Probabilistic Analysis of Electric Vehicle Energy Consumption Using MPC Speed Control and Nonlinear Battery Model

Jun Chen
ECE, Oakland University
Rochester, MI 48309, USA
Email: junchen@oakland.edu

Man Liang
UniSA STEM, Univ. of South Australia
Adelaide, 5000, Australia
Email: annie.liang@unisa.edu.au

Xu Ma
Optimization and Control Group, PNNL
Richland, WA 99352, USA
Email: xu.ma@pnnl.gov

Abstract—This paper conducts a probabilistic analysis of energy consumption of electric vehicle. In particular, the vehicle speed is controlled by a model predictive control (MPC) to follow given reference speed while minimizing energy consumption, and the battery is modeled by nonlinear dynamic equations. Speed tracking accuracy and energy economy of MPC speed control is evaluated on EPA Federal Test Procedure driving cycle, which is commonly used for city driving testing and includes an approximate driving distance of 17.77 km. Furthermore, to conduct probabilistic analysis, Monte Carlo approach is taken to simulate a total number of ten thousands of synthetic FTP driving cycles, each with different profile. Numerical results and conclusion are then drawn to confirm the robustness of MPC speed control and the environmental friendliness of electric vehicle. Insights on battery operations for maximum energy efficiency are also discussed.

I. INTRODUCTION

Model predictive control (MPC) can deal with linear and nonlinear dynamical systems while keeping states and input within specified regions. Due to such flexibility and capability, MPC has been one of the most popular control methods [1], and has been applied to vehicle systems [2]–[4], smart grids [5], power converters [6], and communications [7], etc. For each control loop, MPC solves a constrained optimization problem using prediction over a finite horizon based on a dynamical model. Theoretical properties of MPC have also been studied in literature, concerning its stability, robustness, and feasibility.

Recent efforts have been focused on applying MPC in automotive embedded systems. For example, [8] applied MPC for vehicle speed control in the context of connected and automated vehicle (CAV), particularly for heavy-duty vehicle platoon where the pedal information of lead vehicle is available to ego vehicles. Reference [9] applies speed profile shaping before MPC in order to reduce excessive acceleration and deceleration, while similar approach is taken in [10] that used a speed planar to ensure ride comfort. Besides longitudinal speed control, MPC has also been successful in lateral vehicle dynamics control and energy management. For example, [11], [12] applied MPC to optimize energy management in electric vehicle (EV). Reference [4] used MPC to coordinate steering and braking inputs for vehicle stability control, while [13] adopted MPC for simultaneous longitudinal and lateral control. MPC has also been investigated for possible usage in active safety [14].

After developing an MPC design for a given control problem, its performance and correctness are often demonstrated through simulation or experiment under specific maneuvers. Such approach, however, does not provide insights on the robustness with respect to different maneuvers. To alleviate this and to enhance MPC applicability, researchers have conducted investigation to incorporate a prior knowledge about the disturbance distribution into MPC design to guarantee robustness [15], [16], to cast the problem of finding control systems flaws into a optimal control problem [17], and to conduct stochastic sampling over control systems parameter space [18], [19].

Despite these promising research findings, a probabilistic analysis of MPC for longitudinal speed control is missing, leading to lack of insights on the average/worst/best MPC performance on a given cycle. In this paper, we address this limitation by using Monte Carlo approach. Firstly, we consider the vehicle longitudinal speed tracking problem, where MPC optimizes wheel torque inputs in order to track the given speed profile while minimizing energy consumption. Secondly, a higher fidelity model is then used as simulation plant, hence creating a model mismatch between MPC prediction and virtual vehicle. Finally, Monte Carlo approach is taken to simulate a total number of ten thousands synthetic FTP driving cycles, each with different profile, to investigate the robustness of MPC design. Compared to aforementioned work [15]–[17], this work does not aim at designing a robust MPC against known disturbance distribution, but rather to demonstrate and quantify the robustness of a given MPC design through simulation.

The rest of this paper is organized as follows. Section II introduces the vehicle dynamics model, together with the nonlinear battery model, which will be used as virtual plant for simulation. Section III presents the MPC speed control, while numerical results are discussed in Section IV. The paper is concluded in Section V.
TABLE I
PARAMETERS FOR MAGIC FORMULA (2).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>10</td>
</tr>
<tr>
<td>C</td>
<td>1.3</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
</tr>
<tr>
<td>E</td>
<td>0.97</td>
</tr>
</tbody>
</table>

TABLE II
TYPICAL VALUE FOR CALCULATING AERO DRAUGHT FORCE $F_a$ IN (4), ADOPTED FROM [20].

<table>
<thead>
<tr>
<th>Variable</th>
<th>Typical value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho$</td>
<td>1.225 kg/m$^3$</td>
</tr>
<tr>
<td>$C_d$</td>
<td>0.25 ~ 0.3</td>
</tr>
<tr>
<td>$A_F$</td>
<td>1.0 + 0.00056(m - 756)</td>
</tr>
</tbody>
</table>

II. VIRTUAL VEHICLE MODEL

A. Quarter Car Vehicle Dynamics

This section describes the quarter car model used in the virtual vehicle plant. More details can be found in [20], [21]. Consider the following quarter car longitudinal dynamics,

$$\dot{v}_x = \frac{n_w}{m}F_x - g \sin \sigma - \frac{1}{m}F_a$$  (1a)

$$\dot{\omega} = \frac{1}{I_w} \left( \frac{T}{n_w} - F_x R \right)$$  (1b)

where $n_w$ is the number of driving wheels, which equals 2 for front/rear driving vehicle and 4 for all wheel drive configuration, $\sigma$ is the road bank angle, $I_w$ is the wheel rotational inertial, $T$ is the total driving torque (as applied to all driving wheels) and $F_x$ is the total tire force, as computed by the following Magic formula [22]

$$F_x = F_z D \sin \left( C \arctan \left[ B s - E (Bs - \arctan(Bx)) \right] \right)$$  (2)

with the parameters $B$, $C$, $D$, $E$ given in Table I. Note that $F_x$ is normal force and $s$ is slip ratio defined as

$$s = \frac{\omega R - v_x}{v_x}$$  (3)

with $R$ being the wheel effective radius.

Finally, according to [20], the aerodynamic drag force $F_a$ in (2) can be modeled by (assuming wind speed is 0):

$$F_a = \frac{1}{2} \rho C_d A_F v_x^2$$  (4)

where $\rho$ is the air mass density, $C_d$ is the aerodynamic drag coefficient, $A_F$ is the effective front area. Typical values for $\rho$, $C_d$ and $A_F$ are listed in Table II.

B. Driveline Dynamic

The closed loop driveline (electric motor and transmission) dynamic is modeled as a first order transfer function, as follows

$$G(s) = \frac{T}{T_c} = \frac{1}{T_d s + 1}$$  (5)

where $T_c$ is the commanded wheel torque and $T$ is the actual delivered value, and $T_d$ represents the time constant of the driveline.

C. Nonlinear Battery Model

As will be seen shortly, MPC indirectly penalizes energy usage by minimizing wheel torque. In order to simulate the effectiveness of such approach, actual wheel torque $T$ and wheel speed $\omega$ are converted into power charge/discharge command $P_b$ to battery, and the battery state-of-charge (SOC) dynamic model is given as follows.

$$\dot{\zeta} = h(\zeta, P_b)$$  (6)

where $\zeta$ is the current battery SOC. Furthermore, in this study, a nonlinear battery SOC model is used, i.e., the right hand side $h(\zeta, P_b)$ of (6) is assumed to be nonlinear on $\zeta$ and linear on $P_b$. In particular, the nonlinear function (6) is specified by Figure 7, which is obtained by scaling the 3.2MWh battery presented in [23] to 66kWh.

D. FTP Driving Cycle

As mentioned earlier, EPA Federal Test Procedure (FTP) driving cycle is used to test the effectiveness and robustness of the MPC speed control. FTP driving cycle is commonly used for city driving testing, and includes an approximate driving distance of 17.77 km and average speed of 34.1 km/h [24]. Figure 2 plots the time profile of a typical FTP driving cycle. In this section, ten thousands synthetic FTP driving cycles are used to perform probabilistic analysis of the MPC control, which are generated by the autoregressive moving average model following the similar procedure as in [18]. In particular, Figure 3 plots a selected period of the FTP driving cycle, together with the standard deviation of the ten thousands cycles. Despite the variation introduced in the ten thousands driving cycles as demonstrated in Figure 3, the key statistic of each driving cycle still conform to a typical FTP cycle. For
III. MPC FOR SPEED TRACKING

This section details the prediction model used for MPC, followed by the formulation of a linear parametric varying MPC (LPV-MPC).

A. Model Linearization

Recall that model (1)-(4) is nonlinear. To simplify and linearize (1)-(4) for MPC to predict the system evolution and to compute optimal control action, one can assume that during the prediction steps the wheel is at steady state. In other words, throughout the prediction horizon

\[ \dot{\omega} = 0 \]

Therefore,

\[ F_x = \frac{T}{n_w R} \]

and

\[ \dot{v}_x = \frac{1}{m} T - g \sin \sigma - \frac{1}{m} F_a \]

Given nominal point \((T_0, v_{x0})\), (7) can be linearized as:

\[
\delta v_x = \frac{1}{m} T_0 - g \sin \sigma - \frac{1}{2m} \rho C_d A_F v_{x0}^2 \\
- \frac{1}{m} \rho C_d A_F v_{x0} \delta v_x + \frac{1}{m R} \delta T
\]

which can be discretized into the following difference equation using Euler’s method

\[
\delta v_x(t + 1) = \delta v_x(t) \\
+ \left[ \frac{1}{m} T_0 - g \sin \sigma - \frac{1}{2m} \rho C_d A_F v_{x0}^2 \right] T_s \\
- \frac{1}{m} \rho C_d A_F v_{x0} T_s \delta v_x(t) + \frac{1}{m R} T_s \delta T(t)
\]

where \(T_s\) is the sampling time and \(t\) denotes current discrete time index.

Remark 1: Note that the simplification, linearization, and discretization discussed in this session is for MPC prediction.
only, which will be utilized in the optimal control problem formulation presented shortly. For the virtual vehicle simulation, the full nonlinear model (1)-(4) is used.

B. Optimal Control Problem Formulation

At each sampling time $t$, the LPV-MPC linearized and discretize the nonlinear model to obtain the linearized prediction model (9) around the nominal operating point $(T(t), v_x(t))$, where $T(t)$ and $v_x(t)$ are feedback values. The following optimal control problem (OCP) is then formulated:

$$\min_{Z_t, U_t} J(Z_t, U_t) = \sum_{k=1}^{p} \left[ \frac{1}{Q_s} |v_x(t+k) - v_x^*(t+k)|^2 \right]_{\text{speed tracking}}$$

$$+ \sum_{k=0}^{p-1} \left[ |T(t+k)|^2_{Q_T} \right]_{\text{fuel consumption}}$$

$$+ \sum_{k=0}^{p} \left[ |T(t+k) - T(t+k-1)|^2_{Q_d} \right]_{\text{ride comfort}}$$

s.t. $\delta v_x(t) = 0$ (10b)

$\delta v_x(t+k) = g(\delta v_x(t+k-1), \delta T(t+k-1), 1 \leq k \leq p)$ (10c)

$v_x(t+k) = v_x(t) + \delta v_x(t+k), 1 \leq k \leq p$ (10d)

$T(t+k) = T(t) + \delta T(t+k), 0 \leq k \leq p-1$ (10e)

$T_{min} \leq T(t+k) \leq T_{max}, 0 \leq k \leq p-1$ (10f)

$\Delta_{min} \leq T(t+k) - T(t+k-1) \leq \Delta_{max}, 0 \leq k \leq p-1$ (10g)

where $Z_t = \{v_x(t+1), \ldots, v_x(t+p)\}$ and $U_t = \{T(t), \ldots, T(t+p-1)\}$ are the state and input trajectory over prediction horizon, which are to be optimized, and $v_x^*(t+k), k = 1, \ldots, p$ are the state reference for tracking. The OCP (10) has three parameters to be calibrated, namely $Q_s$, $Q_T$, and $Q_d$. Specifically,

- $Q_s$ penalizes speed tracking error;
- $Q_T$ imposes a cost for driving torque and consequently imposes a cost on energy consumption; and finally
- $Q_d$ ensures the smoothness of input trajectory by penalizing the variation of driving torque. Note that by suppressing the variation of driving torque, one also decreases the jerk (rate of change of acceleration) and improves ride comfort.

Remark 2: It is trivial to see that the above OCP can be translated into linear constrained quadratic programming (QP) problems, and can be solved in real-time by embedded devices [2], [3], [25], [26].

IV. NUMERICAL RESULTS AND DISCUSSION

This section presents numerical results concerning the MPC speed control presented above. The numerical values for various parameters are summarized in Table III. Recall from previous section that ten thousands synthetic FTP driving cycles, as plotted in Figures 3 and 4 are used for simulation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m$ [kg]</td>
<td>1500</td>
</tr>
<tr>
<td>$n_{wa}$ [-]</td>
<td>2</td>
</tr>
<tr>
<td>$R$ [m]</td>
<td>0.2159</td>
</tr>
<tr>
<td>$\mu_s$ [-]</td>
<td>1</td>
</tr>
<tr>
<td>$Q_v$ [-]</td>
<td>10</td>
</tr>
<tr>
<td>$Q_T$ [-]</td>
<td>0.001</td>
</tr>
<tr>
<td>$Q_d$ [-]</td>
<td>1</td>
</tr>
<tr>
<td>$T_{max}$ [Nm]</td>
<td>500</td>
</tr>
<tr>
<td>$\Delta_{max}$ [Nm]</td>
<td>-500</td>
</tr>
<tr>
<td>$\Delta_{min}$ [Nm]</td>
<td>-200</td>
</tr>
<tr>
<td>$T_s$ [ms]</td>
<td>100</td>
</tr>
<tr>
<td>$T_d$ [ms]</td>
<td>300</td>
</tr>
</tbody>
</table>

**Fig. 5.** Maximum speed tracking error.

**Fig. 6.** RMSE of speed tracking.
The MPC speed tracking results are summarized in Figures 5 and 6, with the red lines being the Gaussian distribution fitting. In particular, the distribution of maximum speed tracking error is skewed towards left, and its Gaussian fit has a mean of 1.402 m/s and standard deviation of 0.131 m/s. The effectiveness of MPC speed control is proven by the root-mean-square-error (RMSE), which has a mean of only 0.235 m/s and standard deviation of 0.005 m/s. Note that these results provide an intuitive view of the robustness of MPC speed tracking over FTP driving cycle. A robust MPC design can be developed to ensure the worst case performance is satisfactory; however, the tight deviation in Figure 6 suggests that such controller design may be over-restrictive.

The energy efficiency is simulated by utilizing the nonlinear battery SOC model (6). The initial SOC, denoted as \( \text{SOC}_0 \), is randomly generated among 0.2, 0.3, ..., 1. Therefore, for each \( \text{SOC}_0 \), the number of driving cycles being simulated can be different. Figure 7 plots the change of SOC after FTP cycle completion, for the cases of \( \text{SOC}_0 \) equaling 0.2, 0.5, 0.7, and 1. It can be seen that as the initial SOC decreases, the change of SOC throughout the cycle increases, indicating that the battery is operating in less efficient region as more energy are required to drive the similar cycle. This can also be demonstrated by Figures 8 and 9, which plot, for various \( \text{SOC}_0 \), the energy efficiency (mean/deviation and box plot) in terms of kWh per 100 miles. It is worth noting that when initial SOC increases, not only the power consumption drops but also its standard deviation, indicating higher robustness when operating in higher SOC region. These results also suggest to operate the EV in higher battery SOC mode, not only to preserve higher efficiency but also to maintain better resiliency.

V. CONCLUSION

This paper conducts a probabilistic analysis of energy consumption of electric vehicle, which is under speed control mode by a MPC controller. In particular, a quarter car model is used by MPC for prediction and control action optimization, and battery is modeled by nonlinear dynamic equation. The speed tracking accuracy and fuel economy of the MPC are evaluated on ten thousands synthetic FTP driving cycles. The robustness of MPC speed control is demonstrated, and insights on energy consumption and battery operations are discussed. In particular, it is more energy efficient and more robust to operate in higher SOC region. This work lay out a framework to investigate EV speed control and fuel efficiency probabilistically. Future work includes using a higher fidelity vehicle model as virtual plant. The interaction of EV with other energy consumption units, such as quadrocopter [27], water processing plant [28], and renewable energy systems [29], [30], are also future work directions.

ACKNOWLEDGMENT

This work is supported in part by faculty startup fund from School of Engineering and Computer Science at Oakland University.
University.

REFERENCES


