20XX-01-XXXX

Powertrain Modeling and Model Predictive Longitudinal Dynamics Control for Hybrid Electric Vehicles

Author, co-author (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Affiliation (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Abstract

This paper discusses modeling of a power-split hybrid electric vehicle and the design of a longitudinal dynamics controller for the University of Waterloo's self-driving vehicle project. The powertrain of Waterloo's vehicle platform, a Lincoln MKZ Hybrid, is controlled only by accelerator pedal actuation. The vehicle's power management strategy cannot be altered, so a novel approach to grey-box modeling of the OEM powertrain control architecture and dynamics was developed. The model uses a system of multiple neural networks to mimic the response of the vehicle's torque control module and estimate the distribution of torque between the powertrain's internal combustion engine and electric motors. The vehicle's power-split drivetrain and longitudinal dynamics were modeled in MapleSim, a modeling and simulation software, using a physics-based analytical approach. All model parameters were identified using Controller Area Network (CAN) data and measurements of wheel torque data that were gathered during vehicle road testing. Using the grey-box powertrain model as a framework, a look-ahead linear time-varying (LTV) model predictive controller (MPC) for reference velocity tracking is proposed. Using some simplifying assumptions about the powertrain dynamics, the control-oriented model was reformulated in a pseudo-Hammerstein form. Inversion of the nonlinearities allows linear MPC algorithms to be applied directly to the linear portion of the system. The performance of the MPC was tested using multiple model in the loop (MIL) reference velocity tracking scenarios, and benchmarked against a tuned proportional-integral (PI) controller. Using the novel control-oriented model of the OEM powertrain, the MPC was found to track the desired velocity trajectory and reject measurable disturbance inputs, such as road slope, better than the PI controller.

Introduction

The introduction of hybrid electric vehicles (HEVs) to the automobile industry has been a significant part of a larger push for 'green' transportation options. HEVs combine the advantages of internal combustion vehicles and electric vehicles by integrating both power sources into a single powertrain system. By the year 2020 over one million new HEVs are expected to be sold in the US per year because of government incentives and shifting consumer preferences [1]. One of the most common forms of HEV available for purchase in the modern consumer vehicle market is the power-split HEV. The Toyota Prius, Ford Focus Hybrid, and Lincoln MKZ Hybrid are all examples of power-split HEVs. The powertrain of power-split HEVs has been designed such that it can switch powertrain operating modes to behave similarly to a pure electric vehicle, a series hybrid, or a parallel hybrid.

The versatility of power-split powertrains makes them significantly more complex than the powertrains of internal combustion vehicles, electric vehicles, or series and parallel HEVs. As a result, modeling the dynamics of power-split powertrains for the purposes of vehicle dynamics control or energy management control can be challenging. Previous work in the literature has modeled similar power-split powertrains using primarily analytical models of the engine, electric motors, battery, and drivetrain [2] [3], but such approaches require significant a priori knowledge of the powertrain control architecture and component designs. Other approaches to modeling of power-split powertrains have relied heavily on experimentally determined maps of individual component performance [4]. Fully experimental approaches result in highly accurate models of system performance, but they require powertrain disassembly and extensive testing of individual components. The Lincoln MKZ Hybrid powertrain model proposed in this paper combines the analytical approach with the experimental approach by introducing a grey-box modeling method. The control architecture and power source dynamics of the powertrain are modeled with neural networks while the drivetrain is modeled using a physics-based approach. Related works in the literature have used NNs to design energy management systems for hybrid vehicles [5], however using an NN to emulate an existing HEV torque controller appears to be a unique application. The proposed model was defined with limited a priori knowledge of the system's architecture, and is identified entirely from full vehicle road testing.

A unique model predictive control (MPC) approach to longitudinal vehicle dynamics control for power-split HEVs is also proposed in this paper as a proof-of-concept. A linear time-varying (LTV) MPC for reference velocity tracking was designed by using the grey-box powertrain model as a framework and converting it to a control-oriented formulation. Application of neural network models in predictive control for automotive application has been explored frequently in literature [6] [7] [8]. For HEV applications, neural network based model predictive control has primarily been used for design of energy management controllers [9] [10]. The application of neural network hybrid powertrain models in vehicle dynamics MPC has not been previously studied.

MPC for longitudinal dynamics has been explored frequently in literature [11] [12], however most work has assumed that driveshaft torque is applied directly to the wheels without considering powertrain dynamics. The proposed longitudinal dynamics controller considers the neural network and physics-based powertrain dynamics as a part of the MPC control-oriented model. The reformulated control-oriented model of the powertrain takes the form of a pseudo-Hammerstein model, which allows a linear MPC approach to be applied.

Data Acquisition

Experimental data used for model parameter identification was acquired using two data acquisition systems. The first data acquisition system is a Vehicle Measurement System (VMS) by A&D Technologies. The VMS consists of three primary sensors which are mounted at each wheel: The Wheel Force Sensor (WFS), Wheel Position Sensor (WPS), and Laser Ground Sensor/Laser Doppler Velocimeter. Figure 1 depicts a portion of the VMS system mounted on front right wheel of the Lincoln MKZ.

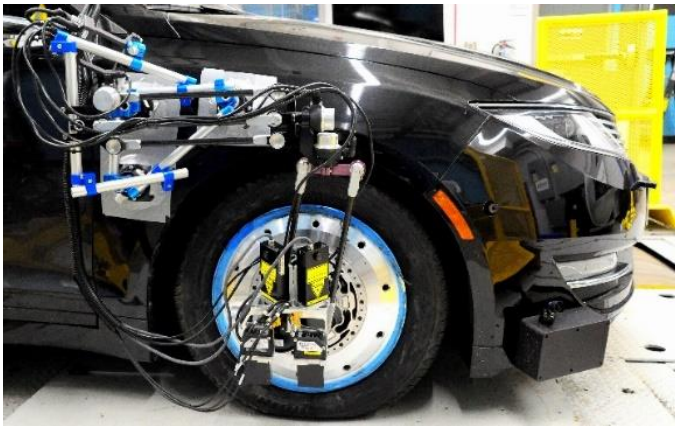


Figure 1. VMS system mounted to the front right wheel of the Lincoln MKZ.

For the purposes of powertrain parameter identification only the WFS and WPS were required. The WFS is a custom hub which is mounted with an array of strain gauges. From the strain gauges, forces and moments in all three principal axes of wheel are determined. The WPS consists of a 6-DOF arrangement of digital encoders. This allows the system to detect the position and orientation of each wheel relative to the chassis. Throughout track testing the VMS system was used to measure force and torque at each wheel as well as wheel angular velocity and acceleration.

The second data acquisition system is a Vector CANalyzer hardware and software tool. The CANalyzer was used to measure various internal signals from the Lincoln MKZ’s Control Area Network (CAN) bus. Signals measured using the CANalyzer include: battery state of charge (SOC), accelerator pedal position (APP), vehicle speed (v_x), desired total driveshaft torque (T_{dsd}), as well as speeds and torques of the generator, traction motor, and engine.

Experimental data was acquired using a combination of road and track testing of the Lincoln MKZ. Road testing was used exclusively to gather data for neural network identification of the supervisory controller subsystem and the TCM and power source subsystem, as depicted in figure 2. Track testing was used as a controlled environment to gather data with the VMS which subsequently was used for parameter identification of the drivetrain subsystem.

High-Fidelity Powertrain System Modeling and Parameter Identification

The powertrain of the Lincoln MKZ Hybrid is controlled using a complex power-split hybrid energy management system. Figure 2 depicts the full system model. For the purposes of this paper, the powertrain system was divided into three primary subsystems that were each modelled and identified separately. The first subsystem is a model of the mapping from accelerator pedal position (APP) to desired total powertrain output torque, T_{dsd}, which is calculated by the vehicle’s supervisory controller. At this stage of the modeling process the system’s response to brake pedal input is neglected; modeling of braking dynamics is considered to be outside of the scope of the paper.

The second subsystem is a model of the low-level vehicle torque control module (TCM), which determines how torque is allocated to each powertrain power source, as well as the dynamics of the power sources themselves. The third subsystem is a model of the vehicle’s power-split drivetrain dynamics and its interaction with the vehicle’s tire and suspension dynamics. For simplicity the steering dynamics and other complexities of the vehicle model are ignored. High fidelity vehicle dynamics modeling of the Lincoln MKZ is outside the scope of this work. The details of each subsystem model are described below.

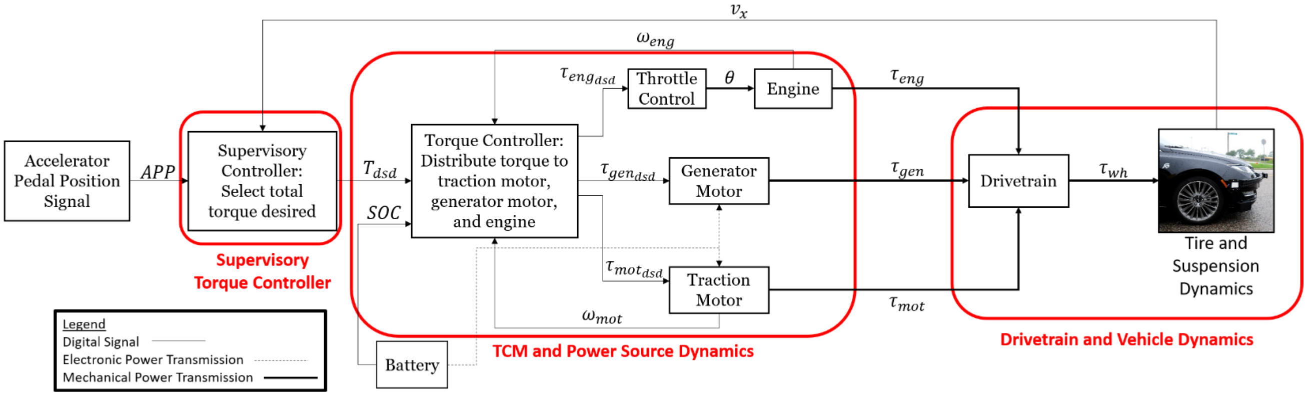

v_x
ω_{eng}
$\tau_{eng_{dsd}}$
Throttle Control
θ
Engine
τ_{eng}
Accelerator Pedal Position Signal
APP
Supervisory Controller: Select total torque desired
T_{dsd}
SOC
Torque Controller: Distribute torque to traction motor, generator motor, and engine
$\tau_{gen_{dsd}}$
Generator Motor
τ_{gen}
Drivetrain
τ_{wh}
Tire and Suspension Dynamics
$\tau_{mot_{dsd}}$
Traction Motor
τ_{mot}
ω_{mot}
Supervisory Torque Controller
Legend
Digital Signal
Electronic Power Transmission
Mechanical Power Transmission
Battery
TCM and Power Source Dynamics
Drivetrain and Vehicle Dynamics

Figure 2. Diagram depicting an overview of the Lincoln MKZ 's hybrid powertrain control system architecture.

Supervisory Torque Controller Subsystem

The supervisory torque controller of the Lincoln MKZ determines the overall desired driveshaft torque, T_{dsd}, as a function of APP and v_x. In addition, a transport delay exists in the desired torque response. The response of the supervisory controller to these inputs is highly nonlinear.

For the purposes of this paper the supervisory controller was modeled using a double-layer perceptron neural network with a time-delayed feedback of the output variable and a timestep of 0.05 seconds. This form of neural network is well-suited for modeling the system since it is a universal approximator of non-linear functions [13].The three inputs to the network are APP, v_x, and T_{dsd} at the previous time step, and the output is T_{dsd} at the current time step. The neural network was constructed, tuned, and validated with the Matlab Neural Network Toolbox against experimental data using the Levenberg-Marquardt algorithm. Figure 3 depicts the supervisory controller mapping at points of equilibrium (previous T_{dsd} equal to current T_{dsd}).

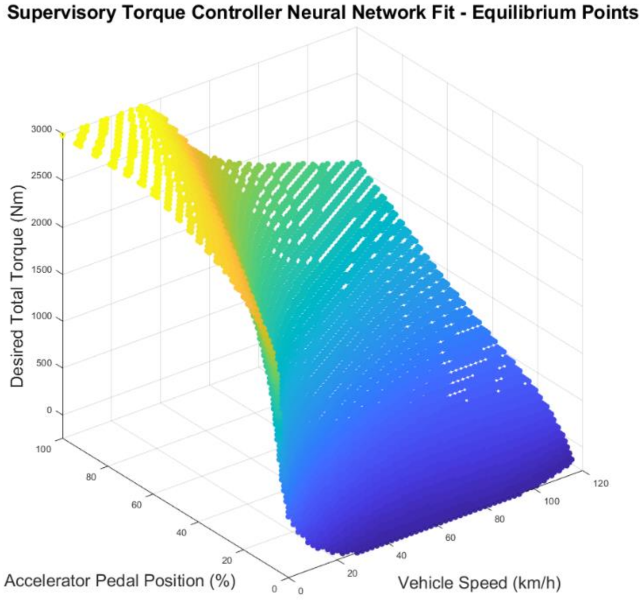


Figure 3. Equilibrium points plot depicting the neural network approximation of the supervisory control model.

The fitted neural network had a regression coefficient of 0.995, indicating a strong fit to the experimental data. However, analysis of experimental data shows that there was limited data gathered for APP between 50% and 100%. Future work will focus on improving the training of the neural network by gathering a more complete experimental data set.

TCM and Power Source Dynamics Subsystem

For the purposes of the model, the TCM, traction motor dynamics, generator dynamics, and engine dynamics were lumped into a single subsystem. This choice is due to the interdependent behavior of each of these portions of the control system. The high voltage battery is used to regulate charge of the vehicle's low voltage battery, which in turn powers onboard electronics, air conditioning, and other vehicle systems. Because of these factors, SOC of the high voltage battery was found to depend on several factors external to the powertrain model. SOC was treated as a random and measurable disturbance variable for the TCM and power source dynamics subsystem.

The TCM of the power-split hybrid powertrain is more complex than traditional internal combustion or fully electric vehicles because there are multiple modes of operation to consider. Each operating mode adjusts how desired torque is distributed to the power sources and alters the kinematic constraints of the drivetrain. In the case of the Lincoln MKZ's power-split powertrain, discrete modes of operation were identified by observation of experimental data and by referencing the model of a previous generation of the powertrain described in [2]. The following modes were identified for forward driving:

1. EV mode: This mode is engaged when torque demand is sufficiently low and battery SOC is sufficiently high for the engine to remain off. In this mode the engine shaft is locked in place.
2. Engine cranking: The TCM has determined that power output from the engine is required, and the engine shaft has been unlocked. Positive torque is produced by the generator to accelerate the engine up to its ignition speed.
3. Power-split mode: At high speeds or high torque demands the engine is running and being used to provide torque to the driveshafts. Depending on overall desired torque the generator and traction motor may be used to either charge the high voltage battery or transmit additional torque to the driveshafts. Two sub-modes of power-split mode exist.
    a. Positive-split: If the high voltage battery is below a threshold SOC, then torque from the engine splits between the path to the driveshaft and the path through the generator to charge the high voltage battery.
    b. Negative-split: This mode is not preferred but is necessary when the high voltage battery is fully charged and the vehicle speed is high. The generator transmits torque through the planetary gear train to drive the vehicle. Due to the kinematics of the planetary gear set, torque from the generator also regulates the engine speed to keep it in the high efficiency operating range. Negative-split mode establishes a power circulation path where some of the power produced by the generator returns to the high-voltage battery through the traction motor (negative motor torque value). Power circulation results in negative-split being a less efficient mode of operation than positive-split.

Additional modes of operation exist for when the vehicle is parked or driving in reverse, but modeling them is outside of the scope of this paper. Figure 4 depicts a sample window of experimental data where each operating mode is observed.

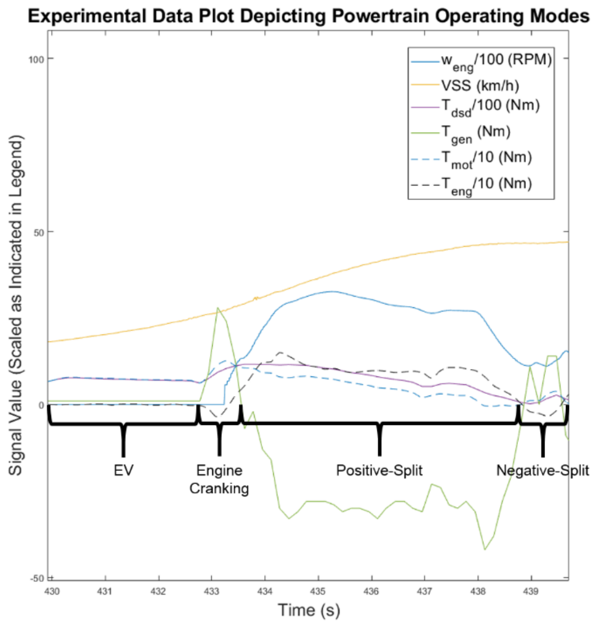


Figure 4. Plot of a sample window of experimental powertrain performance data depicting each of the powertrain operating modes.

Each of the three operating modes were denoted by indices 1,2, and 3, respectively (positive and negative-split were not differentiated by mode). Each experimental data sample was then categorized into one of the operating modes based on measured engine speed, generator torque, and operating mode of the previous time step.

A system of two shallow neural network classifiers was constructed and trained with the Matlab Neural Network Toolbox to predict the behavior of the TCM switching between operating modes based on current states of the powertrain. Both classifiers are two-layer feed-forward networks with a hidden sigmoid layer and a softmax output layer. The first classifier network predicts the conditions for the TCM to begin starting the engine (switch from EV mode to engine cranking mode). The network predicts if the mode will switch to engine cranking mode based on the following inputs: v_x, SOC, and T_{dsd}. The best training performance without overfitting the data was observed for a network with 3 hidden neurons. After training the network using scaled conjugate gradient backpropagation, the classification network was found to correctly identify whether or not engine cranking would begin in 98.8% of experimental data samples.

The second classifier network is used to determine the current TCM mode if the engine shaft is currently rotating. The network predicts if the TCM would select engine cranking mode, power-split mode, or EV mode, meaning the engine shaft begins braking, based on the following inputs: v_x, SOC, T_{dsd}, ω_{eng}, and the operating mode of the TCM at the previous sample time. The network was also trained using scaled conjugate gradient backpropagation, and optimal performance was observed for a network with 3 hidden neurons. The trained classification network was found to correctly identify the TCM mode in 99.7% of experimental data samples.

Once the TCM has selected the powertrain operating mode, torque demand is distributed to the engine, generator, and traction motor. The low-level dynamics of each power source determines their actual torque outputs τ_{eng}, τ_{gen}, and τ_{mot}, respectively. Limitations of the CANalyzer hardware prevented desired and actual torque signals from being measured at a sample rate greater than 10Hz. The sample rate was insufficient to differentiate between desired torque and actual torques, so the low-level dynamics of each power source were neglected. Instead the mapping from T_{dsd} to τ_{eng}, τ_{gen}, and τ_{mot} was modeled using an individual double-layer perceptron neural network. The inputs to the neural network are: T_{dsd}, ω_{eng}, ω_{mot}, $mode$, and SOC. Based on these inputs it is evident that the network assumes quasi-static behavior of each power source, so the model does not include transient behavior.

Despite these limitations, the tuned torque mapping neural network was found to fit well to experimental data. Optimal performance without overfitting data was observed for a network with 5 hidden layers. The network was tuned and validated against experimental data using the Levenberg-Marquardt algorithm. The fitted network was found to have a regression coefficient of 0.972 compared to the experimental data.

Drivetrain and Vehicle Dynamics Subsystem

The final subsystem to be modeled was the drivetrain and vehicle dynamics; however identification of a high-fidelity vehicle dynamics model for the Lincoln MKZ is outside the scope of this paper. The focus of this section will be on modeling and identification of the power-split drivetrain dynamics, and a simplified longitudinal vehicle dynamics model introduced for the purpose of controller simulations.

Drivetrain Model

The power-split drivetrain configuration used by the Lincoln MKZ Hybrid is depicted in figure 5.

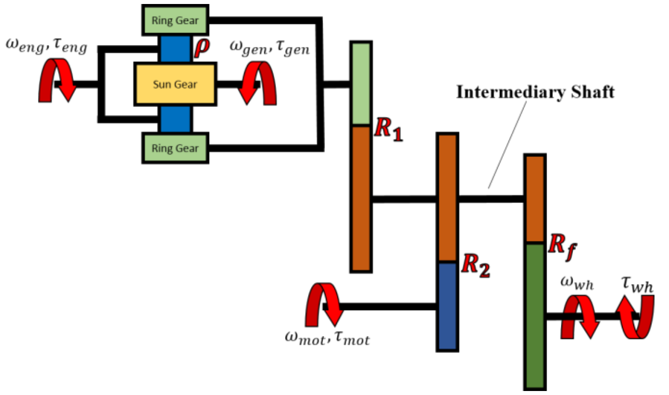


Figure 5. Model of the Lincoln MKZ's power-split-powertrain.

Arrows indicate how positive directions of rotation, ω, and torque, τ, are defined. As depicted in figure 5, the system comprises a planetary gear set that is used to connect the output shafts of all three power sources. The generator is used to drive the sun gear of the planetary while the engine is used to drive the planet carrier. The planetary gear ratio is defined by equation 1:

$$\rho = \frac{d_{sun}}{d_{ring}} \qquad (1)$$

where d_{sun}, and d_{ring} are the pitch diameters of the sun and ring gears, respectively. The ring gear is connected through fixed gear ratios R_1 and R_2 to the output shaft of the traction motor. Similarly, both the ring gear and traction motor shaft are connected through the intermediary shaft and final reduction ratio R_f to the driveshaft and front wheels. For the purposes of this model the differential has been neglected, and both driven wheels are assumed to always spin with the same speed ω_{wh}.

Unlike the other two subsystems, sufficient measured drivetrain data was available to identify a physics-based model. Based on the geometry of the drivetrain depicted in figure 5, the model has only two degrees of freedom (DOF). For the purposes of this paper the DOF are defined as ω_{eng}, and ω_{mot}. The kinematics and dynamics of the drivetrain are dependent upon the powertrain operating mode. When operating in EV mode ω_{eng} is set to zero, and the kinematics of the system are defined by equations 2 and 3a:

$$\omega_{wh} = \frac{\omega_{mot}}{R_2 R_f} \qquad (2)$$

$$\omega_{gen} = -\frac{R_1}{R_2\rho}\omega_{mot} \qquad (3a)$$

The gear train efficiency losses are dependent upon direction of torque transmission at each gear meshing, but efficiencies were assumed to be constant and equal in each direction of torque transmission. For forward velocity and acceleration in EV mode, the dynamics are defined by equations 4, 5, and 6a:

$$I_{gen}\dot{\omega}_{gen} = \tau_{gen} - \tau_s \qquad (4)$$

$$I_{mot}\dot{\omega}_{mot} = \tau_{mot} - \frac{1}{R_f R_2 \eta_f \eta_2}\tau_{wh} - \frac{R_1 \eta_{r1}}{R_2 \eta_f \eta_s \rho}\tau_s \qquad (5)$$

$$0 = \tau_{eng} + \frac{1}{\eta_s}\left(1 + \frac{1}{\rho}\right)\tau_s \qquad (6a)$$

where τ_s is the torque at the sun gear applied by the planets, and I_{gen} and I_{mot} are the lumped rotational inertias of the generator and the motor, respectively. I_{gen} includes the inertias of the generator and sun gear. I_{mot} includes the inertias of the motor, ring gear, intermediary shaft, gear meshings R_1, R_2 and R_f, the driveshafts, and front wheels. The components of I_{mot} are lumped at the motor. Table 1 summarizes the efficiency terms.

Table 1. Summary of gear train efficiency terms.

η_{r1}	Lumped efficiency between the ring gear and the planet gears and through gear ratio R_1
η_2	Efficiency through ratio R_2
η_f	Efficiency through ratio R_f
η_s	Efficiency between the sun gear and the planet gears

For the kinematics of engine cranking mode or power-split mode, equation 3a is replaced by equation 3b:

$$\omega_{gen} = \frac{1+\rho}{\rho}\omega_{eng} - \frac{R_1}{R_2\rho}\omega_{mot} \qquad (3b)$$

Similarly, dynamics equation 6a is replaced by equation 6b:

$$I_{eng}\dot{\omega}_{eng} = \tau_{eng} + \frac{1}{\eta_s}\left(1 + \frac{1}{\rho}\right)\tau_s \qquad (6b)$$

where I_{eng} is the lumped rotational inertia of the engine, which includes the inertias of the engine, carrier, and planets about the axis of the engine shaft. For power-split powertrains the inertia components for rotation of the planet gears about their own axes are typically considered very small [2], so they are neglected in this model.

Drivetrain Parameter Identification

Due to limited a priori knowledge of the actual construction of the drivetrain, all gear ratios, gear train efficiencies, and lumped inertia parameters needed to be identified from experimental data. Gear train ratios could be identified from experimental measurements of ω_{eng}, ω_{gen}, ω_{mot}, and ω_{wh} and equations 2 and 3b. However, without a measurement of speed for the intermediary shaft, not all gear ratios could be identified individually. Instead of identifying R_1, R_2, and R_f directly, lumped gear ratios $R_f R_1$ and $R_f R_2$ were identified. Identification of gear ratios was performed by determining the mean ratios between measurements of ω_{eng}, ω_{gen}, ω_{mot}, and ω_{wh} from multiple acceleration test runs. Each acceleration test was performed at constant APP to prevent discontinuous changes in rate of acceleration from affecting measurements. Each ratio was determined with a standard deviation of less than 1%. Table 2 summarizes the mean and standard deviation of each effective gear ratio.

Table 2. Identified effective drivetrain gear ratios.

Symbol	**Mean Value**	**Standard Deviation**
ρ	0.392	0.001
$R_f R_1$	4.06	0.03
$R_f R_2$	10.3	0.05

Parameter identification of the dynamic parameters, η_2, η_f, η_{r1}, η_s, I_{mot}, I_{gen}, and I_{eng} were complicated by the fact that all seven parameters had to be identified for a system with only two DOF. It is unlikely that the correct set of parameters could be identified from experimental measurements using traditional least-squares performance optimization. A sensitivity analysis was performed on the drivetrain model to observe how model performance was affected by a random variance in each parameter within their range of possible values. Time-varying signals for τ_{mot}, τ_{gen}, τ_{eng}, and τ_{wh} were pulled from experimental data and used to repeat simulations with 1000 different randomly distributed sets of parameters. The measured outputs ω_{mot} and ω_{eng} were used to compare simulation performances. Figure 6 is a tornado plot summarizing the correlation results of the sensitivity analysis.

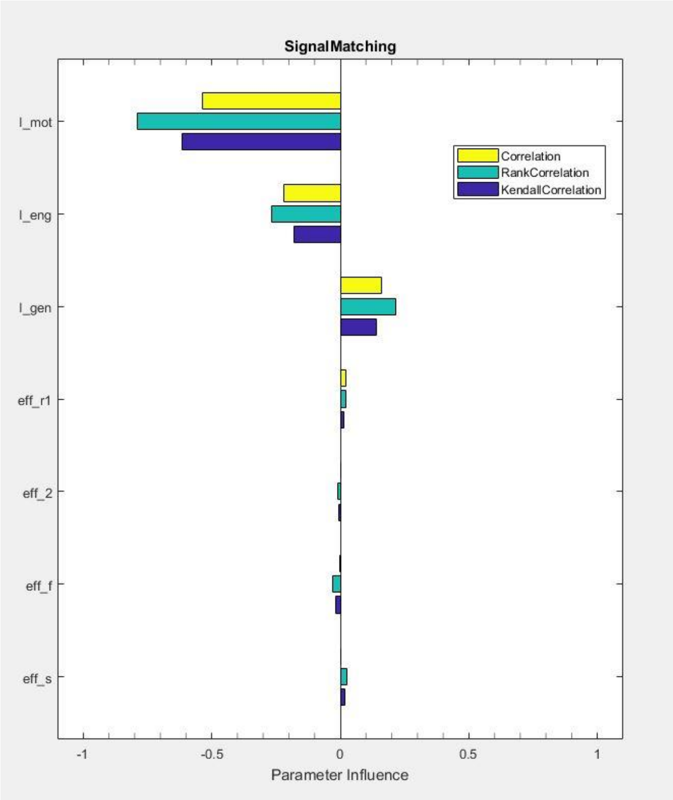


Figure 6: Tornado plot of correlations of variance in drivetrain parameters and model performance.

It is evident from the results of Kendal, Spearman (rank), and linear correlation that performance of the drivetrain model is much more responsive to variance in inertial parameters than variance in individual efficiency values. Based on the results of the sensitivity analysis, the model was simplified by setting each gear meshing efficiency to a typical constant value 0.98.

The remaining three parameters, I_{mot}, I_{gen}, and I_{eng}, were identified from track testing data using the Matlab Parameter Estimation Toolbox. The drivetrain model was iteratively simulated with inputs of τ_{eng}, τ_{gen}, τ_{mot}, and τ_{wh} measured from several experimental track tests. The outputs of ω_{eng} and ω_{mot} from the model were compared to experimental measurements with a sum-squared error cost function. A nonlinear least squares trust-region reflective algorithm was used to optimize the cost function. Noise in experimental data prevented the parameter estimation algorithm from consistently converging on the same values for I_{mot}, I_{gen}, and I_{eng}, so final parameter estimates were approximated based on several varying optimization results. Table 3 summarizes the approximated values of the lumped inertia parameters.

Table 3. Estimated rotational inertia parameters of the drivetrain.

Symbol	**Estimated Rotational Inertia (kg*m^2)**
I_{mot}	0.95
I_{gen}	0.024
I_{eng}	0.0025

Simplified Vehicle Dynamics Model

Modeling and parameter ID of the vehicle dynamics of the Lincoln MKZ was outside the scope of this work. For proof-of-concept of the linearized model-predictive longitudinal dynamics controller, only a simple longitudinal vehicle model with linear tires was used. Future work will integrate a high-fidelity MapleSim multibody vehicle dynamics model and combined slip Pacejka [14] tire model of the Lincoln MKZ, described in [15], with the hybrid powertrain model discussed herein. Figure 7 depicts a diagram of the longitudinal dynamics model.

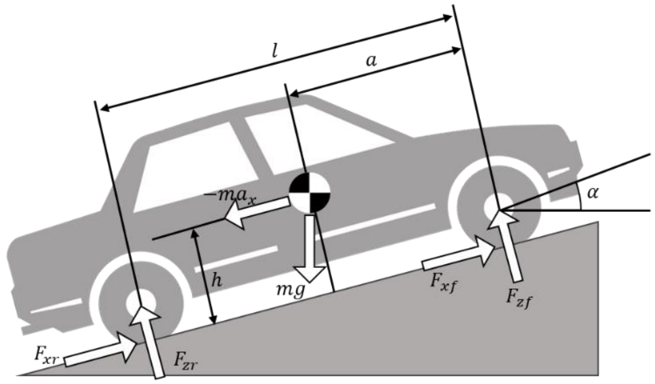


Figure 7. Simplified longitudinal vehicle model.

The acceleration of the vehicle chassis is determined by the longitudinal tire forces F_{xf} and F_{xr}. The equation of motion for the chassis's single DOF, translation in the x direction, is defined by equation 7:

$$ma_x = F_{xf} + F_{xr} - \frac{1}{2}C_d\rho_{air}A_f v_x^2 - mg(C_r v_x + sin(\alpha)) \quad (8)$$

where the additional terms represent air drag force, rolling resistance force, and force due to road slope α, respectively. For a complete table of longitudinal dynamics model variables and their definitions refer to the appendix. The longitudinal tire forces are dependent upon slip and normal tire forces F_{zf} and F_{zr}. These forces are determined by equations 9 and 10:

$$F_{zF} = mgcos(\alpha) * \frac{l-a}{l} - \frac{h}{l}(ma_x - \frac{1}{2}C_d\rho_{air}A_f v_x^2 - \ldots$$

$$mg(C_r v_x + sin(\alpha)))$$

$$(9)$$

$$F_{zR} = mgcos(\alpha) - F_{zF} \qquad (10)$$

The simplified longitudinal vehicle model uses linear tires with saturation. For the front and rear axles, the longitudinal tire force is determined by equations 11 and 12, respectively:

$$F_{xf} = min(F_{zf}\sigma C_l, F_{zf}\mu) \qquad (11)$$

$$F_{xr} = min(F_{zr}\sigma C_l, F_{zr}\mu) \qquad (12)$$

where C_l is the normalized longitudinal tire stiffness (assumed to be the same at each wheel), and σ is the slip ratio. σ is defined by equation 13:

$$\sigma = \frac{\omega_{wh} r_{wh} - v_x}{\max(\omega_{wh} r_{wh}, v_x)} \qquad (13)$$

where r_{wh} is the effective wheel radius. As shown in equations 11 and 12, the longitudinal tire force will increase linearly with σ until a defined maximum longitudinal coefficient of friction, μ.

The rotational inertia of the rear axle has been lumped with the mass of the chassis, m, because the model does not consider braking and the Lincoln MKZ is front-wheel drive. As a result F_{xr} is always zero. Lumping the wheel inertia with I_{mot}, as indicated in equation 5, has also simplified the relationship between longitudinal wheel force and wheel torque to equation 14:

$$\tau_{wh} = F_{xf} r_{wh} \qquad (14)$$

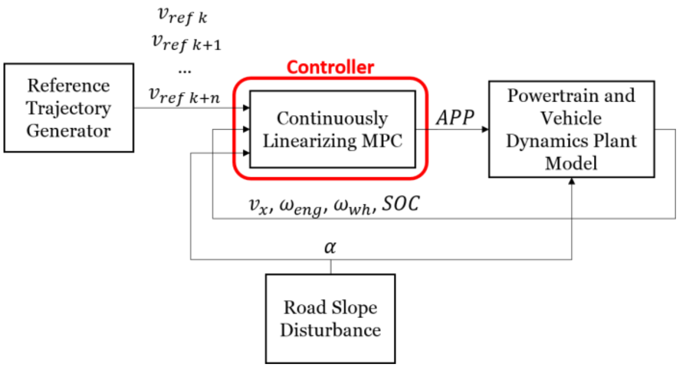
The combined subsystem model of the drivetrain and longitudinal vehicle dynamics was constructed in MapleSim. The acausal modeling environment of MapleSim allowed for modeling of the power-split drivetrain despite the bi-directional torque transmission through many of the gear meshings. An S-function of the model was exported from MapleSim for integration with the other subsystem models in Simulink.

Model Predictive Longitudinal Dynamics Controller

Controller Design

Based on the hybrid neural network model of the Lincoln MKZ's power-split powertrain described in this paper, a unique approach to linearized MPC for longitudinal dynamics control was devised for a proof-of-concept simulation. The proposed controller must follow a reference longitudinal velocity trajectory, $v_x(k)$, which is always known for a finite prediction horizon. A similar velocity reference tracking problem is typical for adaptive cruise control or autonomous driving applications.

Linearized MPC is made possible for this system by taking advantage of the neural network formulation of the powertrain model. By inverting some neural network portions of the powertrain system model and utilizing the quasi-static behavior of other portions of the model, the control-oriented model is reformulated. The formulation of the control-oriented model allows the model to be controlled using a typical linearized MPC approach. Figure 8 depicts a simplified block diagram of how the MPC controller connects with the powertrain and vehicle dynamics plant model for model in the loop (MIL) simulation.



$v_{ref\ k}$
$v_{ref\ k+1}$
...
$v_{ref\ k+n}$
Reference Trajectory Generator
Controller
Continuously Linearizing MPC
APP
Powertrain and Vehicle Dynamics Plant Model
$v_x, \omega_{eng}, \omega_{wh}, SOC$
α
Road Slope Disturbance

Figure 8: Block diagram depicting the controller in the loop with the powertrain and vehicle dynamics plant model.

The controller has been designed such that it can be used to control the plant model without directly measuring diagnostics signals such as torques and speeds at the traction motor and generator. Diagnostics signals are unlikely to be available for implementation on the actual vehicle. As a result, only plant model states v_x, ω_{eng}, and ω_{wh} as well as battery SOC are fed back from the plant as fully measurable inputs to the controller. Road slope α is also assumed to be measurable.

For the MPC's control-oriented model, the supervisory controller neural network model was inverted such that v_x, previous desired T_{dsd}, and current desired T_{dsd} are used to select the desired APP signal. The inverse model was constructed with the Matlab Neural Network Toolbox and tuned with the Levenberg-Marquardt algorithm. The same experimental data that was used for the forward neural network model was also used for the inverse neural network, and three hidden neurons was found to give optimal performance. Figure 9 depicts the supervisory controller inverse mapping at points of equilibrium (previous T_{dsd} equal to current T_{dsd}).

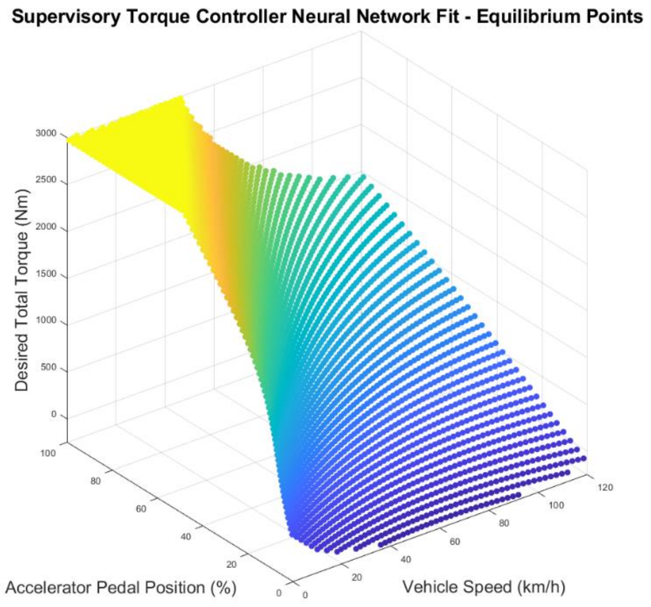


Figure 9: Equilibrium points plot depicting the neural network approximation of the supervisory control model.

The supervisory controller inverse neural network was found to have a regression coefficient of 0.975. The symbolic computing capabilities of Maple were used to extract an optimized nonlinear inverse neural network function for determining APP, hereafter denoted by $INN(v_x, T_{dsd_{k}-1}, T_{dsd_k})$.

The continuously linearizing MPC controller was created by combining components of the TCM and power source subsystem with the inverse supervisory controller model, and the drivetrain and vehicle model subsystem. The controller contains two non-linearized continuous time state-space internal models. The models were formulated by combining and rearranging and modifying equations 5, 8, 11, 14 and either 6a (for engine shaft locked) or 6b (for engine shaft unlocked). Equation 15 represents the nonlinear state-space for when the engine is locked:

$$\dot{X} = \begin{bmatrix} \dot{v}_x \\ \dot{\omega}_{mot} \\ \dot{\omega}_{eng} \end{bmatrix} = \begin{bmatrix} \frac{F_{zf}\sigma C_l - \frac{1}{2}C_d\rho_{air}A_f v_x^2 - mg(C_r v_x + sin(\alpha)}{m} \\ \frac{\tau_{mot} - \frac{1}{R_f R_2}F_{zf}\sigma C_l r_{wh} - \frac{R_1}{R_2\rho}\tau_s}{I_{mot}} \\ 0 \end{bmatrix} \quad (15)$$

and equation 16 represents the nonlinear state-space for when the engine is unlocked:

$$\dot{X} = \begin{bmatrix} \dot{v}_x \\ \dot{\omega}_{mot} \\ \dot{\omega}_{eng} \end{bmatrix} = \begin{bmatrix} \frac{F_{zf}\sigma C_l - \frac{1}{2}C_d\rho_{air}A_f v_x^2 - mg(C_r v_x + sin(\alpha)}{m} \\ \frac{\tau_{mot} - \frac{1}{R_f R_2}F_{zf}\sigma C_l r_{wh} - \frac{R_1}{R_2\rho}\tau_s}{I_{mot}} \\ \frac{\tau_{eng} + \left(1 + \frac{1}{\rho}\right)\tau_s}{I_{eng}} \end{bmatrix} \quad (16)$$

Equations 15 and 16 as they appear in this paper have been simplified, for conciseness, by leaving in the terms σ, τ_s, and F_{zf}. Once the equations for these terms are substituted into equations 15 and 16 the dynamics are clearly coupled and highly nonlinear. For simplification of the controller, all gear meshing efficiencies have been ignored for the control-oriented model. The control-oriented model also does not account for the tire friction limit μ because it is assumed that the controller's response can be tuned to avoid reaching the tire saturation region.

At each time step, k, the controller uses the measurement of ω_{eng} and the previous desired torque estimate $T^*_{dsd\;k-1}$ to estimate the current operating TCM mode. This method of estimating the operating mode is reactive rather than predictive, and it will normally lag the true mode signal by one time step. The controller then uses the estimate of TCM mode to determine whether to use equation 15 or 16 in the MPC at the current time step.

From the current estimate of TCM mode, $T^*_{dsd\;k-1}$ and measurements of ω_{mot}, ω_{eng}, and SOC, the controller uses a copy of the torque mapping neural network to approximate the partial derivatives $\frac{\partial\tau^*_{eng}}{\partial T_{dsd}}$, $\frac{\partial\tau^*_{gen}}{\partial T_{dsd}}$, and $\frac{\partial\tau^*_{mot}}{\partial T_{dsd}}$. Simultaneously, the torques at the previous time step, $\tau^*_{eng\;k-1}$, $\tau^*_{gen\;k-1}$, and $\tau^*_{mot\;k-1}$ are estimated by the internal neural network approximator. The selected state-space equation is then modified by the linearization approximations in equations 17, 18, and 19:

$$\tau_{eng} \cong \tau^*_{eng\;k-1} + \frac{\partial\tau^*_{eng}}{\partial T_{dsd}}(T^*_{dsd} - T^*_{dsd\;k-1}) \quad (17)$$

$$\tau_{gen} \cong \tau^*_{gen\;k-1} + \frac{\partial\tau^*_{gen}}{\partial T_{dsd}}(T^*_{dsd} - T^*_{dsd\;k-1}) \quad (18)$$

$$\tau_{mot} \cong \tau^*_{mot\;k-1} + \frac{\partial\tau^*_{mot}}{\partial T_{dsd}}(T^*_{dsd} - T^*_{dsd\;k-1}) \quad (19)$$

where T^*_{dsd} is treated as the generalized control input, u, for any time step i such that $k-1 < i < k+n$.

The state space equation is converted to discrete time by forward Euler approximation and then linearized by computing the jacobian matrices A and B. To decrease computational cost, these matrices were determined symbolically offline using Maple. An estimate of the state increment, ΔX_{k-1}, at previous time step $k-1$ is also determined from the unlinearized discrete state-space equations. The final set of linearized state-space equations at the nominal operating point $k-1$ is represented by equations 20 and 21:

$$X_{i+1} = X_{k-1} + A(X_i - X_{k-1}) + B(u_i - u_{k-1}) + \Delta X_{k-1} \quad (20)$$

$$y_i = [1 \quad 0 \quad 0]X_i \quad (21)$$

The linear MPC was then defined by cost function in equation 22:

$$J = \sum_{i=k}^{k+n}(q\|y_i - v_{ref\;i}\|^2 + r\|\Delta u_i\|^2) \quad (22)$$

subject to $-250Nm < u < 2985Nm$ (corresponds to the limits of T_{dsd}), and $-5000Nm/s < \Delta u < 5000Nm/s$. The constraint on Δu was tuned to ensure that an impossible step change in T_{dsd} was never demanded from the supervisory controller. Relative weighting terms, q and r, were selected to be 1 and 0.1, respectively. The cost function is minimized using quadratic programming at each time step. Tuning of the controller resulted in selection of a timestep $Ts = 0.05\ s$, a prediction horizon of 20 steps, and a control horizon of 3 steps.

Controller Simulation Results

The velocity tracking performance of the linearized MPC controller was tested against several reference trajectories of v_{ref}. To assess the MPC, it was compared to a tuned PI controller as a performance benchmark. Random variation in measured disturbance variables SOC and α were added to the system to test the robustness of the controller. Figure 10 depicts the results from a reference signal v_{ref} which repeatedly ramps by increments of 5 m/s. This test was used to observe the acceleration performance of the controller at various speeds.

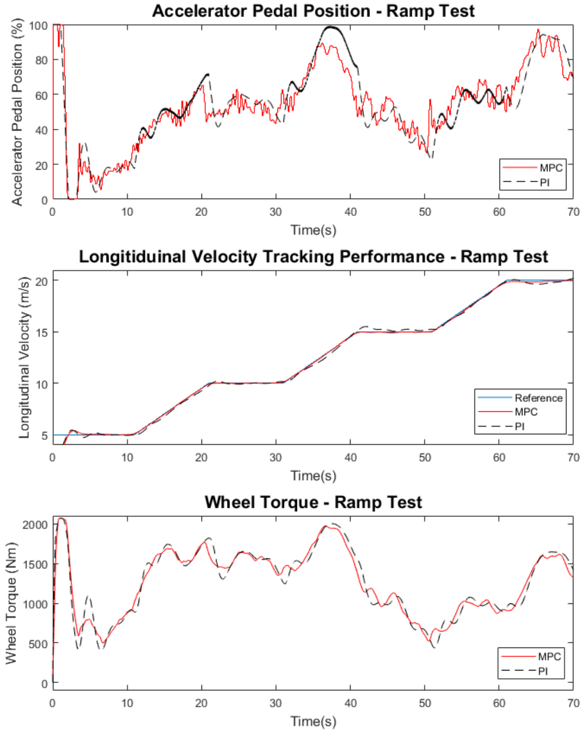


Figure 10: Plot of simulation results from the ramped reference velocity tracking test.

As depicted in figure 10, the MPC's ability to look ahead to future desired values of v_{ref} helps it to respond faster than the PI. The PI is also more sensitive to changes in α and SOC than the MPC. By building the effects of α and SOC into the control-oriented model, the MPC can better respond to their effects. The response of the MPC is also consistent regardless of current v_x. By comparison the PI controller's response varies considerably with v_x, and in particular it has difficulty tracking low speed references. This is likely due to the nonlinear change of τ_{eng}, τ_{gen}, τ_{mot}, and drag resistance force as a function of v_x. The APP signal from the MPC exhibits a much more oscillatory response than the PI controller. However, since the APP signal is subjected to a transfer delay through the plant model's supervisory controller, the wheel torque response was found to be largely smoothed. The MPC had a smoother velocity tracking response.

Figure 11 depicts the results of a test where v_{ref} oscillates with a sinusoidal pattern about a speed of 15 m/s. This test was used to imitate the circumstances of the vehicle following another car that is making small speed changes over time.

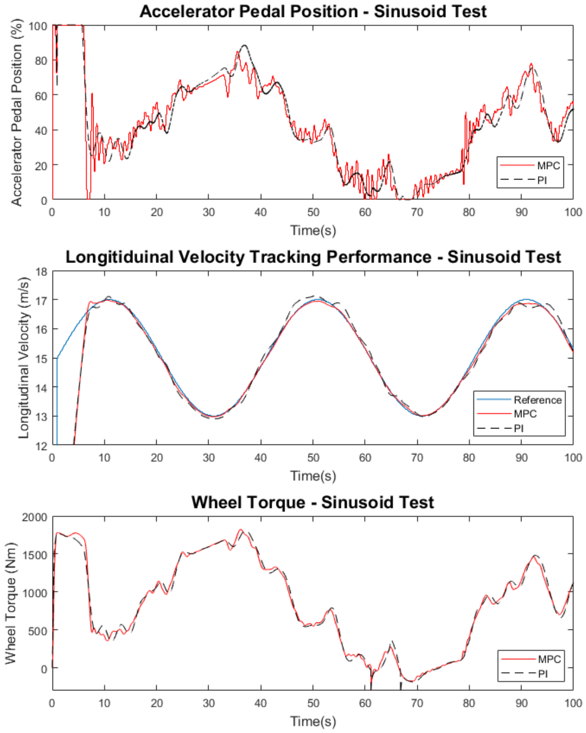


Figure 11: Plot of simulation results from the sinusoidal reference velocity tracking test.

The predictive ability of the MPC allows the controller to adjust to continuous changes in v_{ref}, and stay on the desired trajectory. Conversely the PI controller's performance is reactive, and it does not track the small changes in v_{ref} as effectively. The MPC also continues to respond better to changes in SOC and α, which inject disturbances into the model. Like the previous test, the APP signal from the MPC controller is much more oscillatory. Once again the highly oscillatory APP input from the MPC results in mostly smooth wheel torque signal.

Conclusions

The neural network and physics-based modeling approach described in this paper has been shown to be an effective method of modeling a power-split hybrid powertrain. Despite limited a priori knowledge of the Lincoln MKZ Hybrid's internal powertrain control architecture and power source dynamics, modeling of the system using neural networks was found to be an effective approximator of real performance. Performance of the neural network models can be further improved by gathering additional data. More powertrain performance data for APP inputs between 50% and 100% will help to improve the accuracy of each neural network map.

By modeling the drivetrain portion of the powertrain with a physics-based approach, transient inertial behavior is introduced to the model. The transient response to the drivetrain helps to calculate a more accurate output torque value τ_{wh}. This modeling approach to the powertrain is modular, so it can be connected to a tire and vehicle dynamics model with any level of complexity. Future work will focus on integrating the model with a high-fidelity MapleSim vehicle model and a braking force model for MIL full vehicle simulation.

The proof-of-concept MPC controller introduced in this paper was shown to be an effective approach for longitudinal dynamics control. The grey-box power-split powertrain can be reconfigured for control-oriented implementation. The pseudo-Hammerstein formulation of the control-oriented model allowed linearized MPC control to be implemented, but nonlinear MPC (NMPC) algorithms could also be explored. Future work will investigate NMPC approaches to control of shallow neural network based control systems with sigmoid activation functions.

References

1. Al-alawi, B., and Bradley, T., "Review of hybrid, plug-in hybrid, and electric vehicle market," *Renewable and Sustainable Energy Reviews* 21:190-203, 2013, doi: 10.1016/j.rser.2012.12.048.
2. Syed, F., Kuang, M., Czubay, J., and Ying, H., "Derivation and Experimental Validation of a Power-Split Hybrid Electric Vehicle Model," *IEEE Transactions on Vehicular Technology* 55(6):1731-1747, 2006, doi: 10.1109/TVT.2006.878563.
3. Liu, J., Peng, H., and Filipi, Z., "Modeling and Analysis of the Toyota Hybrid System," presented at *International Conference on Advanced Intelligent Mechatronics*, Monteray, CA, USA, July 24-28, 2008.
4. Hofman, T., Steinbuch, M., and Druten, R., "Modeling for Simulation of Hybrid Drivetrain Components," presented at *Vehicle Power and Propulsion Conference.* Windsor, UK, September 6-8, 2006, doi: 10.1109/VPPC.2006.364269
5. Baumann, B., Rizzoni, G., and Washington, G., "Intelligent Control of Hybrid Vehicles Using Neural Networks and Fuzzy Logic," SAE Technical Paper 981061, 1998, doi: 10.4271/981061
6. Wang, S., Yu, D., Gomm, J., Page, G., and Douglas, S., "Adaptive neural network model based predictive control for air-fuel ratio of SI engines," *Engineering Applications of Artificial Intelligence* 19(2):189-200, 2006, doi: j.engappai.2005.08.005
7. Pan, Y., and Wang, J., "Model Predictive Control of Unknown Nonlinear Dynamical Systems Based on Recurrent Neural Networks," *IEEE Transactions on Industrial Electronics* 59(8):3089-3101, 2011, doi: 10.1109/TIE.2011.2169636
8. Wang, S., Yu, D., Gomm, J., Page, G., and Douglas, S., "Adaptive neural network model based predictive control of an internal combustion engine with a new optimization algorithm," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 220(2):195-208, 2006, doi: 10.1243/095440706X72754.
9. Sun, C., Hu, X., Moura, S., and Sun, F., "Velocity Predictors for Predictive Energy Management in Hybrid Electric Vehicles," *IEEE Transactions on Control Systems Technology* 23(3):1197-1204, 2014, doi: 10.1109/TCST.2014.2359176.
10. Marx, M., Shen, X., and Söffker, D., "A Data-driven Online Identification and Control Optimization Approach to a Hybrid Electric Powertrain System," *IFAC Proceedings Volumes* 45(2):153-158, 2012, doi: 10.3182/20120215-3-AT-3016.00027
11. Borrelli, F., Bemporad, A., Fodor, M., and Hrovat, D., "An MPC/Hybrid System Approach to Traction Control," *IEEE Transactions on Control Systems Technology* 14(3):541-552, 2006, doi: 1109/TCST.2005.860527
12. Corona, D., De Schutter, B., "Adaptive Cruise Control for a SMART Car: A Comparison Benchmark for MPC-PWA Control Methods," *IEEE Transactions on Control Systems Technology* 16(2):365-372, 2008, doi: 10.1109/TCST.2007.908212
13. Hornik, K., Stinchcombe, M., and White, H., "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks* 2(5):359-366, 1989, doi: 10.1016/0893-6080(89)90020-8
14. Pacejka, H., and Bakker, E., "The Magic Formula Tyre Model," *International Journal of Vehicle Mechanics and Mobility* 21(sup001):1-18, 1992, doi: 10.1080/00423119208969994
15. Van Gennip, M., and McPhee, J., "Parameter Identification for Combined Slip Tire Models using Vehicle Measurement System," presented at *SAE World Congress*, Detroit, MI, USA, April 10-12, 2018.

Contact Information

Bryce Hosking
MASc student,
University of Waterloo
Email: b2hoskin@uwaterloo.ca

Acknowledgements

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada and the Ontario Government for providing research funding in the form of scholarships. The authors would also like to thank MapleSoft for guidance, funding, and continued partnership throughout this research.

Appendix

Table 4: Table summarizes parameters of the simplified longitudinal vehicle dynamics model

Symbol	Description
F_x	Longitudinal tire force
F_z	Vertical tire force
r_{wh}	Tire effective radius
C_d	Coefficient of drag
A_f	Frontal vehicle area
ρ_{air}	Air density
v_x	Longitudinal vehicle speed
C_r	Rolling resistance coefficient
a_x	Longitudinal vehicle acceleration
m	Vehicle mass
l	Vehicle wheelbase
a	Distance from front axle to COM
h	COM height
α	Road slope angle
g	Acceleration due to gravity
σ	Longitudinal wheel slip
μ	Coefficient of friction between the tire and road
C_l	Longitudinal tire stiffness