Energy Management of a Parallel Hybrid Electric Vehicle using Model Predictive Control

Kripanshu Yadav
(428311)
Dept. of Commercial Vehicle Technology
RPTU Kaiserslautern
67659,Kaiserslautern, Germany
mej31rac@rptu.de

Soham Paul
(428352)
Dept. of Commercial Vehicle Technology
RPTU Kaiserslautern
67659,Kaiserslautern, Germany
s.paul@edu.rptu.de

***Abstract*— This paper presents an in-depth analysis of energy management strategies for hybrid electric vehicles (HEVs) using Model Predictive Control (MPC). The study explores various MPC-based strategies, focusing on their optimization capabilities in real-time scenarios. The primary objective is to minimize fuel consumption while ensuring battery charge sustainment and respecting system constraints. Through simulations using Simulink, we optimize our strategy for the given drive cycles. The results highlight the trade-offs between computational complexity and energy efficiency, emphasizing the practical applicability of MPC in HEVs.**

Keywords—Energy Management Strategy (EMS), Model Predictive Control, hybrid electric vehicle, Prescient MPC, Simulink, MPC Toolbox, Optimization Toolbox,

ABBREVIATIONS

ACC	Adaptive Cruise Control
DP	Dynamic Programming
ECMS	Equivalent Consumption Minimization Strategy
EM	Electric Motor
EMPC	Explicit Model Predictive Control
EMS	Energy Management Strategy
HEV	Hybrid Electric Vehicles
HFET	Highway Fuel Economy Test
HIL	Hardware-in-loop
ICE	Internal Combustion Engine
LTI	Linear Time-Invariant
MC	Markov Chain
MPC	Model Predictive Control
MV	Manipulated Variable
OV	Output Variable
PHEV	Parallel Hybrid Electric Vehicle
PR	Pattern recognition
QP	Quadratic Programming
RMSE	Root Mean Square Error
SMPC	Stochastic Model Predictive Control
SOC	State of Charge
SVM	Support Vector Machine
SVPP	Synthesis Velocity Profile Prediction

I. INTRODUCTION

A. Motivation

Greenhouse gas emissions from the mobility sector have increased in the last 10 years, whereas emissions from the other sectors have been reduced and 15% of total CO2 emissions in European Union are generated by passenger vehicles [1]. In addition, the European Commission is also applying more stringent CO2 emission regulations to automakers [1]. Moreover, source [2] mentions that most of the vehicles available in the market are powered by internal combustion engines (ICE), which use petrol/diesel as a fuel, are not efficient. Source [1] mentions adapting to other powertrain vehicles, like hybrid electric vehicles (HEVs), can reduce overall emissions and fulfil the emissions requirements, because sources [2], [3] and [4] outline that HEVs can improve the fuel efficiency and reduce the tail-pipe emissions.

HEVs usually consist of ICE and one or more electric motors as energy converters, which take energy from fuel and power storage devices like batteries, to propel the vehicle. According to [3], energy management strategy (EMS) can be defined as a procedure to divide or distribute the power between the energy devices in HEVs and [1] mentions that EMS can affect the overall fuel consumption. Moreover, source [5] also proves that EMS is one of the important and widely researched domains for HEVs.

Currently, EMS can be roughly divided into rule-based control strategies and optimization-based control strategies. Rule-based strategies are real-time logical strategies developed based on some static conditions/rules. Optimization-based control strategies are classified further into Global optimization and Local optimization [5]. Although global optimization cannot be implemented in real-time, but it can be used as a benchmark to compare the results of other control strategies because it optimizes the complete drive cycle [2], [6]. For real-time implementation Local optimization can be used, as it provides optimum solutions locally or for a short period of time [6]. Several online optimization techniques like Equivalent Consumption Minimization Strategy (ECMS), Intelligent Control and

Model Predictive Control (MPC) are mentioned in the [2]. MPC works on the principle of optimizing the objective function in the prediction horizon at every time step [5]. It is one of the appropriate control strategies for HEVs because source [7] commented that MPC based control strategies can provide similar results compared to Dynamic Programming and can be implemented in real time, source [2] mentioned that MPC can handle multiple inputs, outputs and state constraints, and source [4] pointed out that MPC controller can make the switching between different driving modes in a smoother way.

In the past few years many different MPC control strategies based on different prediction models and optimization techniques have been proposed for HEVs.

Sources [2], [8] and [9] outlined that velocity's prediction accuracy is one of the crucial parameters which can affect the optimization problem and can result in improving the overall fuel efficiency.

B. Literature Review

Lei et al. [8] presented an exponentially varying MPC using traffic information to improve fuel efficiency compared to Dynamic Programming (DP). They found that a static decay rate did not yield the desired velocity prediction accuracy; therefore, they used a Support Vector Machine (SVM) with a Radial Basis Function kernel to dynamically compute decay coefficients. VISSIM 'Verkehr In Städten – SIMulationsmodell' was employed to simulate traffic scenarios, obtain State of Charge (SOC) references, and track driving cycles, which enhanced battery operating efficiency and fuel efficiency. The study compared MPC with traffic information (MPC-traffic), MPC with known trip duration (MPC-duration), and DP. MPC-traffic consumed 7% more energy and was 7.5% more expensive than DP, whereas MPC-duration consumed 13% more energy and was 13.5% more expensive than DP. However, the study did not address computational complexity or verified simulations on embedded hardware using HiL or ViL tests. [8]

Ruan et al. [11] improved controller efficiency by integrating Adaptive Cruise Control (ACC) with explicit MPC (EMPC). They developed two controllers: one for ACC using a longitudinal dynamics model and another for Energy Management System (EMS) using a powertrain model, cascading both explicit MPCs. Optimal control inputs were computed offline with a multi-parametric quadratic programming algorithm and stored for online use via a lookup table. Their method improved fuel efficiency and driving comfort, showing an 11% improvement in city driving (UDDS cycle) and a 2.9% improvement on Highway Fuel Economy Test Cycle (HWFET) with ACC compared to without ACC. EMPC consumed less energy than ECMS, using more energy than DP in city driving and 0.7% more on highways, while maintaining similar charge sustainment. The optimal computation time for EMPC was analyzed with respect to prediction horizon, where prediction horizon of 5 is optimal and further increasing the prediction horizon, increases the computation time drastically. The approach was validated with hardware-in-loop (HIL) tests, demonstrating real-time applicability. [11]

Source [10] used stochastic Model Predictive Control (SMPC) with driving Pattern Recognition (PR) to enhance prediction accuracy and reduce fuel consumption. Velocity prediction was based on a Markov chain (MC), adjusted by driving classification determined through an EM algorithm. This approach improved prediction accuracy by nearly 40% in terms of Root Mean Square Error (RMSE) compared to a Markov chain without PR. Results showed that MPC with PR had slightly better charge sustainment than MPC without PR and rule-based strategies during the considered drive cycle. During rapid velocity changes, the rule-based strategy consumed 8.1% more fuel, and MPC without PR consumed around 5% more fuel compared to MPC with PR. The study also found that computational time increased with longer prediction horizons, but it did not address real-time implementation or economic aspects. [10]

Biswas et al. [12] introduced a non-linear MPC using Model Predictive Static Programming (MPSP) to solve a multi-objective optimization problem, which transformed dynamic optimization into static optimization for approximating control inputs. Their non-linear model accounted for factors like temperature, typically ignored in linear models. Weights were tuned using a non-dominant multi-objective approach to reduce space complexity. MPSP-MPC showed 26% better prediction accuracy than Sequential Quadratic Programming MPC (SQP-MPC) and 59% better than Linear Time-Varying MPC (LTV-MPC) in terms of RMSE. Under the Urban Dynamometer Driving Schedule (UDDS) drive cycle, MPSP-MPC consumed 6.5% more fuel and 5.5% more battery energy than DP, while SQP-MPC consumed 7.5% more fuel and 6.4% more battery energy. In the NEDC drive cycle, DP was slightly more fuel-efficient and energy-efficient than both MPSP-MPC and SQP-MPC. Computational time increased with longer prediction horizons, making it a crucial factor alongside fuel consumption for real-time implementation. [12]

Zhang et al. [7] implemented a SMPC by integrating MC and participatory sensing data (PSD) to predict the velocity profile and reference SOC trajectory in a hierarchical approach. The first layer combined MC and PSD with linear regression for Synthesis Velocity Profile Prediction (SVPP). The second layer computed terminal SOC constraints, and the final layer solved the optimization problem using DP. Under the US06 drive cycle, SVPP improved prediction accuracy by 60.4%, 35%, and 15% compared to MPC with PSD, MC, and ANN, respectively. Similarly, in a combined cycle with real vehicle data, SVPP improved accuracy by 60%, 36%, and 17%. MPC with SVPP consumed 4% more equivalent fuel than DP, while MPC with ANN, MC, and PSD consumed 5%, 7.5%, and 8.5% more, respectively, with similar results for the combined cycle. MPC with SVPP had computational times 23% higher than MPC with ANN, 59.7% lower than MPC with MC, and similar to MPC with PSD. Real-time implementation was not verified. [7]

C. Contribution

The objective of the project is to:

- minimize the fuel consumption,
- ensuring charge sustainment (charge of the battery must remain the same on the average)
- Respecting the component constraints (maximum torques, maximum currents, etc.).

With the aforementioned strategies in mind, we chose MPC as our research strategy to fulfil these objectives. Although many novel and complex algorithms have been

proposed, keeping time constraints, we selected to utilize characteristics of Frozen-Time MPC and Prescient MPC.

A rule-based strategy was provided as the Control Unit template and the challenge was to only change the control unit block in Simulink, which meant no additional signals could be utilized, furthermore other subsystem cannot be altered.

D. Outline

In Section II, fundamentals and methodology will be introduced, which includes prediction model, cost function and constraints. Results will be discussed in Section III. The report will end with the conclusion and future work in Section IV.

II. Fundamentals and Methodology

A. Methodology

A parallel HEV (PHEV) architecture is considered in this project. A quasi-static vehicle model is used, where the control unit receives data regarding the required velocity from the drive cycle. The requirement of the control unit must optimally distribute this load between the electric motor and the combustion engine, considering the system constraints and fulfilling the torque requirements.

In a PHEV, the torque coupler is used to split the power from the gearbox between combustion engine and electric motor. Therefore, the relation between the angular velocities can be defined by

$$\omega_{MGB} = \omega_{EM} = \omega_{CE} \quad (1)$$

where ω_{MGB} is angular velocity of gearbox, ω_{EM} is angular velocity of electric motor, ω_{CE} is angular velocity of combustion engine. The power distribution can be defined by eq. 2 [13, eq.1.1]

$$P_{MGB} - P_{CE,mech} - P_{EM,mech} = 0 \quad (2)$$

where P_{MGB} is power from manual gearbox, $P_{CE,mech}$ is mechanical power from combustion engine and $P_{EM,mech}$ is mechanical power from electric motor. Furthermore, the battery model is expressed as

$$\dot{Q_{BT}}(k) = \frac{P_{EM,mech}(k)}{\eta_{EM} U_{nominal}} \quad (3)$$

where Q_{BT} is battery charge, η_{EM} is efficiency of motor and $U_{nominal}$ is battery nominal voltage. Mass flow rate of fuel can be calculated by

$$V_{CE}(k) = \frac{P_{CE,mech}(k)}{\eta_{CE} H_u} \quad (4)$$

where V_{CE} is $fuel\ rate$, η_{CE} is efficiency of engine and H_u is enthalpy of fuel.

B. *Proposed Energy Management Strategy*

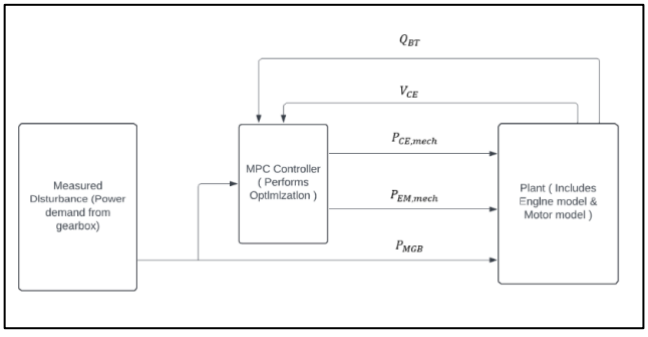

Fig.1. Proposed Energy Management Strategy

Fig.1 shows the flow of signals in the proposed system. The drive-cycle provides velocity data to the vehicle block, which contains the vehicle attributes, thus giving the wheel torque and wheel speed. This data is sent to the manual gearbox which has the gear ratios, the output from this block is the gearbox's angular speed, angular acceleration, and torque. This signal is input to the control unit, which contains the MPC controller. According to the load and speed, the MPC controller adjusts the torque-split ratio between the combustion engine and the electric motor.

In addition to control of the torque-split, our model also incorporates future disturbance values from the drive cycle, which are sent to the MPC controller. These disturbances may represent factors such as upcoming changes in velocity that could affect vehicle performance. Incorporating future disturbances enhances the predictive capabilities of the Frozen-Time MPC (FTMPC), allowing it to better anticipate and react to changes in driving conditions. While Prescient MPC (PMPC) is more complex and computationally intensive, it provides higher performance by considering future disturbances more accurately. Our approach offers a compromise between the simplicity and computational efficiency of FTMPC and the enhanced performance of PMPC.

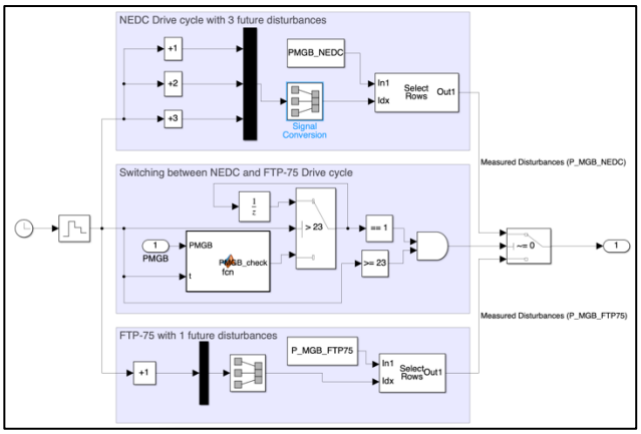

Fig.2. Switching drive cycle for future disturbances

To provide the MPC with accurate information about the upcoming disturbances from the drive cycle requires storing the drive cycles—NEDC and FTP-75—in the memory of the controller as shown in fig. 2. The controller must then identify which drive cycle is being followed to provide the correct disturbance values at the appropriate time. To achieve this, the control unit includes a block that identifies the current drive cycle based on the characteristics of the initial measured disturbances.

For example, it has been observed that the FTP-75 drive cycle has its first non-zero disturbance value at t = 23 seconds, while the NEDC remains at zero until t = 52 seconds. This distinction allows the control unit to switch between different drive cycles by recognizing when the first non-zero disturbance occurs. By leveraging this knowledge, a switching function is implemented, which ensures that the controller provides the correct future disturbance values to the MPC. This method improves the accuracy of the predictions and allows the system to adapt more effectively to different drive cycles.

The overall energy management strategy for the parallel HEV is defined by three main components: the prediction model, the cost function, and the constraints. The prediction model is used to forecast the future states of the system based on the current inputs and disturbances, while the cost function minimizes fuel consumption and ensures that the battery charge remains within acceptable limits. The constraints ensure that the system operates within the physical limitations of the vehicle's components, such as maximum torque or power limits. Together, these elements work in harmony to optimize the vehicle's energy usage, ensuring efficient operation across a range of driving conditions.

i. Prediction Model

A linear time invariant discrete state space model is used to predict the future states which can be described by eq. 5 [14, p.1-12]

$$\begin{Bmatrix} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{Bmatrix} \quad (5)$$

where, x^T = [Q_{BT} , V_{CE}] is the state vector, u^T = [$P_{EM,mech}$, $P_{CE,mech}$, P_{MGB}] where $P_{EM,mech}$ and $P_{CE,mech}$ denote the input vectors and P_{MGB} is the measured input disturbance and $y^T = [Q_{BT}, V_{CE}]$ is the output vector.

Using the previously established equations (3), (4), and (5), the various matrices that define the system's dynamics, namely, the system matrix A, the input matrix B, the output matrix C, and the feedthrough matrix D can be explicitly written as $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $B = \begin{bmatrix} \frac{-T_s}{\eta_{EM} U_{nominal}} & 0 & 0 \\ 0 & \frac{T_s}{\eta_{CE} H_u} & 0 \end{bmatrix}$, $C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

The input matrix B, output matrix C, and feedthrough matrix D are key to controlling and predicting the behaviour of the PHEV. The input matrix B defines how the control inputs—such as mechanical power from the electric motor and combustion engine—affect the state variables. For instance, the efficiency factors η_{EM} and η_{CE} directly influence how much energy is transferred into the system, highlighting the importance of these efficiencies in determining the overall performance of the vehicle.

Similarly, the output matrix C maps the system's internal states to the measurable outputs, ensuring that the control unit can monitor the key indicators of performance, such as the battery charge and engine fuel rate. By using this matrix, the control system ensures that the vehicle operates within its designed parameters while optimizing energy usage. The feedthrough matrix D, in this case, is zero, implying that the current inputs do not directly affect the outputs in the same time step. This simplifies the control design by focusing the controller's attention on future states and input-output relationships rather than immediate disturbances.

In our case, we have assumed the η_{EM} as 0.85, η_{CE} as 0.45 and $U_{nominal}$ as 47 Volts to simplify and linearize the prediction model as LTI State Space system. Sample time T_s is 1 second in our case which is same as simulation time-step of model.

Output vector is given by $y^T = [Q_{BT}, V_{CE}]$, meaning that the control unit is provided with both the battery charge Q_{BT} and the engine fuel rate V_{CE} as feedback signals. Due to the absence of V_{CE} signal from the plant, a method to calculate V_{CE} is required for accurate control and prediction of system behaviour.

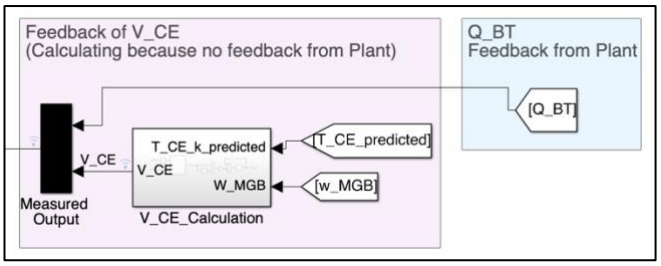

Fig.3. V_{CE} calculation

This is done by estimating the requested torque from the combustion engine for the next step and using the current angular velocity as shown in fig. 3. The requested torque from the combustion engine for the next step is directly taken from the output of the MPC, which is the first value of the MV sequence. These values are used to calculate the V_{CE} from the engine fuel consumption map. Although this estimation method introduces some degree of inaccuracy, it provides a practical means to calculate V_{CE} without requiring additional sensors or signals. By relying on the predicted torque from the MPC and the known engine characteristics, the control unit can infer the fuel rate for the next step and proceed with the control actions accordingly. This approach ensures the system remains functional and predictive, even in the absence of complete sensor data for an output variable.

ii. Cost Function

In Simulink, MPC block uses a standard linear cost function, given by eq.6 [14, p.1-7]

$$J(z_k) = J_y(z_k) + J_u(z_k) + J_{\Delta u}(z_k) + J_\varepsilon(z_k) \quad (6)$$

where $J_y(Z_k)$ is output reference tracking, $J_u(Z_k)$ is manipulated variable tracking, $J_{\Delta u}(z_k)$ is manipulated variable move suppression and $J_\varepsilon(z_k)$ is constraint violation. The output reference tracking is defined as eq.7 [14, p.1-7]

$$J_y(z_k) = \sum_{j=1}^{n_y} \sum_{i=1}^{p} \left\{ \frac{w_{i,j}^y}{s_j^y} \left[r_j(k+i|k) - y_j(k+i|k) \right] \right\}^2 \quad (7)$$

where k is current control interval, p is prediction horizon, n_y is number of plant output variables and Z_k is Quadratic Programming (QP) decision variables vector, given by $z_k^T = [\, u(k|k)^T \; u(k+1|k)^T \ldots u(k+p-1|k)^T \; \varepsilon_k]$, $y_j(k+i|k)$ is Predicted value of the j^{th} plant output at the i^{th} prediction horizon step, $r_j(k+i|k)$ is Reference value for the j^{th} plant output at the i^{th} prediction horizon step, s_j^y is Scale factor for the j^{th} plant output, $w_{i,j}^y$ is Tuning weight for the j^{th} plant output at the i^{th} prediction horizon step.

Manipulated variable tracking is defined as eq.8 [14, p.1-8]

$$J_u(z_k) = \sum_{j=1}^{n_u} \sum_{i=0}^{p-1} \left\{ \frac{w_{i,j}^u}{s_j^u} \left[u_j(k+i|k) - u_{j,target}(k+i|k) \right] \right\}^2 \quad (8)$$

which is like that of output reference tracking, but with adjustments to the parameters, specifically for the input

variables, the details can be found in [14, p. 1-8].

In this project, charge sustainment, while minimizing the fuel flow was a priority, thus a value of 18000 was chosen for the Q_{BT} reference, and a minimal value of 0.01 for V_{CE} reference. The control input doesn't follow a target or reference value in this project, therefore $u_{j,target}$ is 0.

Suppression of sudden changes in the torque-split ratio is not considered in this project as it deviates from the optimal ratio, thus $J_{\Delta u}(z_k)$ is neglected by having the tuning weights for this term as 0. Additionally, soft constraints allow violation of the limits, for example the MPC controller might send a value exceeding the motor power, this triggers the overload systems in the vehicle model and stops the simulation. Thus, constraint violation is not allowed by keeping the tuning weights for the term $J_{\varepsilon}(z_k)$ as 0.

iii. Constraints

In a PHEV, the angular velocities of the Combustion engine, Electric motor and the gearbox are correlated, therefore the torque limits of each power source change as per the vehicle velocity. Thus, we update the constraints of the variables at each control interval as per the current state, to maximize the real-time limits.

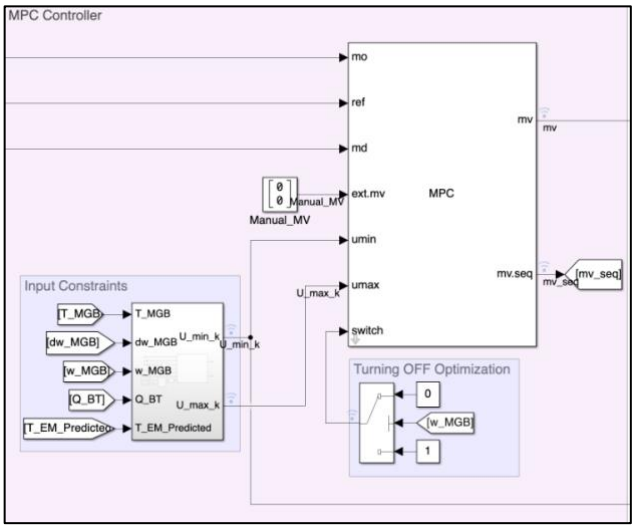

Fig. 4. Input Constraints connected with MPC controller

In Fig. 4 the Input Constraints block provides the MPC with limits ($U_{min}$and U_{max}) of the MV. This block takes ω_{MGB}, T_{MGB}, $d\omega_{MGB}$, Q_{BT} and predicted torque for electric motor at each time step. The limits for the combustion engine power are straightforward and directly dependent on the ω_{MGB}, and is found from the maximum torque map. Whereas the limits for the electric motor depend on torque and current. The torque limits are dependent on $\omega_{MGB}$and found from the max torque map, and the current limits are calculated from the current Q_{BT}.

The Simulink model of PHEV consists of various subsystems, which have their own limits, which need to be respected. Thus, we require constraints on the MV and OV, which are given as follows:

$$\begin{Bmatrix} P_{EM(min)} < u_1 < P_{EM(max)} \\ P_{CE(min)} < u_2 < P_{CE(max)} \\ Q_{BT(min)} < y_1 < Q_{BT(max)} \\ V_{CE(min)} < y_2 < V_{CE(max)} \end{Bmatrix} \quad (9)$$

Explicit Constraints for MV $[P_{EM};P_{CE}]$ and OV $[Q_{BT};V_{CE}]$ used in linear MPC block from MPC toolbox in Simulink-MATLAB can defined by eq. 10 [14, p.1-11]

$$\begin{Bmatrix} \frac{y_{j,min}(i)}{s_j^y} - \varepsilon_k V_{j,min}^y(i) \le \frac{y_j(k+i|k)}{s_j^y} \le \frac{y_{j,max}(i)}{s_j^y} + \\ \varepsilon_k V_{j,max}^y(i), \quad i = 1{:}p,\ j = 1{:}n_y \\ \frac{u_{j,min}(i)}{s_j^u} - \varepsilon_k V_{j,min}^u(i) \le \frac{u_j(k+i-1|k)}{s_j^u} \le \frac{u_{j,max}(i)}{s_j^u} \\ +\varepsilon_k V_{j,max}^u(i), \quad i = 1{:}p,\ j = 1{:}n_u \\ \frac{\Delta u_{j,min}(i)}{s_j^u} - \varepsilon_k V_{j,min}^{\Delta u}(i) \le \frac{\Delta u_j(k+i-1|k)}{s_j^u} \le \\ \frac{\Delta u_{j,max}(i)}{s_j^u} + \varepsilon_k V_{j,max}^{\Delta u}(i), \quad i = 1{:}p,\ j = 1{:}n_u \end{Bmatrix} \quad (10)$$

V parameters are dimensionless controller constants analogous to the cost function weights and ε_k (slack variable) parameters are considered 0, to define it as Hard Constraints on MV and OV. $s_j^y$and s_j^u are scaling factor for inputs and outputs, which are taken as 1 for our case.

In Simulink, we implemented (2) using a mixed input-output constraints defined by eq.11 [14, p.1-11]

$$Eu(k+j) + Fy(k+j) + Sv(k+j) \le G + \varepsilon V \quad (11)$$

where E =$\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}$, F =$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$, S =$\begin{bmatrix} 1 \\ -1 \end{bmatrix}$, G=$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, V= $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and $v(k+j)$ is a column vector of measured disturbance variable.

III. Result and Discussion

The MPC controller has many variables which can be tuned and changed to suit the requirements, such as the prediction horizon, control horizon etc. Figures 5 and 6 show the change in the fuel consumption in both NEDC and FTP-75 drive cycle with respect to the prediction horizon and control horizon and choose the values which correspond to the lowest fuel consumption. Therefore, prediction horizon is selected 9 for FTP-75 and 11 for NEDC, while the control horizon is 1 for FTP-75 and 2 for NEDC.

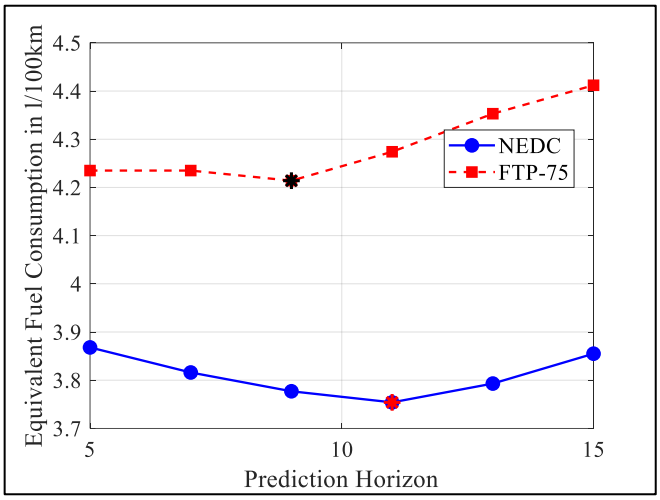

Fig. 5. Eqv. fuel consumption vs prediction horizon

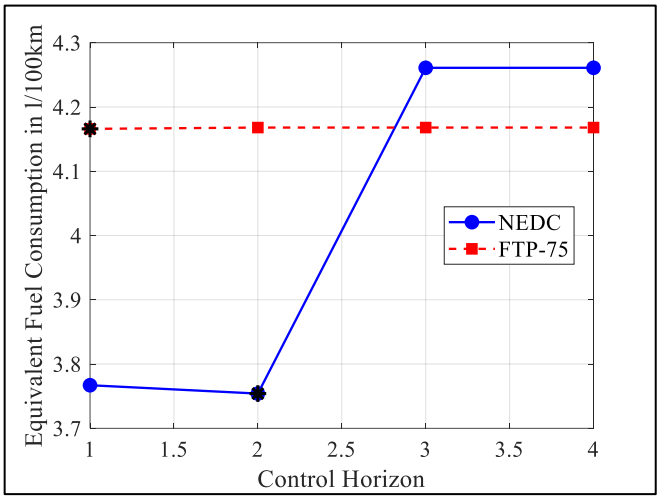

Fig. 6. Eqv. fuel consumption vs control horizon

Moreover, to improve the performance of the controller the future measured disturbance is provided, fig. 7 shows the number of future steps of the measured disturbances, which is optimal at 1 step for FTP-75, and 3 for NEDC.

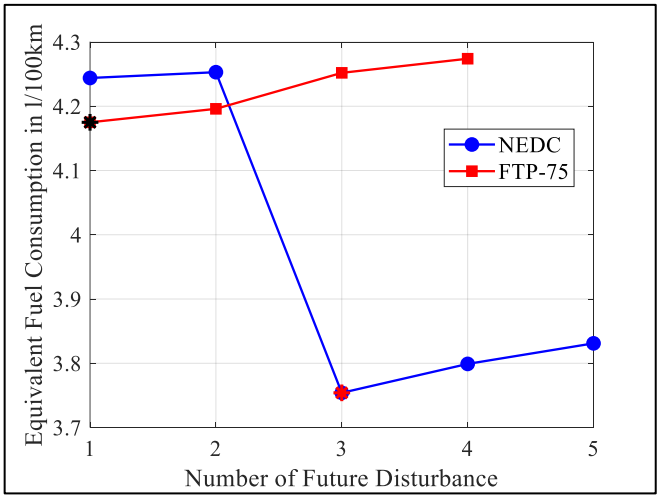

Fig. 7. Eqv. fuel consumption vs number of future disturbances

Figures 8 and 9 show the change in the value of tuning weights of the cost function, which penalizes the MV (P_{EM} and P_{CE}) relative to each other. Each graph is constructed by taking the other value constant, i.e. the tuning weights of P_{EM} is changed by keeping the tuning weights for P_{CE} constant and vice versa. It can be observed from fig. 8 that with increase in the weights for P_{EM}, the controller penalizes the usage of electric motor, similarly increasing the tuning weights for P_{CE} penalizes the usage of combustion engine. This is evident from the graphs for NEDC drive cycle, as fuel consumption decreases with the increased usage of electric motor i.e. with a lower tuning weight for P_{EM} and higher tuning weight for P_{CE}.

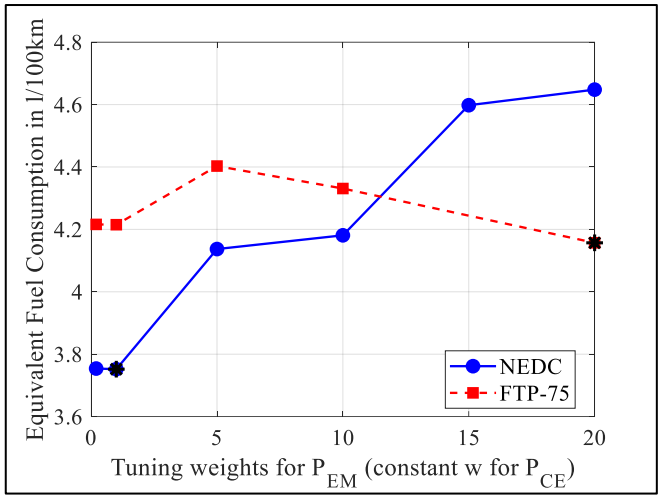

Fig. 8. Eqv. fuel consumption vs varying tuning weights for P_EM

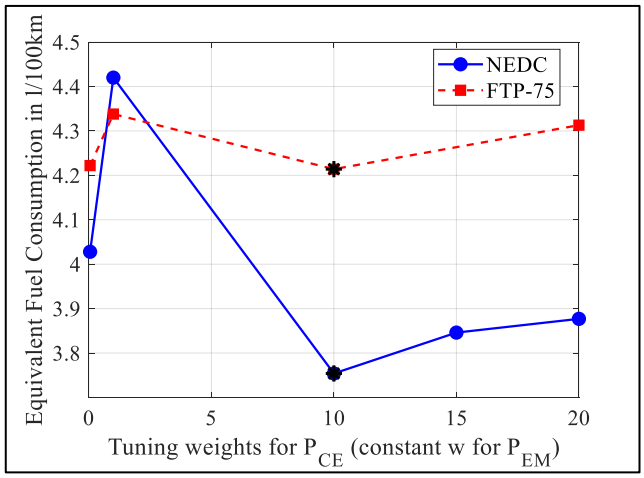

Fig. 9. Eqv. fuel consumption vs varying tuning weights for P_CE

A hybrid vehicle utilizes the power sources in different combinations. These operation modes are shown in figures 10. When the vehicle speed is comparatively low and constant, the MPC controller requests power from the electric motor. As the vehicle decelerates, the electric motor functions as a generator and energy is harvested. The figures also show an unexpected trend, which is the utilization of combustion engine rather than electric motor during acceleration, which is normally not expected. Furthermore, in the time segment 1030s to 1065s, load point shifting takes place, showing a curve like a sine wave. Although the velocity is constant in this segment, constant power from electric motor was expected. This discrepancy is because of higher velocity and a large error from target Q_{BT} value at around 1030s and a decreasing Q_{BT} , so the MPC demands power from combustion engine to charge the battery to bring it closer to the target value.

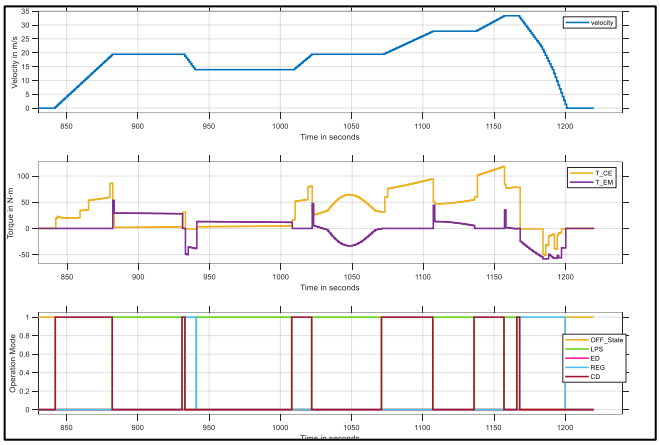

Fig. 10. Velocity, Torque and Operation Mode for NEDC

The operation modes in fig.8 also shows the controller switching to conventional driving due to velocity increasing to the maximum at the end of the NEDC cycle and a decreasing Q_{BT}. At the end of the NEDC drive cycle, a large deceleration event can be seen, here we would like to harvest the energy through the generator, but the power capacity of the motor limits it and the rest is lost to engine braking in the model according to eq.2.

The operation modes show almost no electric drive mode, and more load point shifting. When scrutinized further, it can be observed that the torque-split ratio is around 0.9, which means 90% of the power is supplied by the electric motor. Unlike rule-based strategies, where the mode can be easily controlled, MPC strategies can be influenced by the parameters mentioned before but cannot be coerced to be pure electric drive during a particular segment. Thus, a lot of segments marked as Load Point Shift mode approach Electric Drive mode.

IV. Conclusion and Future Work

In this study, we proposed an MPC strategy for energy management in a parallel HEV. By balancing the trade-off between computational complexity and performance, our approach, which combines features of Frozen-Time and Prescient MPC, successfully minimized fuel consumption and maintained charge sustainment without violating the constraints of the vehicle model. The performance of the MPC was influenced by various factors, including the prediction horizon, control horizon, tuning weights, and the number of future disturbances. Through simulations on the NEDC and FTP-75 drive cycle, the MPC demonstrated the ability to switch smoothly between driving modes, optimizing the power distribution between the electric motor and combustion engine while adhering to system constraints.

In the current prediction model, the state space equation is fixed; thus, the efficiencies of the combustion engine, electric motor and the nominal voltage of the battery are taken constant throughout the simulation. Future improvements to the proposed MPC strategy could focus on incorporating Adaptive MPC, where system parameters such as the efficiency of the electric motor and combustion engine are updated dynamically based on real-time data. This would allow the prediction model to more accurately reflect changes in vehicle behavior and environmental conditions, resulting in better performance and more precise control. For example, the efficiency of the combustion engine might change depending on the temperature, fuel type, or load conditions, and an adaptive controller could account for these variations in real time.

Another area of potential improvement is the inclusion of stochastic predictive models, which could handle uncertainties in future disturbances or driving conditions. This would be particularly useful in scenarios where the driving cycle is not known in advance or where external factors like traffic or road conditions could significantly affect vehicle performance. By using stochastic models, the MPC could better anticipate and respond to unpredictable changes, improving the robustness and overall efficiency of the control strategy.

References

[1] M. Josevski and D. Abel, "Energy management of parallel hybrid electric vehicles based on stochastic model predictive control," IFAC Proceedings Volumes, vol. 47, no. 3, pp. 2132-2137, Jan. 2014. doi: 10.3182/20140824-6-ZA-1003.02058..

[2] Y. Huang, H. Wang, A. Khajepour, H. He, and J. Ji, "Model predictive control power management strategies for HEVs: A review," Journal of Power Sources, vol. 341, pp. 91-106, Feb. 2017. doi: 10.1016/j.jpowsour.2016.11.092.

[3] B. Zhang and T. Shen, "An optimal energy management strategy for parallel HEVs," in 2019 22nd International Conference on Electrical Machines and Systems (ICEMS), Aug. 2019, pp. 1-5. doi: 10.1109/ICEMS.2019.8922345.

[4] A. Chandrasekar, S. Sengupta, S. Hingane, C. Gururaja, and S. Pandit, "Comparative analysis of model predictive control (MPC) and conventional control in supervisory controller of a retrofit HEV," SAE Technical Paper, Jan. 2017. doi: 10.4271/2017-26-0104.

[5] Q. Xue, X. Zhang, T. Teng, J. Zhang, Z. Feng, and Q. Lv, "A comprehensive review on classification, energy management strategy, and control algorithm for hybrid electric vehicles," Energies, vol. 13, no. 20, p. 5355, Oct. 2020. doi: 10.3390/en13205355..

[6] Z. Chen, H. Gu, S. Shen, and J. Shen, "Energy management strategy for power-split plug-in hybrid electric vehicle based on MPC and double Q-learning," Energy, vol. 245, p. 123182, Apr. 2022. doi: 10.1016/j.energy.2021.123182.

[7] Y. Zhang, L. Chu, Y. Ding, N. Xu, C. Guo, Z. Fu, L. Xu, X. Tang, and Y. Liu, "A hierarchical energy management strategy based on model predictive control for plug-in hybrid electric vehicles," IEEE Access, vol. 7, pp. 81612-81629, Jun. 2019. doi: 10.1109/ACCESS.2019.2923271.

[8] Z. Lei, D. Sun, J. Liu, D. Chen, Y. Liu, and Z. Chen, "Trip-oriented model predictive energy management strategy for plug-in hybrid electric vehicles," IEEE Access, vol. 7, pp. 113771-113785, Aug. 2019. doi: 10.1109/ACCESS.2019.2935084.

[9] L. Guo, X. Zhang, Y. Zou, N. Guo, J. Li, and G. Du, "Cost-optimal energy management strategy for plug-in hybrid electric vehicles with variable horizon speed prediction and adaptive state-of-charge reference," Energy, vol. 232, p. 120993, Oct. 2021. doi: 10.1016/j.energy.2021.120993.

[10] S. Ruan, Y. Ma, N. Yang, C. Xiang, and X. Li, "Real-time energy-saving control for HEVs in car-following scenario with a double explicit MPC approach," Energy, vol. 247, p. 123265, May 2022. doi: 10.1016/j.energy.2022.123265.

[11] J. Hao, S. Ruan, and W. Wang, "Model predictive control based energy management strategy of series hybrid electric vehicles considering driving pattern recognition," Electronics, vol. 12, no. 6, p. 1418, Mar. 2023. doi: 10.3390/electronics12061418.

[12] D. Biswas, S. Ghosh, S. Sengupta, and S. Mukhopadhyay, "Energy management of a parallel hybrid electric vehicle using model predictive static programming," Energy, vol. 250, p. 123505, Jul. 2022. doi: 10.1016/j.energy.2022.123505.

[13] D. Görges, Lecture notes, Electric and Hybrid Vehicles: Introduction to Electric and Hybrid Vehicles. Kaiserslautern, Germany, SS2024. [Online]. Available: https://olat.vcrp.de/url/RepositoryEntry/4531979893

[14] MathWorks, Inc., "Model Predictive Control Toolbox User's Guide," Natick, MA, USA, 2021. [Online]. Available: https://www.mathworks.com/help/m