Real-time Driving Mode Advice for Eco-driving using MPC

Yutao Chen [*] **Mircea Lazar** [**]

[*] *College of Electrical Engineering and Automation, Fuzhou University, China. (e-mail: ytchen_fzu@163.com).*
[**] *Department of Electrical Engineering, Eindhoven University of Technology, the Netherlands. (e-mail: m.lazar@tue.nl).*

Abstract: This paper proposes an on-line advice eco-driving assistance system (EDAS) for providing the optimal velocity profile to improve fuel economy. The EDAS employs a driver-in-the-loop (DIL) framework, where an adviser is designed to provide high-level driving mode suggestions while the low-level control commands such as throttle and brake, are left to the driver to implement. A simplified dynamic model is developed in the adviser excluding continuous-time control variables such as the engine torque and engine brake torque. The adviser employs an event-triggered model predictive control (MPC) algorithm to provide suggestions in real-time using predictive road and traffic information. On-line computational cost for the MPC has been significantly reduced using an efficient mixed-integer optimal control (MIOC) algorithm. To demonstrate the efficiency and effectiveness of the proposed EDAS, a numerical study and a simulation using measured data from a real-life driving test is conducted. Comparisons are made between the proposed EDAS and an eco-driving controller considering both high and low level control inputs without a driver.

Keywords: Eco-driving, driving assistance system, velocity profile optimization, model predictive control

1. INTRODUCTION

Eco-driving is a driving strategy to reduce fuel consumption and emissions and has been studied and applied extensively in the past decade (Huang et al., 2018). Eco-driving is a complex task and traditionally relies on driver experience (Brackstone and McDonald, 1999).

A key factor of eco-driving is to provide an optimal velocity profile that minimizes fuel consumption on a road segment. This can be formulated as an optimal control problem (OCP), where control inputs including the engine torque and gear positions are optimized. (Hellström et al., 2009; Kirches, 2011; Saerens, 2012; Sciarretta et al., 2015; Ozatay et al., 2017). To solve the OCP, different algorithms have been employed including dynamic programming (DP) (Hellström et al., 2009), analytic solution methods (Sciarretta et al., 2015), Pontryagin maximum principle (Ozatay et al., 2017) and nonlinear optimization (Padilla et al., 2018). However, such an all-in-one optimization faces safety and implementation difficulties. First, it is not safe to implement the low-level solutions such as the brake torque by an automatic controller. Second, there are additional costs to install and control actuators to implement the driving commands. Therefore, this strategy is often used in driving scenarios such as adaptive/predictive cruising control (A/PCC), where minimum operation is required and automated driving is currently possible (Sciarretta et al., 2015).

On the other hand, eco-driving assistance systems (EDAS) can be immediately installed in real vehicles. The reason is that it is safer and easier to have a driver-in-the-loop (DIL) control framework. Traditionally EDAS were designed for driving style training including theoretical lessons and practice with an observer (Beloufa et al., 2017). More recently, in-vehicle assistance systems have been adopted to give real-time feedback and guiding to human drivers (Kamal et al., 2010; Hibberd et al., 2015). On-line assessment systems can provide real-time eco-driving scores that reflect the driver performance. Active feedback systems on accelerator and brake pedal can help drivers to choose more economic driving operations (Yin et al., 2013; Thijssen et al., 2014). In addition, on-line advice systems have been exploited to help drivers track the optimal velocity profile obtained by solving an OCP with predictive information (Cheng et al., 2013; Lin et al., 2014). DP and analytic solution methods are usually employed to solve the optimization problem.

In this work, we propose an in-vehicle on-line advice EDAS for velocity, gear and driving mode profile optimization. We limit ourselves to an optimal deceleration scenario where the vehicle has to decelerate to a certain speed within a given distance. Extensions to other scenarios such as optimal acceleration, stop-and-go and eco-cruising are also possible (Saerens, 2012). An event-triggered model predictive control (MPC) scheme is developed to repeatedly identify deceleration events and to provide driver suggestions in real-time. As a first step study, this work assumes an *ideal* driver who can perfectly follow the suggestions. We evaluate the fuel economy of the proposed control strategy in a numerical study and in a simulation using measured experimental data from a test drive and compare it with the all-in-one optimization method, which optimizes high and low level control variables at the same time. The key contributions of the paper are the following:

- A MPC based adviser is designed to provide high-level suggestions on how and when to choose an appropriate driving mode based on predictive road and traffic information. Low-level commands such as engine torque and brake torque are left to the driver to implement in order to track the optimal suggestions. Safety is thus reasonable.
- An efficient mixed-integer optimal control (MIOC) algorithm is employed to solve the optimization problem on-line. Model simplification and gear pruning heuristics are proposed to make the algorithm run in real-time.
- A detailed comparison between the proposed adviser and a traditional all-in-one optimization-based controller is given. We show that the proposed adviser can achieve comparable fuel savings with significantly less computational cost.

2. DYNAMIC MODEL FOR OPTIMAL CONTROL BASED ECO-DRIVING

We review a relatively complete dynamic model of internal combustion engine vehicles (Kirches, 2011; Saerens, 2012; Sciarretta et al., 2015). The model consists of two continuous-time control inputs: the engine torque T_e and the engine brake torque T_{eb}. The model has an additional integer-valued control variable: the gear position y. In this work, model parameters are measured from a DAF heavy duty truck.

2.1 Vehicle dynamic model

The longitudinal dynamics of the vehicle is given by

$$\frac{\mathrm{d}v}{\mathrm{d}s} = \frac{1}{mv}\left(\frac{i_r}{r_w}(T_{pr} - T_b) - T_r\right), \tag{1}$$

where v is the vehicle velocity and m is the total mass. The model is within the spacial domain since the independent variable is the arc length s. There are three kind of forces: the propulsion torque T_{pr}, the brake torque T_b and the resistance torque T_r. The rear axle transmission ratio is i_r and the wheel radius is r_w and regarded as static.

In detail, the three forces can be computed by

$$T_{pr} = i_t(y)\eta(y)T_e, \tag{2}$$

$$T_b = i_t(y)T_{eb} + i_t(y)T_{fric}, \tag{3}$$

$$T_r = T_{roll} + T_{aero} + T_{slope}, \tag{4}$$

where $i_t(y)$ and $\eta(y)$ are the transmission ratio and gear efficiency of the corresponding gear position y respectively and ω_e is the engine speed. The forces are the engine friction force T_{fric}, rolling resistance force T_{roll}, aerodynamic drag force T_{aero} and the gravitational force T_{slope} due to road slope. The resistance forces are functions of the velocity v and the slope angle α:

$$T_{roll} = mgC_r\cos\alpha, \tag{5}$$

$$T_{aero} = \frac{1}{2}\rho C_d A v^2, \tag{6}$$

$$T_{slope} = mg\sin\alpha, \tag{7}$$

where ρ is the air density, C_d the coefficient of air drag, A the frontal area of the vehicle, C_r the coefficient of rolling resistance and g the gravitational acceleration.

The maximum engine torque $T_{e,max}$, maximum brake torque $T_{eb,max}$ and friction torque T_{fric} are functions of the engine speed ω_e, which depends on the vehicle velocity v and the gear position y:

$$\omega_e(v, y) = \frac{60 v i_t(y) i_r}{2\pi r_w}, \quad \text{if } y > 0 \tag{8}$$

where $y = \{0, 1, \ldots, 12\}$ and ω_e is in revolutions per minute (RPM). In this work, we ignore clutch slipping. When $y = 0$, the gear is in neutral with a gear ratio $i_t(0) = 0$ and the engine maintains an idle speed at $\omega_{e,min} = 550$ RPM.

Note that the disk brake is not considered in this work since it can be chosen by the driver at any time for safety considerations. In safe driving scenarios, it is never an optimal driving mode for eco-driving.

2.2 Fuel consumption model

The fuel consumption has the following dynamics in spacial domain:

$$\frac{\mathrm{d}m_f}{\mathrm{d}s} = \frac{1}{v}\dot{m}_f, \tag{9}$$

where $\dot{m}_f[gram/s]$ is the fuel consumption rate. To approximate $\dot{m}_f$, a number of models have been compared in (Saerens, 2012). We consider the second order polynomial model given by

$$\dot{m}_f(\omega_e, T_e) = \beta_1 + \beta_2\omega_e + \beta_3\omega_e^2 + \beta_4\omega_e T_e + \beta_5 T_e + \beta_6 T_e^2, \tag{10}$$

where β_i, $i = 1, \ldots, 6$ are polynomial coefficients and can be obtained by fitting the model with measurement data. The heavy duty DAF truck adopted in this work has a fuel consumption map shown in Fig. 1.

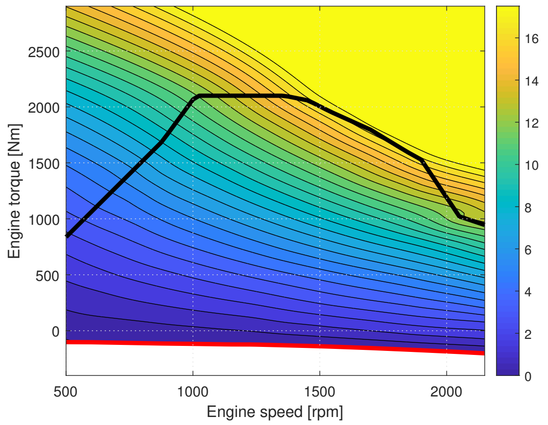

Fig. 1. Fuel consumption rate $[g/s]$ w.r.t. engine speed and torque. The thick black curve is the maximum engine torque w.r.t. the engine speed. The thick red curve is the engine friction torque w.r.t. the engine speed.

2.3 All-in-one optimal control

An optimal control problem can be formulated to minimize the combination of fuel cost and time cost along a trip in $[s_0, s_f]$:

$$\min_{\substack{g,u_c\\T_e,T_{eb}}} \quad w_1 \int_{s_0}^{s_f} \frac{1}{v}\dot{m}_f \mathrm{d}s + w_2 \int_{s_0}^{s_f} \frac{1}{v}\mathrm{d}s \tag{11a}$$

$$s.t. \quad (1), \tag{11b}$$

$$v(s_0) = v_{current}, \tag{11c}$$

$$v(s_f) = v_{final}, \tag{11d}$$

$$v(s) \le v_{law}(s), \tag{11e}$$

$$\omega_{e,min}(y) \le \omega_e \le \omega_{e,max}(y), \tag{11f}$$

$$0 \le T_e \le T_{e,max}(\omega_e), \tag{11g}$$

$$0 \le T_{eb} \le T_{eb,max}(\omega_e), \tag{11h}$$

where $v_{current}$ is the current vehicle velocity, v_{final} is the final velocity, $v_{law}(s)$ is the velocity limit imposed by law, w_1 and w_2 are weighting parameters. Constraints (11f)-(11g) are engine specific constraints, which vary according to the gear position. As a result, they can be considered as vanishing constraints (Kirches, 2011).

After solving problem (11), the optimal torque T_e, T_{eb} and gear position y are obtained simultaneously. However, problem (11) is not easy to solve, especially when there are gear dependent constraints (11f)-(11h). In addition, it is not safe to implement the optimal torques without a driver. The control scheme corresponding to solving (11), denoted as Controller, is shown in Fig. 2.

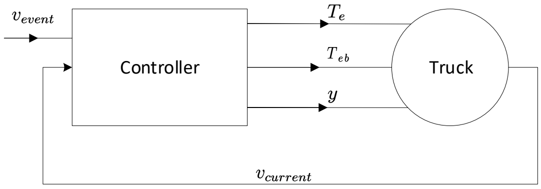

Fig. 2. The scheme of eco-driving control based on (11).

3. DYNAMIC MODEL FOR THE ADVISER

In this work, we limit ourselves to developing an adviser for an optimal deceleration scenario. The adviser will provide optimal velocity, gear and mode profiles to the driver in real-time.

3.1 Driving modes for deceleration

The dynamic model for the adviser does not include the lower level control variables T_e and T_{eb}, but instead consists of abstracted driving modes. We consider four driving modes $M = \{1, 2, 3, 4\}$ in an optimal deceleration scenario:

- Cruising ($M = 1$): The vehicle maintains a constant speed with gear not in neutral ($y > 0$). The dynamics is given by

$$\frac{\mathrm{d}v}{\mathrm{d}s} = \frac{1}{mv}\left(\frac{i_r}{r_w}(T_{pr} - T_b) - T_r\right) = 0, \tag{12}$$

  where T_{pr} is can be simply obtained by solving (12) and $T_b = i_t(y)T_{fric}$.
- Eco-roll ($M = 2$): The vehicle coasts with gear in neutral, i.e. $y = 0$. To ensure the engine is still running, a minimum engine speed is required, thus consuming a small amount of fuel. The dynamics is given by

$$\frac{\mathrm{d}v}{\mathrm{d}s} = \frac{1}{mv}\left(\frac{i_r}{r_w}(T_{pr} - T_b) - T_r\right), \tag{13}$$

  where $T_{pr} = i_t(0)\eta(0)T_e = 0$ and $T_b = i_t(0)T_{fric} = 0$. The torque $T_e = 150$ Nm is assumed for Eco-roll with zero throttle load.
- Coasting ($M = 3$): The vehicle coasts with a positive gear position ($y > 0$). In this mode, there is no fuel injected into the engine, hence the fuel consumption is zero. The dynamics is given by

$$\frac{\mathrm{d}v}{\mathrm{d}s} = \frac{1}{mv}\left(-\frac{i_r}{r_w}T_b - T_r\right), \tag{14}$$

  where $T_b = i_t(y)T_{fric}$. The engine torque is $T_e = 0$.
- Engine brake (ENBR) ($M = 4$): The vehicle decelerates using the modern engine brake functionality with gear positive ($y > 0$). Here, it is always assumed that the maximum engine brake torque $T_{eb} = T_{eb,max}(\omega_e)$ is applied. The dynamics is given by

$$\frac{\mathrm{d}v}{\mathrm{d}s} = \frac{1}{mv}\left(-\frac{i_r}{r_w}T_b - T_r\right) \tag{15}$$

  where $T_b = i_t(y)T_{eb,max} + i_t(y)T_{fric}$. The engine torque is $T_e = 0$.

The fuel consumption map is modeled as (10) with T_e known for each mode and ω_e a function of vehicle velocity v. In the end, the model can be written as

$$\frac{\mathrm{d}v}{\mathrm{d}s} = f(v, M, y), \tag{16}$$

where M and y are the discrete-valued decision variables to be optimized.

3.2 Gear position pruning

A popular way to determine the gear position in eco-driving is to use a backward-looking model. Such model usually employs a velocity reference to predict the gear position along a path given the gear shift look-up table w.r.t. engine speed and vehicle velocity. However, when the velocity reference and actual velocity differ significantly, the prediction is not accurate.

In this work, we employ forward-looking models (1) and (16) for velocity profile optimization hence the velocity reference is not an input but an output. We define sub-modes for the Cruising, Coasting and ENBR mode with different gear positions. In particular, we define $M_{1,i}$, $M_{3,i}$ and $M_{4,i}$ for $i = y_{min}, \ldots, y_{max}$ with y_{min} and y_{max} tuning parameters. As a result, the total number of driving modes is $3(y_{max} - y_{min}) + 1$. Rewrite (16) as

$$\frac{\mathrm{d}v}{\mathrm{d}s} = f(\tilde{M}, v), \tag{17}$$

where $\tilde{M} = \{M_{1,i}, 2, M_{3,i}, M_{4,i}\}$. This is still a dynamic system with only discrete-valued control variables.

If $y_{min} = 1$ and $y_{max} = 12$, all the gear positions are free to choose. However, a low/high gear position may lead to excessively large/small engine speed which is not allowed. Making all gear positions available also increases the computational time since the number of modes is large. In this work, we propose to prune the number of gear positions using heuristics as follows.

$$\begin{aligned} a = \quad & \arg\min y \\ s.t. \quad & \omega_e(\tilde{v}, y) \leq \omega_{e,max}(y) \end{aligned} \tag{18}$$

$$\begin{aligned} b = \quad & \arg\max y \\ s.t. \quad & \omega_e(\tilde{v}, y) \geq \omega_{e,min}(y) \end{aligned} \tag{19}$$

where $\tilde{v} = \frac{1}{2}(v_{current} + v_{event})$. The minimum and maximum gear position available is computed by

$$y_{min} = \begin{cases} 1, & \text{if } a = \emptyset \\ a, & \text{if } a \geq 1 \end{cases} , \tag{20}$$

$$y_{max} = \begin{cases} 12, & \text{if } b = \emptyset \\ b, & \text{if } b \leq 12 \end{cases} . \tag{21}$$

The solution of (20)-(21) are the minimum and maximum gear positions allowed assuming that the velocity $\tilde{v}$ satisfies the engine speed constraints. This heuristics is based on the fact that the gear ratio is monotonically decreasing with the gear position and the truck velocity is also decreasing. As a result, the maximum engine speed constraint would never be violated for $v \leq \tilde{v}$ if no gear down-shift is performed. Similarly, the minimum engine speed constraint would never be violated for $v \geq \tilde{v}$ if no gear up-shift is performed. Note that (20)-(21) give a conservative choice of gear position pruning with only partial constraint satisfaction guarantees.

4. EVENT-TRIGGERED MPC

In this work, the deceleration of a vehicle is triggered by pre-defined speed limiting events. These events include speed limit changes, deceleration of the vehicle in front and highway exits. The preview information together with vehicle status and GPS position is taken into account in an event triggered MPC based adviser. The adviser outputs optimal velocity profile and other driver advice such as gear position and driving mode choice. To distinguish from the deceleration in (adaptive) cruising, we focus on letting the vehicle decelerate to a lower speed within a given distance. The speed of the vehicle is monotonically decreasing and operations needed after the deceleration are not considered.

4.1 Speed limit event

While there may be a number of speed limit events along a path, we take the closest speed limit event as the terminal constraint given by

$$v(s_f) = v_{event}, \tag{22}$$

where v_{event} is the speed limit of the closest event, including the closest vehicle in front and speed limit sign. As a result, only boundary constraints at the start and the end of a road segment are enforced.

4.2 Event triggering rules

A MPC algorithm is designed as the adviser to compute the optimal driving mode suggestions in real-time while driving. Unlike traditional MPC which updates its solution at every sampling instant, the MPC in this work is triggered by detecting the closest speed limit event (22). The rules for the detection are:

- The event is more than 20 meters and less than 1.5 kilometer away in front, i.e. $20 \leq s_f \leq 1500$;
- The event has a speed limit lower than the current vehicle velocity, i.e. $v_{event} \leq v_{current}$.

The driving suggestions are fed back to the driver only if the same event is detected for more than one sampling instant. This can avoid unexpected sensor disturbances and transient speed limiting events. It is worth noting that the proposed event triggered MPC does not affect the physical control loop of the vehicle. The optimal solutions are suggestions to the real actuator (i.e. driver) hence vehicle safety is not an issue.

4.3 Model predictive control

Once an event is detected, we solve (11) with dynamics given in (17). Hence the decision variable is the driving mode with gear position $\tilde{M}$. The optimal velocity profile is obtained by applying the optimal solution to (17). The control scheme denoted as the Adviser is shown in Fig. 3.

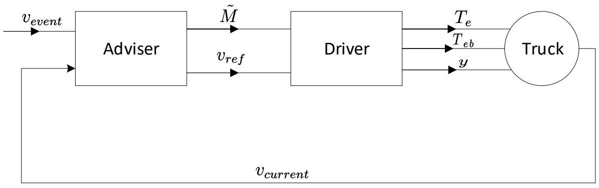

Fig. 3. The scheme of eco-driving adviser based on (11) and (17).

In the proposed control architecture, a (virtual) driver is introduced to convert the driving mode (i.e. $\tilde{M}$, which includes gear positions) and velocity reference to low level commands that a vehicle can understand. The driver can be a human, or a low level controller (e.g. a driver model) that is able to accomplish this conversion. In this work, we assume an *ideal* driver model who follows exactly the suggestions provided by the adviser.

5. NUMERICAL ALGORITHM

To achieve real-time computation, we employ the method for MIOC problem proposed by Sager (2009). First, the problem is discretized and then reformulated by outer convexification and relaxation to be

$$\min_{v,H} \quad \sum_{k=0}^{N-1} \sum_{j=1}^{Q} H_j^k J_k(v, \tilde{M}_j^k) \tag{23a}$$

$$s.t. \quad v_{k+1} = \sum_{j=1}^{Q} H_j^k \phi(v, \tilde{M}_j^k), \tag{23b}$$

$$v_0 = v_{current}, \tag{23c}$$

$$v_N = v_{event}, \tag{23d}$$

$$H_j^k \in [0, 1], \tag{23e}$$

$$\sum_{j=1}^{Q} H_j^k = 1, \forall k = 0, \ldots, N-1 \quad \text{(SOS1)} \tag{23f}$$

where $j = 1, \ldots, Q$ with $Q = 3(y_{max} - y_{min}) + 1$ denoting the number of modes. The index k denotes the discretized points in the prediction horizon which is divided into N sub-intervals of length Δs. The function ϕ is discretized from f in (16) using the 4^{th} order Runge-Kutta method. Similarly, J_k is the discretized cost function for the interval k. The constraint (23f) is the special ordered set of type 1 (SOS1) enforcing only one active mode at any time. The binary solution $\widetilde{M}$ can be recovered from the relaxed solution H by the sum-up rounding (SUR) technique (Sager, 2009), which has a guaranteed bounded approximation error. It has been proven that the computational cost of SUR is negligible, hence we need to only solve a single NLP (23), which can be solved by a variety of standard NLP solvers in real-time. In this work, we choose the interior point method solver IPOPT (Wächter and Biegler, 2006).

6. NUMERICAL SIMULATION

First, a numerical case study is performed with artificially chosen configurations, e.g. the current and event velocity and the distance to the event. Second, a simulation is performed using experimental data from a test drive with a human driver without EDAS.

6.1 Numerical case study

A static deceleration scenario is created with the following configurations:

- current velocity: $v_{current} = 80$ km/h;
- event velocity: $v_{event} = 40$ km/h;
- distance to the event: $s_f = 1$ km;
- discretization distance: $\Delta s = 20$ m.

The results from the eco-driving Controller and Adviser are shown in Fig. 4 and Fig. 5, respectively. It can be observed that the Controller gives a typical pulse-and-glide behavior for the first 300 m, while the Adviser is limited to cruising at a constant speed. However, similar driving styles are obtained for the rest of the distance which includes Eco-roll (gear in neutral), Coasting (gear not in neutral) and ENBR (maximum ENBR torque). For the Adviser, the gear pruning heuristics keeps the engine speed in a reasonable region where the peak of $\omega_e = 2058$ RPM occurs when switching from Coasting to ENBR. The fuel consumption and trip time from the Controller are 38.9g and 52.7s, while those from the Adviser using a conservative tuning are 39.2g and 54.1s. In this case, the Adviser is able to achieve a comparable fuel economy against the Controller at the cost of spending more trip time. Nevertheless, it should be noted that the trade-off between fuel consumption and trip time is always possible for both the Controller and the Adviser. The Adviser requires significantly less computational effort than the Controller. The computational time for the Adviser is 0.33s while the time for the Controller is 80.4s, more than two orders of magnitude faster. The results are obtained using the solver IPOPT in MATLAB on a PC with Intel Core i7-6700 processor running at 3.40 GHz. As a consequence, the proposed eco-driving Adviser can be seen as a safe and sub-optimal alternative to the eco-driving Controller that could be implemented immediately in real-life heavy-duty trucks.

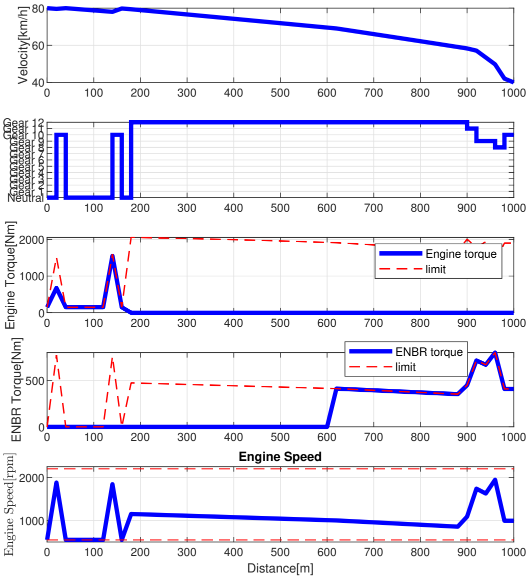

Fig. 4. Velocity profile, gear position, engine torque and ENBR torque results of the numerical case study from the eco-driving Controller using $w_1 = 1$, $w_2 = 18$.

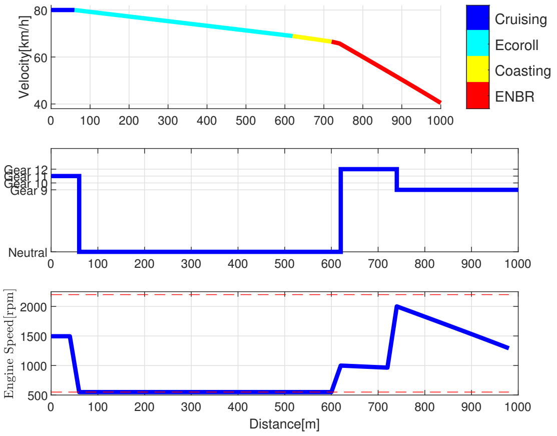

Fig. 5. Velocity profile, mode advice, gear position and engine speed results of the numerical case study from the eco-driving Adviser using a conservative tuning $w_1 = 1$, $w_2 = 60$, $y_{min} = 9$, $y_{max} = 12$.

6.2 Simulation using measured data

A simulation is performed using measured data from a baseline driving performed by a driver without EDAS. The baseline driving profile is obtained by driving a DAF truck on the roads around Eindhoven, The Netherlands. It includes a typical driving circle of 10 km, with multiple speed limiting events defined by 7 roundabouts, 2 highway exits, multiple legal speed limits and sharp corners. The cycle has a negligible slope. During the test, the driver was free to accelerate and decelerate according to the traffic and road conditions.

In Fig. 6, we compare the results from the eco-driving Controller and Adviser using the collected data. A total of 12 deceleration events have been detected excluding two with pulse-and-glide behavior at around 4 km and 6 km. This is because pulse-and-glide is known to be a driving style with much better fuel economy, which is not introduced in the Adviser in this paper. The fuel consumption for the Adviser is 135.3g comparing to 139.3g for the Controller. The trip time for the Adviser is 265.2s comparing to 208.7s for the Controller. It can be concluded that the Adviser is able to achieve 2.9% of fuel reduction with 21.3% more trip time using one specific tuning configuration. More aggressive or conservative tuning options are intuitive but not shown here due to lack of space.

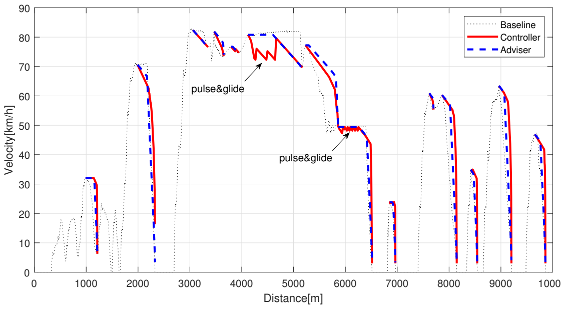

Fig. 6. Simulation velocity profiles: eco-driving Controller, eco-driving Adviser and baseline human driver.

7. CONCLUSION

In this paper, an in-vehicle EDAS has been developed to improve fuel economy. A DIL control framework has been developed, in which, at the high-level, an adviser provides driving mode suggestions while at the low-level, a driver can be incorporated to implement control commands to guarantee safety. For the optimal deceleration scenario, an event-triggered MPC algorithm has been developed for the adviser to compute the optimal driving suggestions using predictive information. A MIOC algorithm has been employed to solve the optimization problem with constraints in real-time. The EDAS has been illustrated using a numerical study and a simulation using measured experimental data. A comparison with the all-in-one optimization-based controller shows the efficiency and effectiveness of the proposed EDAS.

ACKNOWLEDGEMENTS

This work was carried out within the IMPERIUM project (EU H2020 GV-06-2015), while the first author was a PostDoc researcher at the Eindhoven University of Technology. The authors would like to thank Dr. John Kessels (DAF) for providing data and valuable feedback. The contribution of MSc. Nazar Rozkvas to data processing is also gratefully acknowledged.

REFERENCES

Beloufa, S., Cauchard, F., Vedrenne, J., Vailleau, B., Kemeny, A., Mérienne, F., and Boucheix, J.M. (2017). Learning eco-driving behaviour in a driving simulator: Contribution of instructional videos and interactive guidance system. *Transportation research part F: traffic psychology and behaviour.*

Brackstone, M. and McDonald, M. (1999). Transportation research part f: Traffic psychology and behaviour. *vol*, 2, 181–196.

Cheng, Q., Nouveliere, L., and Orfila, O. (2013). A new eco-driving assistance system for a light vehicle: Energy management and speed optimization. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, 1434–1439. IEEE.

Hellström, E., Ivarsson, M., Åslund, J., and Nielsen, L. (2009). Look-ahead control for heavy trucks to minimize trip time and fuel consumption. *Control Engineering Practice*, 17(2), 245–254.

Hibberd, D., Jamson, A., and Jamson, S. (2015). The design of an in-vehicle assistance system to support eco-driving. *Transportation Research Part C: Emerging Technologies*, 58, 732–748.

Huang, Y., Ng, E.C., Zhou, J.L., Surawski, N.C., Chan, E.F., and Hong, G. (2018). Eco-driving technology for sustainable road transport: A review. *Renewable and Sustainable Energy Reviews*, 93, 596–609.

Kamal, M.A.S., Mukai, M., Murata, J., and Kawabe, T. (2010). On board eco-driving system for varying road-traffic environments using model predictive control. In *2010 IEEE International Conference on Control Applications*, 1636–1641. IEEE.

Kirches, C. (2011). *Fast numerical methods for mixed-integer nonlinear model-predictive control.* Springer.

Lin, X., Görges, D., and Liu, S. (2014). Eco-driving assistance system for electric vehicles based on speed profile optimization. In *2014 IEEE Conference on Control Applications (CCA)*, 629–634. IEEE.

Ozatay, E., Ozguner, U., and Filev, D. (2017). Velocity profile optimization of on road vehicles: Pontryagin's maximum principle based approach. *Control Engineering Practice*, 61, 244–254.

Padilla, G., Weiland, S., and Donkers, M. (2018). A global optimal solution to the eco-driving problem. *IEEE control systems letters*, 2(4), 599–604.

Saerens, B. (2012). Optimal control based eco-driving. *Theoretical Approach and Practical Applications. Heverlee: Katholieke Universiteit Leuven.*

Sager, S. (2009). Reformulations and algorithms for the optimization of switching decisions in nonlinear optimal control. *Journal of Process Control*, 19(8), 1238–1247.

Sciarretta, A., De Nunzio, G., and Ojeda, L.L. (2015). Optimal ecodriving control: Energy-efficient driving of road vehicles as an optimal control problem. *IEEE Control Systems Magazine*, 35(5), 71–90.

Thijssen, R., Hofman, T., and Ham, J. (2014). Ecodriving acceptance: An experimental study on anticipation behavior of truck drivers. *Transportation research part F: traffic psychology and behaviour*, 22, 249–260.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1), 25–57.

Yin, F., Hayashi, R., Pongsathorn, R., and Masao, N. (2013). Haptic velocity guidance system by accelerator pedal force control for enhancing eco-driving performance. In *Proceedings of the FISITA 2012 World Automotive Congress*, 37–49. Springer.