Proceedings of the ASME 2016 Dynamic Systems and Control Conference
DSCC2016
October 12-14, 2016, Minneapolis, Minnesota, USA

DSCC2016-9780

ASSESSING FUEL ECONOMY FROM AUTOMATED DRIVING: INFLUENCE OF PREVIEW AND VELOCITY CONSTRAINTS

Niket Prakash*
Gionata Cimini
Anna G. Stefanopoulou
University of Michigan
Ann Arbor, Michigan 48109
Email: niketpr@umich.edu

Matthew J. Brusstar
US Environmental Protection Agency
Ann Arbor, Michigan 48105

ABSTRACT

Constrained optimization control techniques with preview are designed in this paper to derive optimal velocity trajectories in longitudinal vehicle following mode, while ensuring that the gap from the lead vehicle is both safe and short enough to prevent cut-ins from other lanes. The lead vehicle associated with the Federal Test Procedures (FTP) [1] is used as an example of the achieved benefits with such controlled velocity trajectories of the following vehicle. Fuel Consumption (FC) is indirectly minimized by minimizing the accelerations and decelerations as the autonomous vehicle follows the hypothetical lead. Implementing the cost function in offline Dynamic Programming (DP) with full drive cycle preview showed up to a 17% increase in Fuel Economy (FE). Real time implementation with Model Predictive Control (MPC) showed improvements in FE, proportional to the prediction horizon. Specifically, 20 s *preview MPC was able to match the DP results. A minimum of* 1.5 s *preview of the lead vehicle velocity with velocity tracking of the lead was required to obtain an increase in FE.*

The optimal velocity trajectory found from these algorithms exceeded the presently allowable error from standard drive cycles for FC testing. However, the trajectory was still safe and acceptable from the perspective of traffic flow. Based on our results, regulators need to consider relaxing the constant velocity error margins around the standard velocity trajectories dictated by the FTP to encourage FE increase in autonomous driving.

*Address all correspondence to this author.

1 INTRODUCTION

Vehicle autonomy is steadily increasing and in the coming years several manufacturers would offer vehicles with the capability for highly autonomous driving. Presently, adaptive cruise control systems lack the ability to navigate through all traffic conditions and are recommended only at highway speeds for safe operations [2]. However, autonomous vehicles at the very least would have longitudinal traffic navigation capabilities at all speeds and traffic conditions. The navigation algorithms for autonomous vehicles can be designed to improve their fuel economy (FE), as compared to the FE obtained by a human driving through the same traffic conditions.

In [3] a reduction of 0.5% – 10% in CO2 emissions is achieved using an adaptive cruise control algorithm, acting in the speed range 18 MPH-100 MPH. CO2 emissions can be further reduced by eco-driving or trace smoothing strategies, which entails reducing the total accelerations and decelerations. Indeed, [4] conservatively estimated a reduction of 33 million metric tons of CO2 annually by adopting eco-driving strategies. As an example of a trace-smoothing following algorithm, [5] employed a time-headway based linear strategy that improved the FE by increasing the time headway. In [6] a set of linear equations were used to reduce the accelerations and decelerations. Both showed significant improvements in FE. However, these results are obtained without imposing conditions on how far the follower vehicle could fall behind the lead, resulting in possible cut-ins that would change the velocity. Hence maintaining the appropriate gap between the vehicles is an important consideration that limits the scope for trace smoothing.

The authors in [7] compared a linear quadratic and a Model Predictive Control (MPC) following algorithm, showing a fuel consumption decrease of 8.8% if a 5s look-ahead capability is guaranteed. However, the authors employed time-invariant constraints on position. These could result in an awkward traffic pattern since real-world driving varies the following gap based on the speed. Moreover, they did not use standard drive cycles and hence were unable to show how the controller would perform in known and regulated conditions. Also their baseline was not a human drive cycle but a LQR derived trajectory.

In this paper, the Federal Test Protocols (FTP) and associated drive cycles are considered using the hypothetical lead vehicle. The concept of a hypothetical lead vehicle for any standard drive cycle was introduced in [1]. The idea was to find the lead vehicle velocity trajectory followed by the driver of a standard drive cycle. Autonomous vehicles could follow the same lead with various algorithms. This method would allow for simulation of actual traffic conditions associated with the Federal drive cycles and also provide a consistent comparison between how humans follow traffic and how optimal controllers would follow the same traffic conditions. Obviously any vehicle with automatic longitudinal control can employ these optimal controls.

This paper will develop optimal control algorithms that use the preview of the hypothetical lead vehicle to chart a velocity trajectory for the autonomous vehicle. The constraints on the autonomous vehicle are imposed by the position and speed of the lead. The objective is to minimize fuel consumption of the autonomous vehicle as it navigates through different traffic conditions, represented by different drive cycles. Fuel consumption is indirectly minimized by reducing energy consumed during accelerations and energy loss in decelerations. The objective thus translates into a constrained optimization problem.

To solve this optimization problem, Dynamic Programming (DP) and MPC have been used. The DP method provides the benchmark for performance improvements by solving the optimal control problem with a perfect knowledge of the future behavior. Due to the high computational and prediction requirements, DP cannot be used for real-time control of the autonomous vehicle. On the other hand MPC is becoming a standard choice when dealing with multivariable, constrained systems and can be used for real-time control [8] of the velocity trajectory, which can then be used to evaluate FC. The Advanced Light-Duty Powertrain and Hybrid Analysis Tool (ALPHA) model developed at the US Environmental Protection Agency (EPA) was used to evaluate fuel consumption [9] for a 2013 Ford Escape with a 1.6L EcoBoost® engine [10].

The remainder of this paper is organized as follows. Section 2 presents the model of the vehicle and the problem statement. Section 3 shows the application and results of DP for the following problem. Section 4 presents three different formulations for MPC implementation and compares them. Section 5 concludes the paper with a discussion on the results obtained.

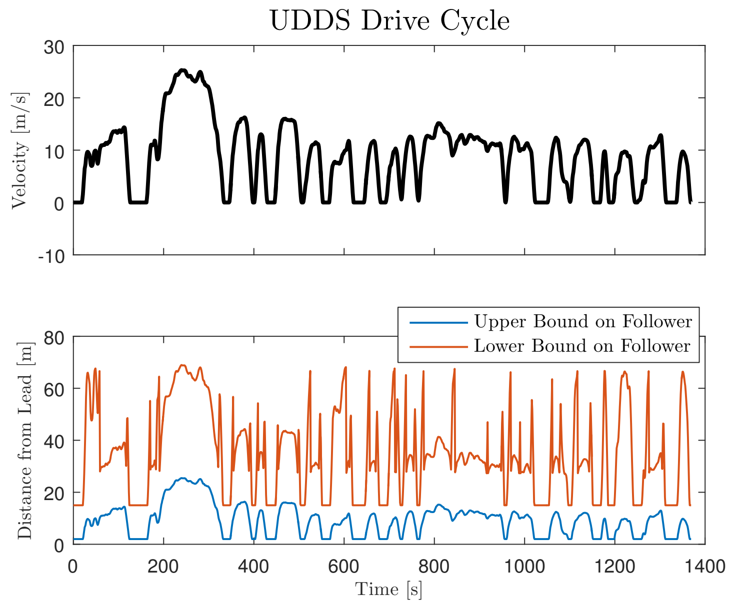

FIGURE 1. Position constraints to be applied on the following autonomous vehicle based on the position and velocity of the hypothetical lead

2 Model Description

In this paper, the dynamics of the autonomous vehicle are described by a simple point mass Linear Time-Invariant (LTI) model, with position (p) and velocity (v) as states and acceleration (a) as the only input, namely

$$x_{k+1} = \underbrace{\begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}}_{A} x_k + \underbrace{\begin{bmatrix} 0.5T_s^2 \\ T_s \end{bmatrix}}_{B} u_k \tag{1}$$

with $T_s = 0.1$s the sampling time, $x = [x_1\ x_2] = [p\ v]' \in \mathbb{R}^2$ completely measurable and $u = a \in \mathbb{R}$. The simple dynamics allow for fast online controller implementation. The velocity state v can be used offline in the ALPHA model to estimate the Fuel Consumption and hence the Fuel Economy of the vehicle. Define $\mathbb{U}$ and $\mathbb{X}_k$ as polyhedral sets of constraints on inputs and states respectively, such that

$$\mathbb{U} = \{u \in \mathbb{R} \,|\, u^{\min} \leq u \leq u^{\max}\} \tag{2a}$$

$$\mathbb{X}_k = \{x \in \mathbb{R}^2 \,|\, [x_{1,k}^{\min}\ x_2^{\min}]' \leq x_k \leq [x_{1,k}^{\max}\ x_2^{\max}]'\}. \tag{2b}$$

The acceleration and deceleration constraints are time-invariant. For this paper they have been derived from the standard drive cycles to be $u^{\min} \equiv -6$ m/s^2, and $u^{\max} \equiv 6$ m/s^2. The same is true for the state x_2 or velocity v where $v^{\min} \equiv 0$ m/s, and $v^{\max} \equiv 40$ m/s. Constraints on the state x_1 or position p however, are time varying as the position of the autonomous vehicle is determined by the position and velocity of the hypothetical lead and hence a speed dependent gap.

The gap between the hypothetical lead and the autonomous vehicle is constrained by an upper and a lower bound. The upper bound is a safety limit and is the closest that the follower vehicle can follow the lead vehicle. This is derived from being 1 car length behind the lead vehicle for every 10 MPH. The lower bound is derived from assuming a distance that would prevent safe cut-ins from adjacent lanes. This is kept at 4 ft/MPH or 2.7 m/m/s. The constraints are further relaxed at low speeds of less than 20 MPH to 10 ft/MPH or 2.7 m/m/s. Indeed, at such low speeds cut-ins are not expected and a longer gap reduces frequent starts and stops, thus delivering better FE. Since the position constraints are dependent on the lead vehicle's states at that instant, these constraints are time varying. Fig. 1 shows the upper and lower bounds on position at different velocities of the lead vehicle. The constraints on position and speed are selected according to

$$x_{1,k}^{\min} = x_L + v_l\, L/10 \tag{3a}$$

$$x_{1,k}^{\max} = x_L + \begin{cases} v_l\, d_{\max} & \text{if } v_l < 20\text{MPH} \\ v_l\, d_{\min} & \text{otherwise} \end{cases} \tag{3b}$$

$$x_2^{\min} = 0 \tag{3c}$$

$$x_2^{\max} = 40 \tag{3d}$$

where x_L is the position of the lead vehicle, v_l is the velocity of the lead vehicle, L is 1 car length, $d_{\max}$ is 10 ft and $d_{\min}$ is 4 ft. The time-invariant limits on the speed are reasonable on almost all U.S.A. roads [11].

The model does not consider the engine, vehicle or powertrain dynamics and is a general formulation applicable to all vehicle types. The resulting optimized velocity trajectory is applied as an input to the ALPHA model [9] of a particular vehicle for offline computation of fuel economy. The results for FE shown in this paper are specific to the 2013 Ford Escape with a 1.6L EcoBoost® engine [10]. The absolute values of fuel economy would change for different vehicles with other powertrain and engine configurations.

3 Dynamic Programming

The objective of this paper is to indirectly minimize the FE of the autonomous vehicle by minimizing its accelerations and decelerations. The optimal velocity trajectory has to be computed while keeping the vehicle within the position constraints governed by the lead vehicle. Assuming that all future references, constraints, and disturbances are perfectly known, dynamic programming (DP) can be used to find a non-causal, global-optimal input (acceleration) sequence that minimizes a defined cost function [12]. In this work the input or the acceleration itself has to be minimized. The DP methodology would provide the minimum input, acceleration and deceleration necessary to ensure that the vehicle adheres to its position constraints.

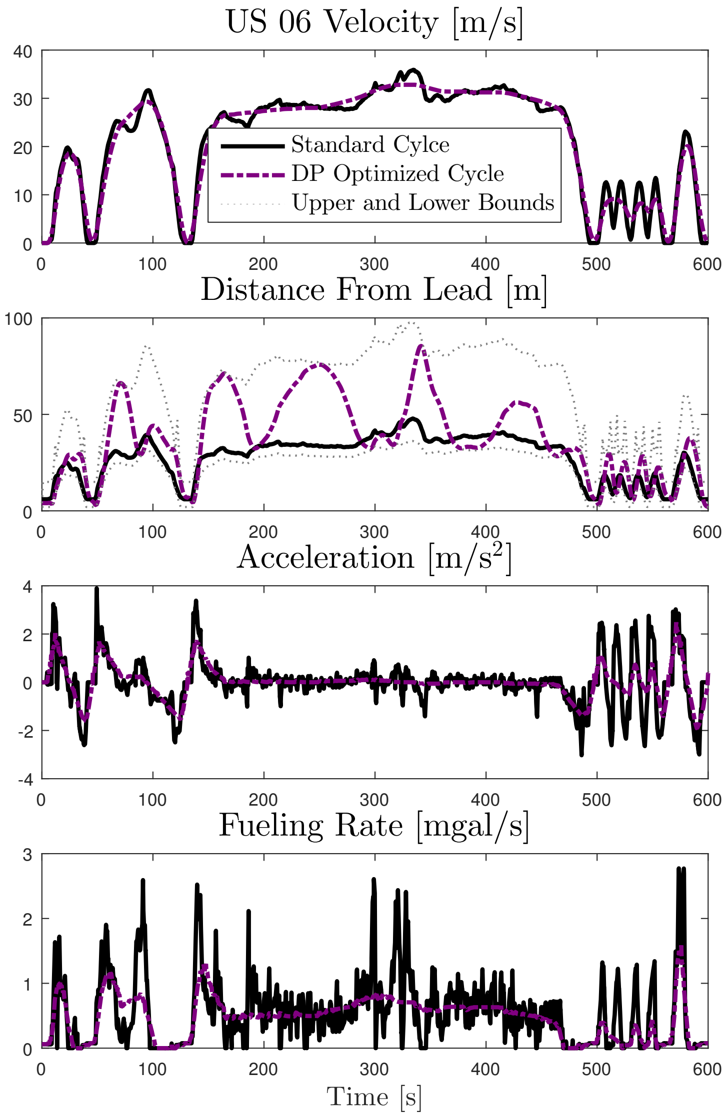

FIGURE 2. Comparison of standard US06 drive cycle with the DP optimized velocity profile. The fueling rates are found from ALPHA model simulations [9].

Despite its limitation as an offline technique, DP results serve as an upper bound on performance for the design of real-

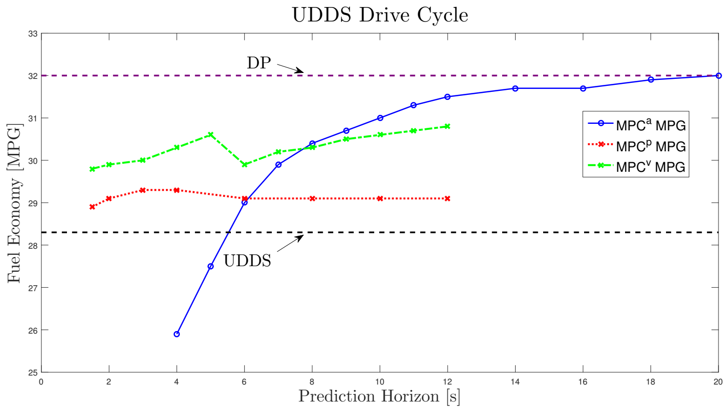

FIGURE 3. Fuel economy using various horizons and cost functions in the MPC formulation

time control strategies. The formulation is as follows

$$\min_{u} \sum_{k=1}^{N_f} \|u_k\|_2^2 \tag{4a}$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k \tag{4b}$$

$$u_k \in \mathbb{U},\ x_k \in \mathbb{X}_k \tag{4c}$$

where N_f is the final time-step, A, B are defined as in (1) and $\mathbb{U}$, $\mathbb{X}_k$ are defined as in (2), constraints on $\mathbb{X}_k$ are defined as in (3).

For this paper the generic dynamic programming MATLAB® function in [13] has been used. It allows for constrained optimal control problems, such as the one in (4). In this work, the states and the inputs are discretized into 201 grid points. The same discretization is applied across all drive cycles. All fuel economies, including those of the standard drive cycles are calculated by applying the velocity trajectory as an input to the ALPHA model of the 2013 Ford Escape with a 1.6 L EcoBoost® engine. The resulting optimal velocity trajectory showed significant improvements in FE for four US Environmental Protection Agency (EPA) drive cycles. The highest increase was seen in US06 at 16.7% over the standard US06 drive cycle [14]. Fig. 2 shows velocity trajectory of the DP optimal trace as compared to the standard EPA defined trace. By utilizing the entire gap between the upper and lower bounds, DP is able to find the velocity trajectory with minimal accelerations and decelerations while remaining close to the lead vehicle. The absence of these acceleration spikes reduces fueling rates and leads to the significant improvements in FE.

4 Model predictive control

Although DP provides the optimal solution, several reasons prevent its use for online control such as the high computational

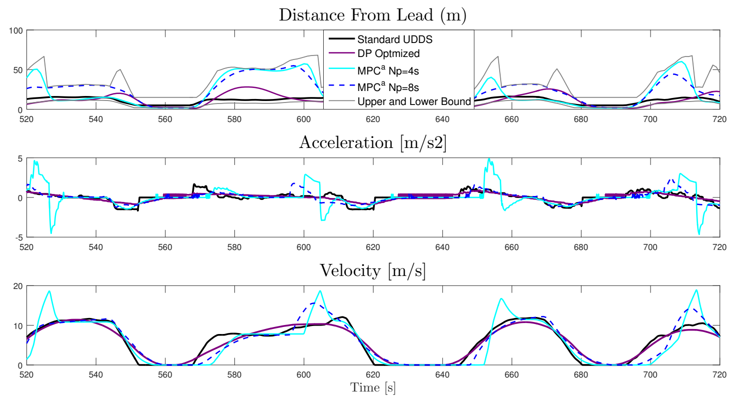

FIGURE 4. Comparison of standard UDDS drive cycle with DP and MPC^a velocity traces. Clearly for MPC^a with $N_p = 4s$, at Time = 530, 605, 655 and 715 s the position is too close to the lower bound and the MPC^a controller applies significant acceleration to stay within the bounds. These accelerations are less pronounced for the $N_p = 8$s case and absent in the DP case.

burden and the complete knowledge about the future behavior of the system. The MPC methodology is becoming the standard technology to handle fast-sampled multivariable processes, especially in automotive, aerospace and power electronics control [8, 15–17]. This methodology solves a constrained, finite-horizon, optimal control problem, and by following the receding horizon policy it applies only the current inputs to the system. At each sampling time the procedure is repeated and a new open-loop optimal control problem is solved with a one-step shifted horizon. Even if MPC drastically reduces the computational complexity with respect to DP, the time required to solve the optimization problem is still considerable, especially in high frequency sampled systems [18]. However, recent advances in convex and embedded optimization are enabling fast online MPC implementations [19–21]. When dealing with LTI models, quadratic cost function and affine constraints, MPC simplifies to solving the following optimization problem at each sampling instant

$$\min_{u} \sum_{i=1}^{\tilde{N}_p} \|W_x(x_{k+i|k} - r_{k+i|k})\|_2^2 + \|W_u u_{k+i|k}\|_2^2 \tag{5a}$$

$$\text{s.t. } x_{k+i+1|k} = Ax_{k+i|k} + Bu_{k+i|k} \tag{5b}$$

$$x_{k|k} = x_k \tag{5c}$$

$$u_{k+i|k} \in \mathbb{U},\ x_{k+i|k} \in \mathbb{X}_{k+i|k} \tag{5d}$$

where $\tilde{N}_p$ is the prediction horizon expressed in sampling instants, W_x and W_u are square, diagonal, weight matrices, $x_{k+i|k}$ denotes the prediction of the variable x at time $k+i$ based on the information available at time k, and $r_{k+i|k}$ denotes the prediction of the reference r to be tracked at time $k+i$, x_k is the current state. The influence of the prediction horizon, $N_p = \tilde{N}_p.T_s$ expressed in seconds, to the optimal velocity trajectory will be investigated along with three different MPC formulations. All three formulations share the same linear prediction model and the same constraints, as defined in Eqs. (1) and (2), respectively. They differ from each other for the cost function, and we will refer to them as

1. MPC^a: penalty on acceleration;
2. MPC^p: penalty on acceleration and position tracking;
3. MPC^v: penalty on acceleration and velocity tracking.

MPC^a can be considered as a reduced horizon version of the optimal control problem (4), and its cost function is defined by the following weights:

$$\mathrm{MPC}^a \triangleq \begin{cases} W_x^a = \mathrm{diag}\,[0\ 0] \\ W_u^a = w_u^a = 1 \end{cases} \tag{6}$$

The length of the prediction horizon plays an important role in this control problem as not only the dynamics of the model are predicted but also the constraints on the states and the eventual reference trajectory. This means that high prediction horizons are not only computationally demanding, but can even be infeasible depending on the preview, look-ahead, and overall connectivity responsible for accurately predicting the future behavior of the system. Indeed, in [22], it was shown that while velocity trajectories for short prediction horizons can be predicted well for $N_p = 1.5\,\mathrm{s}$, a good accuracy for longer horizon is not very probable.

For autonomous driving it can be assumed that these predictions come from different methods such as vehicle to vehicle communication or from a traffic monitoring system. Given different prediction horizons, this work evaluates the potential increase in fuel economy. It must be stressed that for this work, perfect prediction of constraints along the prediction horizon is assumed for all cases.

Fig. 3 shows that MPC^a gives satisfactory results only for $N_p \geq 6s$. This is a significantly long prediction horizon and would not be feasible. For shorter prediction horizons $N_s < 5\,\mathrm{s}$, MPC^a gives an FE that is even worse with respect to the standard cycle. The reason is clarified in Fig. 4, which shows that without enough prediction of the position constrains, MPC^a keeps the vehicle too close to the lower bound. This leads to acceleration spikes to keep the vehicle within the bounds, decreasing the FE. Fig. 3 also shows that, as expected, the performance of MPC^a equals the one obtained with DP after a certain prediction horizon, i.e. 20s, as the two formulations are equivalent but for the prediction horizon. The RMS error between the velocity trajectory in DP and MPC^a vary as $2.24\,\mathrm{m/s}$ for $N_p = 4\,\mathrm{s}$, $1.26\,\mathrm{m/s}$ for $N_p = 8\,\mathrm{s}$ and $0.58\,\mathrm{m/s}$ for $N_p = 20\,\mathrm{s}$ thus showing how the MPC^a results approach the DP ones.

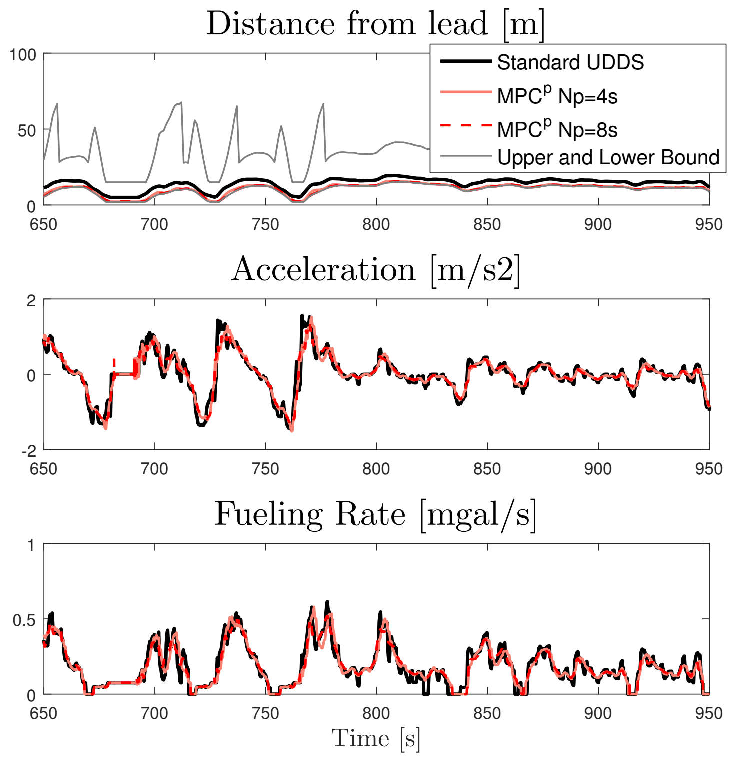

FIGURE 5. Comparison of standard UDDS drive cycle with two MPC filtered velocity traces where the MPC has position tracking in the cost function. Clearly with increased prediction horizon, the position tracking improves. However, the reduction in acceleration between the two MPC cases is small leading to very slight improvement in FE

4.1 Model predictive control with tracking penalty

From the results of MPS^a, the vehicle had to be prevented from falling too close to the lower bound on position. Since the state vector is completely measurable, the deviation from the position or velocity reference can be penalized to ensure that the vehicle is within acceptable bounds. The MPC implementations, MPC^p and MPC^v introduce two different tracking penalties in the cost function. It is straightforward to verify that penalizing the velocity or position tracking error worsens the FE for longer prediction horizons, where MPC^a works well. However, as shown in Fig. 3, the performance improvement for shorter prediction horizons is significant.

In MPC^p, the position state is forced to track the upper bound of the position constraints. This ensures that the position of the vehicle is far from the lower bound. However, position tracking limits the ability of the controller to utilize the entire gap between the bounds and minimize acceleration. An alternative is MPC^v, where the controller is forced to track the velocity of the lead vehicle. Tracking the lead velocity would ensure that for short prediction horizons, the FE at least matches that of the standard cycle. Additionally, acceleration optimization would increase FE to a value beyond that of the standard cycle. The tradeoff between acceleration optimization and velocity tracking by employing appropriate weights for different prediction horizons are discussed below.

In MPC^p the weights on the cost function are assumed to be

$$\text{MPC}^p \triangleq \begin{cases} W_x^p = \text{diag}\left[w_x^p \ 0\right] \\ W_u^p = w_u^p \end{cases} \tag{7}$$

whereas the weights for MPC^v are such that

$$\text{MPC}^v \triangleq \begin{cases} W_x^v = \text{diag}\left[0 \ w_x^v\right] \\ W_u^v = w_u^v \end{cases} \tag{8}$$

Since the main objective is minimization of acceleration, the weights were chosen such that $w_u^p > w_x^p$ and $w_u^v > w_x^v$. This rule ensures that the penalty on tracking is less than the penalty on high acceleration. In the proposed results, the weights are tuned for each prediction horizon. For longer prediction horizons the tracking accuracy can be reduced, thus giving freedom for higher acceleration minimization. As an example, for the case $N_p = 1.5s$ the weights are $w_u^p = w_u^a = 1$, $w_x^p = 0.8$ and $w_x^v = 0.2$.

Fig. 5 shows MPC^p results for two prediction horizons, i.e. 4s and 8s. The upper bound is closely tracked by both cases and the tracking performance increases slightly as prediction horizon increases. The FE improves over the standard cycle even for short prediction horizons. The close position tracking, however, reduces the scope for optimization and hence the FE for both cases is nearly the same. Fig. 3 shows that for $N_p = 8$ s, the MPC^a has a much better FE than MPC^p.

It was found that MPC^v showed an even better FE than MPC^p for short prediction horizons. In Fig. 6 it can be seen that while staying within the position constraints, MPC^v had lower acceleration and engine torque demands than MPC^a and MPC^p. Clearly the DP case with the entire drive cycle preview is the best for FE. But, for $N_p = 1.5$ s MPC^v shows the best results compared to other possible real-time solutions. MPC^v is therefore selected as our real-time following control implementation.

5 Discussion

The objective of this paper was to find a controller that generates an optimal velocity trajectory such that the acceleration is minimized. Table 1 shows the improvements in FE for both the DP and MPC^v cases. Significant improvements were shown

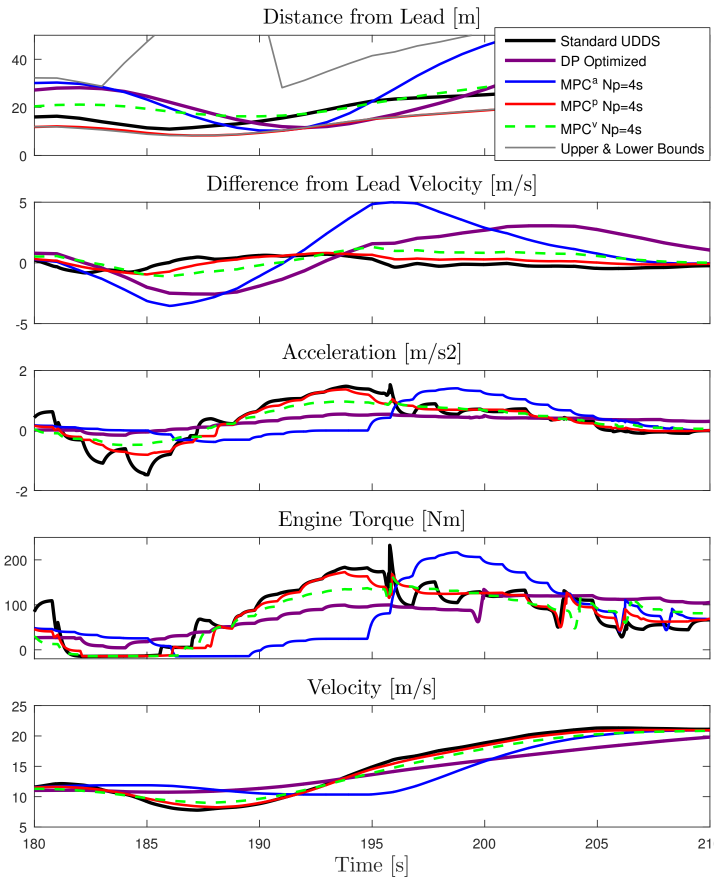

FIGURE 6. Comparison of different drive cycles derived from following the hypothetical lead using DP and MPC algorithms with different cost functions at Np=4s. The acceleration profile of DP is the best followed by MPC with velocity tracking, MPC with position tracking, the standard cycle and finally MPC without reference. This is reflected in the the engine torque and correspondingly the fueling rate.

in FE for the Dynamic Programming case between 13.1% and 16.7% indicating a potential for high gain in FE for autonomous vehicles that enable real time optimization. Real-time implementation with MPC^v controllers shows improvements between 5.3% and 11.8%, for the shortest prediction horizon of 1.5 s. While obviously not performing as well as DP, MPC^v is able to improve upon the baseline. For very long prediction horizons of $N_p \geq 20$ s, MPC^a is able to match the DP results for all the given cycles simulated in ALPHA for the selected vehicle.

The model used in the optimal controller was deliberately kept as a simple point mass LTI acceleration model to generate the optimal velocity trajectory without considering the vehicle power train or engine dynamics. This was achieved by reducing the total accelerations and decelerations while maintaining an acceptable distance from the lead vehicle based on acceptable traffic patterns. This work shows that in meeting the simple ac-

TABLE 1. MPG improvements with DP and MPC

Cycle Name	Standard Cycle [MPG]	DP [MPG | Imp.]		MPC^v (N_p=1.5s) [MPG | Imp.]	
UDDS	28.3	32.0	13.1%	29.8	5.3%
US06	24.5	28.6	16.7%	27.4	11.8%
LA92	26.0	30.7	15.3%	28.2	8.5%
SC03	27.7	31.8	14.8%	29.2	5.4%

celeration objectives, FE can also be increased while following a hypothetical lead vehicle with a velocity profile associated with the federal test procedures [1]. An autonomous vehicle could have this FE objective along with other safety and traffic flow objectives. Tailoring the FE optimization by minimizing fuel consumption instead of the generic acceleration could result in even more benefits, given that not all accelerations are equally fuel consuming. This optimization will require engine and transmission data or models.

6 Conclusion

This paper shows the possibilities for improvement on present vehicle technologies through autonomous driving. Offline DP results achieve up to 17% improvement in FE. Simulation of various MPC formulations with a reasonable horizon was able to achieve 12% improvement in FE. The velocity trajectories that achieved these improvements in autonomous vehicles were significantly different from the velocity profile of the standard cycles. The velocity difference was more than what is currently allowed by regulations. In light of this work, regulators need to reconsider the standard FE testing procedure that imposes a tight band around the velocity trace to encourage use of algorithms that increase FE, acceptable from both traffic and safety perspectives.

REFERENCES

[1] Prakash, N., Stefanopoulou, A. G., Moskalik, A., and Brusstar, M. J., 2016. "The hypothetical lead (HL) vehicle trace: A new method for evaluating fuel consumption in automated driving". In American Controls Conference.

[2] Kesting, A., Treiber, M., Schonhof, M., and Helbing, D., 2008. "Adaptive cruise control design for active congestion avoidance". *Transportation Research Part C,* ***16***(1), pp. 668–683.

[3] Klunder, G. A., Malone, K., Mak, J., Wilmink, I. R., Schirokoff, A., Sihvola, N., and C., H., 2009. "Impact of information and communication technologies on energy efficiency in road transport-final report".

[4] Barkenbus, J. N., 2010. "Eco-driving: An overlooked climate change initiative". *Transportation Research Part C: Emerging Technologies,* ***38***(2), pp. 762–769.

[5] Mersky, A. C., and Samaras, C., 2015. "How will autonomous vehicle technology affect fuel economy ratings?". *Poster presented at Automated Vehicle Symposium, Ypsilanti*.

[6] Manzie, C., Watson, H., and Halgamuge, S., 2007. "Fuel economy improvements for urban driving: Hybrid vs. intelligent vehicles". *Transportation Research Part C: Emerging Technologies,* ***15***(1), pp. 1–16.

[7] Shengbo, K., Wang, J. Zhang, L., Lian, X., Ukawa, H., and Bai, D., 2008. "MPC based vehicular following control considering both fuel economy and tracking capability". *Vehicle Power and Propulsion Conference, IEEE*, pp. 1–6.

[8] Mayne, D. Q., 2014. "Model predictive control: Recent developments and future promise". *Automatica,* ***50***(12), pp. 2967 – 2986.

[9] Byungho, L., Lee, S., Cherry, J., Neam, A., Sanchez, J., and Nam, E. "Development of advanced light-duty powertrain and hybrid analysis tool". *SAE Technical Paper*(2013-01-0808).

[10] Stuhldreher, M., Schenk, C., Brakora, J., Hawkins, D., A., M., and P., D. "Downsized boosted engine benchmarking and results". *SAE Technical Paper*(2015-01-1266).

[11] NHTSA, 2010. *Summary of State Speed Laws*, 11 ed.

[12] Bertsekas, D., 2005. *Dynamic programming and optimal control*. Belmont, Massachusetts: Athena Scientific.

[13] Sundstrom, O., and Guzzella, L., 2009. "A generic dynamic programming matlab function". In Control Applications, (CCA) & Intelligent Control, (ISIC), pp. 1625–1630.

[14] U.S. National Archives and Records Administration. 2016. Code of Federal Regulations. Title 40. Protection of Environment. Section 86.158. Supplemental Federal Test Procedures; overview.

[15] Del Re, L., Allgöwer, F., Glielmo, L., Guardiola, C., and Kolmanovsky, I., 2010. *Automotive Model Predictive Control: Models, Methods and Applications*. Lecture Notes in Control and Information Sciences. Springer London.

[16] Hartley, E. N., and Maciejowski, J. M., 2015. "Field programmable gate array based predictive control system for spacecraft rendezvous in elliptical orbits". *Optimal Control Applications and Methods,* ***35***(7), pp. 585–607.

[17] Kouro, S., Perez, M., Rodriguez, J., Llor, A., and Young, H., 2015. "Model predictive control: MPC's role in the evolution of power electronics". *Industrial Electronics Magazine, IEEE,* ***9***(4), Dec, pp. 8–21.

[18] Necoara, I., 2015. "Computational complexity certification for dual gradient method: Application to embedded MPC". *Systems & Control Letters,* ***81***, pp. 49 – 56.

[19] Cimini, G., Bernardini, D., Bemporad, A., and Levijoki, S., 2015. "Online model predictive torque control for per-

manent magnet synchronous motors". In Industrial Technology (ICIT), 2015 IEEE International Conference on, pp. 2308–2313.
[20] Bemporad, A., 2015. "A quadratic programming algorithm based on nonnegative least squares with applications to embedded model predictive control". *IEEE Transactions on Automatic Control,* ***PP***(99), pp. 1–1.
[21] Ferreau, H. J., Bock, H. G., and Diehl, M., 2008. "An online active set strategy to overcome the limitations of explicit MPC". *International Journal of Robust and Nonlinear Control,* ***18***(8), pp. 816–830.
[22] Nüesch, S., Sterniak, J., Jiang, L., and Stefanopoulou, A. G., 2015. "On beneficial mode switch decisions based on shortterm engine load prediction". *IFAC E-COSM*.