tau_{etc}} & 0 \end{bmatrix};$$

$$C = \begin{bmatrix} 100/SOC_{MAX} & 0 \\ 0 & K_f \end{bmatrix};\; D = \begin{bmatrix} 0 \\ 1 \end{bmatrix}';\, E = \begin{bmatrix} 1 \\ 0 \end{bmatrix}' \tag{36}$$

Inserting the above system matrices to the MPC cost function (28), and performing the necessary algebraic manipulations, gives the standard form of the Quadratic Programming (QP) LPV-MPC optimization problem:

$$\min_z \frac{1}{2} z' H_k z + h_k' z \tag{37}$$

$$s.t.\quad G_k z \leq b_k \tag{38}$$

where the $z = [\Delta u_{k+1|k}, \ldots, \Delta u_{k+N_p|k}]'$ is the vector of optimization variables, H_k is the Hessian matrix, h_k is the linear term, and G_k and b_k are matrices defining the input and output constraints for the optimization problem defined from state-space matrices of Eq.(36) according to the common QP MPC formulation (Cavanini, L., Cimini, G., & Ippoliti, G, 2017).

3.4 QP Solver

The QP solver for solving the optimization problem of Eqs. (37)-(38) involves the Hildreth's QP Procedure (Wang Liuping, 2009). This solver computes the optimization sequence by considering the dual problem. The Hildreth's Procedure does not require matrix inversion and is based on an element-by-element search that was described in (Hossein Nia, S. Hassan, and Michael Lundh, 2016). The authors of that paper note that if the active constraints are linearly independent and their number is less than or equal to the number of decision variables, then the solution will converge, but if one, or both, of these requirements are violated, then the dual variables will not converge to a set of fixed values (in this case the iterations will terminate when the iterative counter reaches its maximum). The algorithm in this case gives a near-optimal solution that is not ideal but is useful. The main benefit of this approach is that since there is no matrix inversion, the computations will continue in such cases, which is important in real-time applications. This ability to recover when there is an ill-conditioned constrained problem is valuable.

3.5 NXP® GreenBox II and PIL Simulation Setup

The proposed LPV-MPC was designed and deployed on the NXP® GreenBox II Electrification Development Platform, in a PIL environment. The GB II board, shown in Figure 2, provides advanced performance, peripherals, and a multi-core Arm® ecosystem for engineers to begin development using NXP's next generation Automotive MCUs.

Fig. 2. NXP GreenBox II Electrification Development Platform

The second component of the system is the NXP Model Based Design Toolbox (MBDT) for MATLAB®. This is a comprehensive collection of tools that plug into the MATLAB® and Simulink® model-based design environment to support fast prototyping, verification, and validation on NXP microcontroller-based real-time targets. In the PIL simulations, the vehicle model was running in Simulink, while the MPC code was executed on the GBII board and provided the computed control actions back to the Simulink engine. In addition, the MBDT toolbox includes a means of profiling the algorithm execution by sending in relevant CPU timing and memory usage information. Note that the performance benchmarked was in principle the same as used in production Automotive ECUs.

4. RESULTS

In this section, the proposed MPC is compared with respect to two baseline controllers' results to assess fuel economy / battery degradation performance and PIL profiling results. The baseline controllers used were ECMS and A-ECMS (Musardo, C., Rizzoni, G., Guezennec, Y., & Staccia, B, 2005), the sampling time was $0.1\ s$, and the SoC reference value was $SOC_{ref} = 60\%$. The MPC calibration parameters are listed in Table 3.

Table 3. MPC Parameters

Symbol	*Parameter*	*Value*
N_p	Prediction Horizon	20
N_u	Control Horizon	10
W	Control action weight matrix	$[301, 100]$
R	Control action rate weight matrix	$[1, 1]$
Q	Output tracking error weight matrix	$[14 \times 10^5, 2]$
W_t	Torque tracking error weight matrix	1000

The FTP75 driving cycle was used to benchmark controller performance. The main performance criterion was the equivalent fuel economy index MPGe, which included the standard US EPA equivalent of 33.7 kWh/USgal. In addition, in this work, an emphasis was also placed on the battery stress, which was quantified using a battery damage index D_b, computed based on the rainflow counting algorithm (Shi, Y., Xu, B., Tan, Y., Kirschen, D., & Zhang, B., 2018). This index represents a fraction of the nominal lifetime of the battery 'consumed' during the driving cycle evaluated with respect to the SoC charging and discharging trajectory. There is normally a trade-off between the (short-term) energy optimization and (long-term) battery life. The performance of the controllers is shown in Figure 3 with ECMS (green), A-ECMS (blue), and MPC (red). The benchmark figures are collected in Table 4. Note that the MPGe index incorporates the actual fuel consumption and also accounts for the deviation of the final SOC from the nominal.

Table 4. Controllers' Performance Comparison

Controller	*Fuel* **[L/100km]**	*MPGe*	*SOC(t_{end})* **[%]**	D_b×10^5
ECMS	7.53	31.44	62.5	9.10
A-ECMS	7.43	31.78	61.0	5.31
MPC	7.17	32.85	60.2	0.86

The results shown in Figure 3 compare the MPC performance with respect to the baseline controllers. The battery SoC for the MPC exhibits much smaller excursions from its reference value of 60% than the ECMS and A-ECMS. The direct result of this is a significant reduction in the battery degradation index D_b. The MPC reduces this index by 91% and 83% with respect to ECMS and A-ECMS, designs, respectively. This would be reflected in increased Remaining Useful Lifetime (RUL) for the battery. It is interesting to note that the MPC achieves this result while generally providing similar or slightly improved performance in terms of MPGe (improvement of 3.4 % and 4.5% with respect to the A-ECMS and ECMS, respectively). This is thanks to performing the optimization over a horizon, rather than instantaneously, allowing the MPC to find a solution that exploits the battery's role as an energy buffer, while at the same time limiting the SOC variations to the relatively narrow region. Finally, the speed tracking error for all the controllers was found to be within $\pm 2\ m/s$.

The computational burden of the algorithms was also evaluated for the A-EMCS controller and two different MPC designs deployed in single precision: MPC3 (with minimum horizon lengths) and MPC20 (with nominal horizon lengths). The GBII PIL profiling results are reported in Table 5, which includes both the average and maximum CPU utilization figures. This performance information is valid for Arm CPU cores in the S32S247TV with Arm NEON® SIMD acceleration. Additionally, NXP provides vector acceleration technology that can speed up this performance by 200% or more at ½ the clock speed.

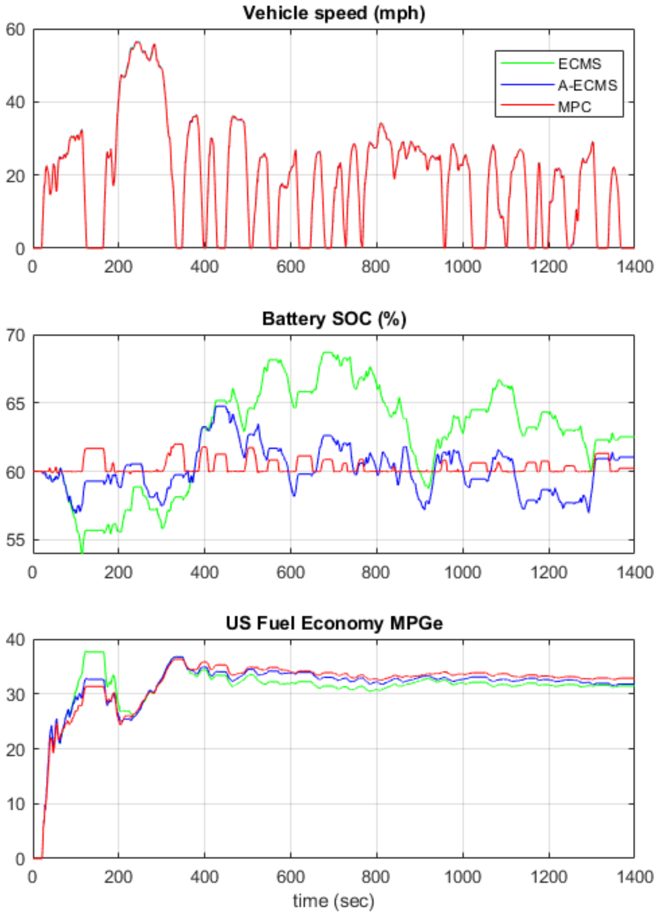

Fig. 3. Controller Comparisons for the FTP75 Driving Cycle. From top to bottom: the vehicle speed tracking performance (top); the SoC trajectory (middle); the MPGe fuel economy (bottom).

The results indicate the nominal MPC algorithm is on average 2.5x more demanding than the baseline A-ECMS, but it can still be executed on the GBII board in real-time. MPC designs with significantly longer horizons would benefit from further code optimization and/or MPC problem simplification. The maximum number of iterations for the QP solver may also need to be reduced, to limit the worst-case CPU loading.

Table 5. MPC Processor-In-the-Loop Profiling Results for the FTP75 driving cycle (sample time $T_s = 100$ms)

Controller	**CPU avg (%)**	**CPU max (%)**
A-ECMS	11.18	19.13
MPC3 (Np=3, Nu=2)	0.30	0.76
MPC20 (Np=20, Nu=10)	28.6	70.5

5. CONCLUSIONS

The performance results of a Model Predictive Controller applied to a hybrid electric vehicle energy management system have been presented. The MPC design was based on a Linear Parameter-Varying model and has been deployed using the NXP GreenBox II development platform. It was assessed on the FTP75 driving cycle using Processor-In-the-Loop simulation. Reported results show the capability of the proposed controller running on the GBII to improve baseline controller performance while guaranteeing feasible computational and memory requirements. Future research directions will consider the study of data-driven algorithms capable of exploiting external information data provided by vehicle sensors to improve the optimization performance of LPV-MPC by increasing the effectiveness of the predicted vehicle speed and torque signals. MPC code optimization will also be considered for more efficient QP problem formulation and solution.

Acknowledgements: We are grateful for the cooperation established with Mr. Brian McKay of the MathWorks.

REFERENCES

Camacho, Eduardo F., and Carlos Bordons Alba, 2013. *Model predictive control.* s.l.:Springer science & business media.

Cavanini, L., Cimini, G., & Ippoliti, G, 2017. *A fast model predictive control algorithm for linear parameter varying systems with right invertible input matrix.* s.l., Mediterranean Conference on Control and Automation.

Cavanini, L., Ippoliti, G., 2018. Fault tolerant model predictive control for an over-actuated vessel. *Ocean Engineering,* p. 19.

Cavanini, Luca, Gionata Cimini, and Gianluca Ippoliti, 2018. Computationally efficient model predictive control for a class of linear parameter-varying systems. *IET Control Theory & Applications,* pp. 1384-1392.

Grimble, M. J. and Majecki, P., 2020. *Nonlinear Industrial Control Systems: Optimal Polynomial Systems and State-Space Approach.* London: Springer-Verlag.

Horrein, L., Bouscayrol, A., Cheng, Y., & Dumand, C., 2015. *Hybrid energy management Strategy for hybrid electric vehicle.* s.l., s.n.

Hossein Nia, S. Hassan, and Michael Lundh, 2016. A general robust mpc design for the state-space model: Application to paper machine process. *Asian Journal of Control,* Volume 18`5, pp. 1891-1907.

Johnson, Valerie H., Keith B. Wipke, and David J. Rausen, 2000. HEV control strategy for real-time optimization of fuel economy and emissions. *SAE Technical Paper.*

Musardo, C., Rizzoni, G., Guezennec, Y., & Staccia, B, 2005. A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management. *European Journal of Control,* pp. 509-524.

Serrao, L., Onori, S., & Rizzoni, G., 2011. A comparative analysis of energy management strategies for hybrid electric vehicles.

Shi, Y., Xu, B., Tan, Y., Kirschen, D., & Zhang, B., 2018. Optimal battery control under cycle aging mechanisms in pay for performance settings. *IEEE Transactions on Automatic Control,* pp. 2324-2339.

The MathWorks, I., 2019. [Online]
Available at: Available: https://uk.mathworks.com/help/autoblks/ug/hev-p4-reference-application.html

Wang Liuping, 2009. *Model predictive control system design and implementation using MATLAB.* s.l.:Springer Science & Business Media.

Wu, J., Zou, Y., Zhang, X., Liu, T., Kong, Z., & He, D, 2019. An online correction predictive EMS for a hybrid electric tracked vehicle based on dynamic programming and reinforcement learning. *IEEE Access,* pp. 98252-98266.

Yan, F., Wang, J., & Huang, K., 2012. Hybrid electric vehicle model predictive control torque-split strategy incorporating engine transient characteristics. *IEEE transactions on vehicular technology,* pp. 2458-2467.