

Eco-driving with Learning Model Predictive Control

Yejun Kim, Samuel Tay, Jacopo Guanetti, Francesco Borrelli

Department of Mechanical Engineering

University of California, Berkeley, USA

E-mail: yk4938@berkeley.edu

Topics/model predictive control, learning, fuel efficiency

May 15, 2018

We present a predictive cruise controller which iteratively improves the fuel economy of a vehicle traveling along the same route every day. Our approach uses historical data from previous trip iterations to improve vehicle performance while guaranteeing a desired arrival time. The proposed predictive cruise controller is based on the recently developed Learning Model Predictive Control (LMPC) framework, which is extended in this paper to include time constraints. Moreover, we reformulate the modified LMPC with time constraint into a computationally tractable form. Our method is presented in detail, applied to the predictive cruise control problem, and validated through simulations.

1 INTRODUCTION

The U.S. Department of Transportation estimates that nearly 120 million Americans commute each day with an average commute time of 26 minutes [1]. In addition, 90 % of Americans drive to work with privately owned vehicles [2]. These daily commutes often follow the same route.

Advanced driving assistant systems (ADAS) aim to assist drivers in performing common driving tasks and maneuvers safely [3], [4]. With recent advancements in perception technologies and computational power, ADAS can play an important role to improve the energy efficiency of a vehicle, as well as its safety [5], [6]. Predictive cruise control (PCC) is an example of this potential [7], [8]. PCC is a longitudinal velocity control driving assistance system that uses look-ahead information about the downstream road. This information includes static information such as speed limits or road grade, and dynamic information such as traffic speed or intersection delays.

Often PCC is cast as an optimization problem. In [9], [10], optimization methods to calculate the optimal velocity trajectory for maximizing energy efficiency are presented. In [11], [12], an optimal control problem is formulated to minimize braking when the vehicle is going through a series of traffic lights. However, their approaches are limiting in the following senses. First, they require a priori knowledge about the environment. Second, the complexity of their approaches increases with the length of trip. Third, they cannot ensure global safety and time constraint guarantees unless the horizon of the optimal control problem is long enough.

In this work we tackle the problem of PCC using the Learning Model Predictive Control (LMPC) framework presented in [13]. This reference-free controller is an attractive approach because it is able to solve long horizon optimal control problems while ensuring global safety constraint guarantees. It has been successfully implemented in autonomous driving applications [14], [15]. However, the original work of LMPC does not address the problem of completing the task within a given time limit as it is usually formulated for minimum time problems, where this issue does not arise. As a result, depending on how the cost of the optimal control problem is designed, the task may result in taking more time to actually finish a task.

In order to apply the LMPC framework to PCC of a vehicle which repeatedly drives along the same route, we modify

the original LMPC so that we can enforce the task to be completed within a time limit. Moreover, in every trip, our controller can learn the static environment features such as road grade and attempts to improve its performance, which includes total fuel consumption and comfort. The contributions of this paper can be summarized as follows:

- We present the new design for LMPC with time constraint which accommodates a constraint on the total duration of the task.
- We reformulate LMPC with time constraint into a computationally tractable formation so that it can be implemented in real time.
- We design a predictive cruise control using the proposed LMPC with time constraint and show its effectiveness with simulation results.

The remainder of this paper is organized as follows. Section 2 introduces the vehicle dynamics model and a fuel estimation model, and defines the optimization problem for the predictive cruise control. Section 3 formulates the learning model predictive controller with time constraint. Section 4 provides a tractable reformulation of LMPC with time constraint for real time implementation. Section 5 demonstrates the effectiveness of our controller design as a predictive cruise controller for a repetitive trip in the Berkeley area. Section 6 concludes the work with some remarks.

2 PROBLEM FORMULATION

We aim to build the predictive cruise control of a vehicle performing repetitive trips along a fixed route, subject to a position-dependent road grade and a total completion time constraint. We refer to each successful trip of the route as an *iteration*. Each iteration has the same boundary conditions (initial and final position, speed, and force), and has to be completed within the desired time limit N_f ; $x_0 = x_s$ and $x_{N_f} = x_f$. At each iteration, the controller finds a velocity trajectory that maintains or improves the specified vehicle performance objective (such as fuel economy or comfort).

2.1 Vehicle Dynamics and Fuel Estimation

The simplified vehicle longitudinal dynamics are modeled as a connection of the first order systems with parameters identified from experiments. The states of our model include the distance travelled along the route s , the velocity v , and the force at the wheel F . The inputs to the model are F_t and F_b , which represent the wheel-level desired traction and braking forces, respectively; $u_k = [F_{t,k} \ F_{b,k}]^T$. Denoting the state at time k as $x_k = [s_k \ v_k \ F_k]^T$, the system dynamics are

$$x_{k+1} = \begin{bmatrix} s_k + t_s v_k \\ v_k + \frac{t_s}{m}(F_k + F_{R,k}) \\ (1 - \frac{t_s}{\tau})F_k + \frac{t_s}{\tau}(F_{t,k} + F_{b,k}) \end{bmatrix} \quad (1)$$

where t_s is the sampling time; m and τ are the mass and the time constant of force actuation, respectively; F_R is the resistance force which includes aerodynamic drag, rolling resistance, and gravitational force. This force can be represented as

$$F_{R,k} = mgc_r \cos(\theta) + mg \sin(\theta) + \frac{1}{2} \rho A C_d v_k^2 \quad (2)$$

where θ is the road pitch angle; g and c_r are the gravity and rolling coefficients, respectively; ρ , A , and C_d are the air density, frontal area, and drag coefficient, respectively. Moreover, the road angle is approximated using a quadratic function of the distance

$$\hat{\theta}_t(s_k) = a_{0,k} + a_{1,k}s_k + a_{2,k}s_k^2, \quad (3)$$

where $(a_{0,k}, a_{1,k}, a_{2,k})$ are parameters that are computed as follows. During each iteration of the trip, we store velocity and force values at each position along the route; we then introduce $\hat{\theta}$ to estimate the angle by inverting the dynamics (1). At each time step k of the j -th iteration, given s_k^j the parameters $(a_{0,k}^j, a_{1,k}^j, a_{2,k}^j)$ are estimated on-line solving the following least mean squares problem,

$$\arg \min_{(a_{0,k}^j, a_{1,k}^j, a_{2,k}^j)} \sum_{(p,l) \in \mathcal{G}(s_k^j)} \left\| \begin{bmatrix} 1 & s_p^l & (s_p^l)^2 \end{bmatrix} \begin{bmatrix} a_{0,k}^j \\ a_{1,k}^j \\ a_{2,k}^j \end{bmatrix} - \bar{\theta}_p^l \right\| \quad (4)$$

where $\mathcal{G}(s_k^j)$ is the set of indices and the iteration numbers with the following property

$$\mathcal{G}(s_k^j) = \{(p, l) : s_k^j \leq s_p^l \leq s_k^j + d_f\}. \quad (5)$$

where d_f is look-ahead distance which is considered a tuning parameter. A similar method is adopted for system identification of road curvature in [14].

In the remainder of this paper, the approximated vehicle dynamics model (1)-(4) is compactly rewritten as

$$x_{k+1} = f(x_k, u_k). \quad (6)$$

Also, we consider state and input constraints in the form

$$x_k \in \mathbb{X} := \{(s, v, F) : 0 \leq v \leq v_{\max}, F_{\min} \leq F \leq F_{\max}\}, \quad (7a)$$

$$u_k \in \mathbb{U} := \{(F_t, F_b) : 0 \leq F_t \leq F_{\max}, F_{\min} \leq F_b \leq 0\}. \quad (7b)$$

We are interested in designing a predictive cruise controller which tries to improve its performance as it repeats the same route. The performance objective can be fuel consumption, jerk, or travel time. In this paper we seek a better fuel economy by improving both so-called *tank-to-wheels* and *wheels-to-miles* efficiency [16]. Higher *tank-to-wheels* and *wheels-to-miles* efficiency involve improving peak management of engine, shaping

velocity, and reducing the aerodynamic and rolling losses. In order to approximate the fuel consumption, we adopted a polynomial fuel approximation method from [17] which can be written the following form,

$$f_{\text{fuel}}(v, F_t) = f_{\text{cruise}}(v) + f_{\text{accel}}(v, F_t), \quad (8)$$

where

$$f_{\text{cruise}}(v) = (b_0 v + b_1 v^2 + b_2 v^3),$$

$$f_{\text{accel}}(v, F_t) = F_t(c_0 + c_1 v + c_2 v^2),$$

and $(b_0, b_1, b_2, c_0, c_1, c_2)$ are parameters identified by least mean squares fitting of the experimental fuel rate data. The goal of our controller is to minimize this estimate of the fuel consumption, $h(x, u) = f_{\text{fuel}}(v, F_t)$.

2.2 Predictive Cruise Control Problem

For each iteration of trip, we can formulate the predictive cruise control problem as the following constrained finite horizon optimal control problem.

$$\min_{u_0^j, \dots, u_{N_f-1}^j} \sum_{k=0}^{N_f-1} h(x_k^j, u_k^j) \quad (9a)$$

subject to

$$x_0^j = x_s, \quad x_{N_f}^j = x_f \quad (9b)$$

$$x_{k+1}^j = f(x_k^j, u_k^j), \quad \forall k \in [0, \dots, N_f - 1], \quad (9c)$$

$$x_k^j \in \mathbb{X}, \quad u_k^j \in \mathbb{U}, \quad \forall k \in [0, \dots, N_f - 1], \quad (9d)$$

where j is the iteration number; (9b) and (9c) represent the boundary conditions and the vehicle dynamics, respectively; (9d) represents the state and input constraints; The stage cost, $h(\cdot)$ in (9a), represents the estimated fuel consumption.

3 LEARNING MODEL PREDICTIVE CONTROL WITH TIME CONSTRAINT

In this section, a formulation of LMPC with time constraint is proposed. Solving a finite time constrained optimal control problem such as (9) in real time can be difficult, especially when N_f is large. Therefore, we design LMPC which tries to solve the problem (9) and can be implemented in real time. In previous works, LMPC was introduced for repetitive and iterative tasks [13]. LMPC leverages past data to progressively improve performance while ensuring recursive feasibility, asymptotic stability, and non-increasing cost at every iteration. In this work, we extend the LMPC framework with a constraint on the time required to complete the task. In other words, we guarantee that each iteration or repetition does not exceed a total time limit.

Remark 1. In the original work of LMPC in [13], the optimal control problem is defined on infinite horizon. In our problem, we focus on the finite time formulation (9).

3.1 Time Sampled Safety Set

We denote the input sequence applied to the dynamics (1) and the corresponding closed loop state trajectory at j -th iteration as

$$u^j = [u_0^j, u_1^j, \dots, u_{N_f-1}^j], \quad (10a)$$

$$x^j = [x_0^j, x_1^j, \dots, x_{N_f}^j] \quad (10b)$$

where u_t^j and x_t^j are the input and the state at time t of the j -th iteration, respectively.

The main contribution of LMPC with time constraint is the modification of the safety set in [13] to the *Time Sampled Safety Set*. We define the time sampled safety set $\mathcal{SS}_{\text{time}}^j$ at j -th iteration as

$$\mathcal{SS}_{\text{time}}^j(t) = \cup_{i=1}^j \cup_{k=t}^{N_f} x_k^i \quad (11)$$

where N_f is the time limit for each iteration. The difference from the original definition of the safety set is that it only includes the states visited during the remaining time, $N_f - t$. Note that $\mathcal{SS}_{\text{time}}^j(N_f)$ is only x_f since each trip must finish within the time constraint N_f .

3.2 Preliminaries

In this section, we introduce some terminology used for the LMPC problem with time constraint.

At time t of the j -th iteration, we define the cost-to-go associated with the input sequence (10a) and the corresponding state trajectory (10b) as

$$J_{t \rightarrow N_f}^j(x_t, t) = \sum_{k=t}^{N_f} h(x_k^j, u_k^j) \quad (12)$$

where $h(\cdot)$ is the stage cost function such as the fuel estimation function (8). We have the following assumption about the stage cost $h(x, u)$.

Assumption 1. $h(\cdot, \cdot)$ is a continuous function which has the following property:

$$h(x_f, u) = 0 \text{ and } h(x, u) \geq 0 \quad \forall x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}.$$

where n_x and n_u are the dimensions of x and u , respectively.

For any $x \in \mathcal{SS}_{\text{time}}^j(t)$, we can define the minimum cost-to-go function $Q^j(x, t)$ as

$$Q^j(x, t) = \begin{cases} \min_{(i,l) \in \mathcal{F}^j(x,t)} J_{l \rightarrow N_f}^i(x, l) & \text{if } x \in \mathcal{SS}_{\text{time}}^j(t) \\ +\infty & \text{else} \end{cases} \quad (13)$$

where $\mathcal{F}^j(x, t)$ is defined as

$$\mathcal{F}^j(x, t) = \{(i, l) : i \in [0, j], l \geq t, x = x_l^i \text{ such that } x_l^i \in \mathcal{SS}_{\text{time}}^i(l)\}.$$

Note that the definition of the function $Q(\cdot, \cdot)$ is modified from the original definition in [13] because we use the new time sampled safety set $\mathcal{SS}_{\text{time}}^i(\cdot)$.

3.3 LMPC with Time Constraint Formulation

At time t of iteration $j \geq 1$, our LMPC with time constraint solves the following optimization problem:

$$\min_{u_{0|t}, \dots, u_{N-1|t}} \sum_{k=0}^{N-1} h(x_{k|t}^j, u_{k|t}^j) + Q^{j-1}(x_{N|t}, t) \quad (15a)$$

subject to

$$x_{0|t}^j = x_t^j, \quad (15b)$$

$$x_{k+1|t}^j = f(x_{k|t}^j, u_{k|t}^j), \quad \forall k \in [0, \dots, N-1], \quad (15c)$$

$$x_{k|t}^j \in \mathbb{X}, u_{k|t}^j \in \mathbb{U}, \quad \forall k \in [0, \dots, N-1], \quad (15d)$$

$$\mathbb{1}_A(k)x_{k|t}^j = \mathbb{1}_A(k)x_f, \quad \forall k \in [0, \dots, N-1], \quad (15e)$$

$$x_{N|t}^j \in \mathcal{SS}_{\text{time}}^{j-1}(t + N), \quad (15f)$$

where $\mathbb{1}_A(\cdot)$ is the indicator function of the set A defined as $A = \{t \in \mathbb{R} : t \geq N_f\}$. Constraints (15b) and (15c) represent the initial condition and vehicle dynamics, respectively; (15d) represents the state and input constraints; (15e) is the constraint which forces the system to stay at x_f at $t \geq N_f$; (15f) is the terminal constraint which imposes the system to be driven into the safe set sampled from last iteration. $Q^0(\cdot, \cdot)$ and $\mathcal{SS}_{\text{time}}^0(\cdot)$ are defined by the initial successful trip.

The resulting optimal states and inputs of (15) are denoted as

$$\mathbf{x}_t^{*,i} = [x_{0|t}^{*,i}, x_{1|t}^{*,i}, \dots, x_{N|t}^{*,i}], \quad (16a)$$

$$\mathbf{u}_t^{*,i} = [u_{0|t}^{*,i}, u_{1|t}^{*,i}, \dots, u_{N-1|t}^{*,i}]. \quad (16b)$$

Then, the first input $u_{0|t}^{*,i}$ is applied to the system during the time interval $[t, t + 1)$;

$$u_t^i = u_{0|t}^{*,i}. \quad (17)$$

At the next time step $t + 1$, a new optimal control problem in the form of (15), based on new measurements of the state, is solved over a shifted horizon, yielding a *moving* or *receding* horizon control strategy with control law.

It is noted that with the assumptions 1, LMPC with time constraint (15)-(17) is recursively feasible and the cost of each iteration monotonically decreases. The proof is similar to the original work of LMPC in [13]. The key difference between the two LMPC frameworks is that the time sampled safety set shrinks in the course of time whereas in [13], the safety set is time independent; however, this doesn't affect the proof because the time sampled safety set always includes at least one point which guarantees the existence of the feasible input and the cost decrease at the next time step; therefore, iteration cost decreases as the iteration progresses.

Remark 2. LMPC with time constraint (15)-(17) can be reformulated as a robust control and take into account of dynamic environment with minor modifications [18].

4 LMPC RELAXATION FOR PREDICTIVE CRUISE CONTROL

In this section we apply the LMPC with time constraint (15)-(17) to a predictive cruise controller subject to repetitions of the same commute with a total time limit. Because solving the optimization problem (15) in real time is computationally challenging

[14], we use an approximation method for (15): we introduce approximation functions for the time sampled safety set, $\mathcal{SS}_{\text{time}}(\cdot)$, and the terminal set, $Q(\cdot)$.

Because we restrict the vehicle velocity to be positive semi-definite, the distance travelled s always monotonically increases with time t . Therefore, we use s to shrink the time sampled set at each time. At time t of j -th iteration, we approximate $\mathcal{SS}_{\text{time}}(\cdot)$ with

$$\hat{\mathcal{S}}_{\text{time}}(t) = \left\{ (s, v, F) : s \geq s_t^{j-1}, \begin{bmatrix} v \\ F \end{bmatrix} = \Lambda \begin{bmatrix} 1 \\ s \\ s^2 \end{bmatrix} \right\} \quad (18)$$

where $\Lambda \in \mathbb{R}^{2 \times 3}$ is the solution of the following least mean square optimization problem

$$\arg \min_{\Lambda} \sum_{k \in \mathcal{T}(s_t^j)} \left\| \begin{bmatrix} v_k^{j-1} \\ F_k^{j-1} \end{bmatrix} - \Lambda \begin{bmatrix} 1 \\ s_k^{j-1} \\ (s_k^{j-1})^2 \end{bmatrix} \right\| \quad (19)$$

where $\mathcal{T}(s_t^j)$ defines the time steps in which the distance travelled during the previous $j-1$ -th iteration of trip is between the current distance and a far enough distance forward, d_f :

$$\mathcal{T}(s_t^j) = \left\{ k : s_t^j \leq s_k^{j-1} \leq s_t^j + d_f \right\}. \quad (20)$$

It's noted that \hat{d}_f is a tuning parameter decided by the control designer.

In order to approximate the cost-to-go function $Q(\cdot, \cdot)$, we introduce the third-order polynomial function $\mathcal{C}(\cdot)$

$$\mathcal{C}(s) = [1 \quad s \quad s^2 \quad s^3] \Delta \quad (21)$$

where $\Delta \in \mathbb{R}^4$ is the solution of the following least mean square optimization problem

$$\arg \min_{\Delta} \sum_{k \in \mathcal{T}(s_t^j)} \left\| J_{k \rightarrow N_f}^{j-1} - [1 \quad s_k^j \quad (s_k^j)^2 \quad (s_k^j)^3] \Delta \right\| \quad (22)$$

where $J_{k \rightarrow N_f}^{j-1}$ is the cost-to-go function defined in (12).

Finally, we approximate $Q(\cdot, \cdot)$ with

$$\hat{Q}^j(x, t) = \begin{cases} \mathcal{C}(s) & \text{if } x \in \hat{\mathcal{S}}_{\text{time}}^j(t) \\ +\infty & \text{else} \end{cases} \quad (23)$$

where $x = [s \quad v \quad F]^T$.

4.1 LMPC with Time Constraint Relaxation

With the approximation functions (18)-(23), we can reformulate LMPC with time constraint (15)-(17) as the following optimal

control problem:

$$\min_{u_{0|t}, \dots, u_{N-1|t}} \sum_{k=0}^{N-1} h(x_{k|t}^j, u_{k|t}^j) + \hat{Q}^{j-1}(x_{N|t}, t) \quad (24a)$$

subject to

$$x_{0|t}^j = x_t^j, \quad (24b)$$

$$x_{k+1|t}^j = f(x_{k|t}^j, u_{k|t}^j), \quad \forall k \in [0, \dots, N-1], \quad (24c)$$

$$x_{k|t}^j \in \mathbb{X}, u_{k|t}^j \in \mathbb{U}, \quad \forall k \in [0, \dots, N-1], \quad (24d)$$

$$\mathbb{1}_A(k)x_{k|t}^j = \mathbb{1}_A(k)x_f, \quad \forall k \in [0, \dots, N-1], \quad (24e)$$

$$x_{N|t}^j \in \hat{\mathcal{S}}_{\text{time}}^{j-1}(t+N), \quad (24f)$$

The resulting optimal states and inputs of (24) are denoted as

$$\hat{x}_t^{*,i} = [x_{0|t}^{*,i}, x_{1|t}^{*,i}, \dots, x_{N|t}^{*,i}], \quad (25a)$$

$$\hat{u}_t^{*,i} = [u_{0|t}^{*,i}, u_{1|t}^{*,i}, \dots, u_{N-1|t}^{*,i}]. \quad (25b)$$

Then, the first input $u_{0|t}^{*,i}$ is applied to the system during the time interval $[t, t+1)$;

$$\hat{u}_t^i = u_{0|t}^{*,i}. \quad (26)$$

At the next time step $t+1$, a new optimal control problem (24) with new measurements of the state, is solved over a shifted horizon.

5 SIMULATION RESULTS

In this section we validate the proposed LMPC controller (24)-(26) with simulation results. A vehicle is repeating the same trip from $s_s = 0\text{m}$ to $s_f = 5000\text{m}$ in the Berkeley hills area, depicted in Figure 1. This route is subject to position-dependent slope as shown in the top plot in Figure 2. Each trip is initialized with $x_s = [s_s, 0, 0]$ and ends with $x_f = [s_f, 0, 0]$. We initialize the trip with a simple velocity tracking controller with a constant velocity reference.

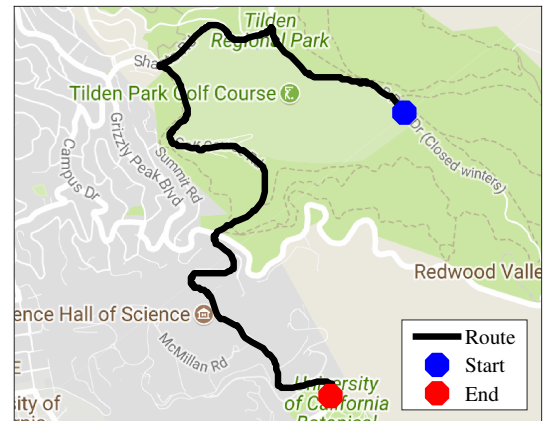


Figure 1: Fixed route from an origin A and a destination B in the Berkeley hill area

Figure 2 depicts the closed loop trajectories for the first iteration of the trip and the 8-th iteration of the same trip. In every

iteration of the trip, the arrival times does not exceed the terminal time. Also, as the iteration progresses, the velocity becomes higher in downhill sections and lower in uphill sections. This trend helps decrease the total fuel consumption of the trip as it uses the downhill regions to speed up and the uphill regions to slow down, leading to reduced acceleration. This behavior is also seen in force trajectories. Over the course of iterations, only in uphill, our controller maintains positive wheel force whereas in downhill, it tends to apply less braking (except near the end of trip where the vehicle must come to a full stop); therefore, it wastes less amount of energy.

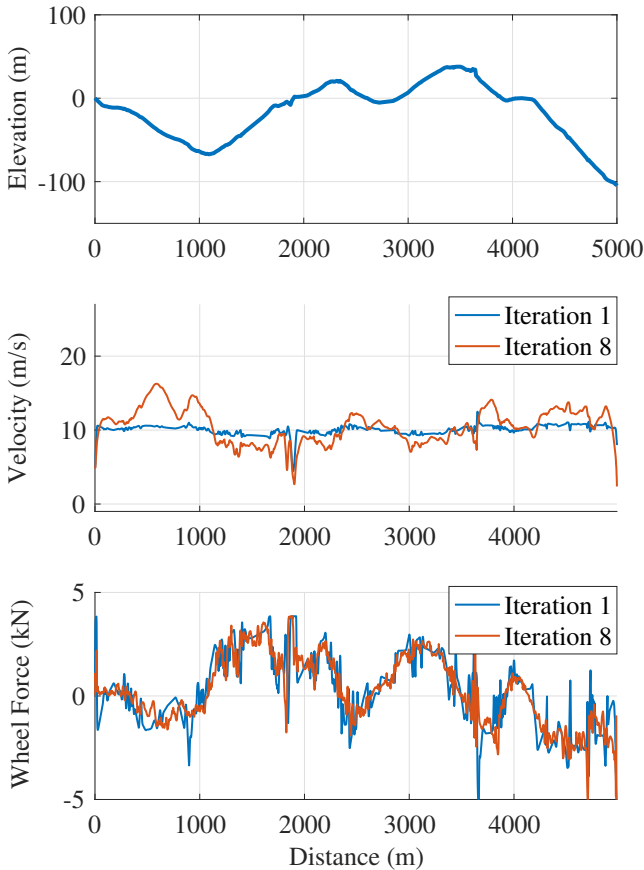


Figure 2: Plots of slope (top), closed loop velocity trajectories (middle), and closed loop wheel force trajectories (bottom)

Figure 3 depicts the normalized total cost (fuel consumption) for each iteration of the complete trips when the first trip is completed with a constant velocity tracking controller. As seen, the total cost generally decreases as the iteration number increases. There is about 4.5% reduction in fuel consumption only after 8-th trip compared to the 1st trip. It is also noted that the learning rate decreases with the iterations, as the total cost converges. This result is analogous to those in other applications of LMPC [14], [15].

6 CONCLUSION

In this paper we proposed a predictive cruise controller which improves fuel efficiency by learning from the historical trips. The key aspect is the modification of learning model predictive cruise control to guarantee completion of the task within a total time constraint, while still improving the control performance. We validated our controller with simulation results. Future work includes the experimental validation of our developed predictive cruise controller and its modification to account for disturbances from the front vehicle. Finally, we can apply the learning model

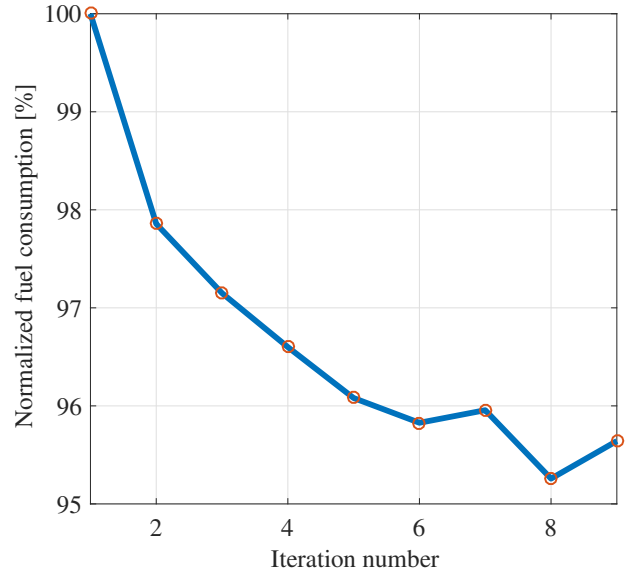


Figure 3: Plot of fuel consumption per iteration

predictive control with time constraint to other real world problems.

7 ACKNOWLEDGEMENT

The information, data, or work presented herein was funded in part by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000791. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

REFERENCES

- [1] M. L. Cohen, “Making the most of the daily commute”, *Family practice management*, vol. 14, no. 7, p. 56, 2007.
- [2] F. Dews, *Ninety percent of americans drive to work*, 2013.
- [3] P. Knoll, “Driving assistance systems”, in *Brakes, Brake Control and Driver Assistance Systems*, Springer, 2014, pp. 180–185.
- [4] A. Carvalho, S. Lefèvre, G. Schildbach, J. Kong, and F. Borrelli, “Automated driving: The role of forecasts and uncertainty a control perspective”, *European Journal of Control*, vol. 24, pp. 14–32, 2015.
- [5] V. Turri, Y. Kim, J. Guanetti, K. H. Johansson, and F. Borrelli, “A model predictive controller for non-cooperative eco-platooning”, in *American Control Conference (ACC), 2017, IEEE, 2017*, pp. 2309–2314.
- [6] J. Guanetti, Y. Kim, and F. Borrelli, “Control of connected and automated vehicles: State of the art and future challenges”, *Annual Reviews in Control*, vol. 45C, pp. 18–40, 2018.
- [7] T. Van Keulen, G. Naus, B. De Jager, R. van de Molengraft, M. Steinbuch, and E. Aneke, “Predictive cruise control in hybrid electric vehicles”, *World Electric Vehicle Journal*, vol. 3, no. 1, 2009.
- [8] N. J. Kohut, J. K. Hedrick, and F. Borrelli, “Integrating traffic data and model predictive control to improve fuel economy”, *IFAC Proceedings Volumes*, vol. 42, no. 15, pp. 155–160, 2009.
- [9] F. Lattemann, K. Neiss, S. Terwen, and T. Connolly, “The predictive cruise control—a system to reduce fuel consumption of heavy duty trucks”, SAE Technical paper, Tech. Rep., 2004.

- [10] V. Turri, B. Besselink, and K. H. Johansson, “Cooperative look-ahead control for fuel-efficient and safe heavy-duty vehicle platooning”, *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 12–28, 2017.
- [11] B. Asadi and A. Vahidi, “Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time”, *IEEE transactions on control systems technology*, vol. 19, no. 3, pp. 707–714, 2011.
- [12] C. Sun, J. Guanetti, F. Borrelli, and S. Moura, “Robust eco-driving control of autonomous vehicles connected to traffic lights”, *arXiv preprint arXiv:1802.05815*, 2018.
- [13] U. Rosolia and F. Borrelli, “Learning model predictive control for iterative tasks. a data-driven control framework.”, *IEEE Transactions on Automatic Control*, 2017.
- [14] U. Rosolia, A. Carvalho, and F. Borrelli, “Autonomous racing using learning model predictive control”, in *American Control Conference (ACC), 2017*, IEEE, 2017, pp. 5115–5120.
- [15] M. Brunner, U. Rosolia, J. Gonzales, and F. Borrelli, “Repetitive learning model predictive control: An autonomous racing example”, in *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, IEEE, 2017, pp. 2545–2550.
- [16] L. Guzzella, A. Sciarretta, *et al.*, *Vehicle propulsion systems*. Springer, 2007, vol. 1.
- [17] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, “On board eco-driving system for varying road-traffic environments using model predictive control”, in *Control Applications (CCA), 2010 IEEE International Conference on*, IEEE, 2010, pp. 1636–1641.
- [18] U. Rosolia, X. Zhang, and F. Borrelli, “Robust learning model predictive control for iterative tasks: Learning from experience”, in *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, IEEE, 2017, pp. 1157–1162.