Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Receding Horizon Reference Governor for Implementable and Optimal Powertrain-Aware Eco-Driving ⋆

Daliang Shen * **Jihun Han** * **Jongryeol Jeong** *
Dominik Karbowski * **Aymeric Rousseau** *

* *Center for Transportation Research, Energy Systems Division, Argonne National Laboratory, Lemont, IL 60439 USA (e-mail: {dshen, jihun.han, jjeong, dkarbowski, arousseau}@anl.gov).*

Abstract: This paper presents an adaptive, two-level control structure that makes it possible to implement a numerical optimization algorithm for eco-driving in real time. The reference governor in the higher level is characterized by a low, flexible sampling rate and adopts a receding horizon for preview and optimization. The optimization algorithm thereby finds the energy-minimizing solution, based on Pontryagin's minimum principle (PMP), for traveling the selected route segments without colliding with the preceding vehicle. The tracking control in the lower level compensates for errors due to modeling inaccuracies and unmodeled disturbances. The hierarchical structure can accommodate different types of numerical solvers and control schemes that apply to various vehicle powertrain configurations. A large-scale simulation study using real-world route data with high-fidelity powertrain models validates the proposed control structure and its online implementation.

Keywords: nonlinear and optimal automotive control, constrained control, eco-driving, minimum principle, connected and automated vehicle

1. INTRODUCTION

In recent years, the system integration of computing units onto onboard vehicle systems has been developing at a fast pace, providing vehicles with sufficient computing capability to carry out advanced driver assistance functions and even fully automated driving. Given advanced connectivity to numerous data sources, such as road infrastructure and neighboring vehicles, these connected and automated vehicles (CAVs) possess an unprecedented potential to optimize their decisions and behaviors on the road. Researchers are harnessing these advances such that energy efficiency is increasingly taking its place beside safety and comfort as the focus of the automotive control realm (Vahidi and Sciarretta, 2018).

Much research has been conducted on solving the optimal control problem (OCP) of eco-driving. An exhaustive list of related papers is presented in Sciarretta et al. (2015). By linearizing the system and using quadratic cost functions, the OCP could be simplified into quadratic programming (Dollar and Vahidi, 2018; Schwickart et al., 2014), or converted into a boundary value problem with solutions in closed-form expressions (Dib et al., 2014; Han et al., 2018). The low computation time that results from these simplifications makes these approaches easy to implement online in a model predictive control (MPC) framework. However, these simplifications may lead to loss of modeling accuracy in nonlinear factors such as aerodynamic drag, road grade, and transmission viscosity, which could lead to suboptimal control policies.

In order to gain more confidence in the optimization results, researchers need to incorporate nonlinearity. For example, Xu et al. (2015) and Khalik et al. (2018) adopt direct methods and convert the OCP into nonlinear programming, whereas the research of Passenberg et al. (2009) falls into the category of indirect methods and applies a multiple shooting algorithm. A few of these numerical methods have been evaluated online in closed-loop simulations using an MPC framework. Generally, the time needed for a single optimization is about 1 s, which makes the real-time implementation of these methods questionable.

In our previous works (Shen et al., 2018a,c), we adopt a Pontryagin's minimum principle (PMP)-based approach to solving the OCP. The approach establishes a mapping of the solution trajectories to the target steady-state speed. For free-flow cruising scenarios, the method is able to yield the solution in one step, which runs within 20 ms. However, it requires iterations, as with numerical methods, to adapt the solution to time constraints in car-following and traffic-light-crossing scenarios. Compared to other nonlinear optimal control algorithms, our approach is faster but not sufficient for a direct closed-loop application.

In this paper, we introduce a two-level control structure to accommodate the nonlinear optimization algorithm without compromising real-time capability (see Fig. 1).

⋆ This report and the work described were sponsored by the U.S. Department of Energy (DOE) Vehicle Technologies Office (VTO) under the System and Modeling for Accelerated Research in Transportation (SMART) Mobility Laboratory Consortium, an initiative of the Energy Efficient Mobility Systems (EEMS) Program.

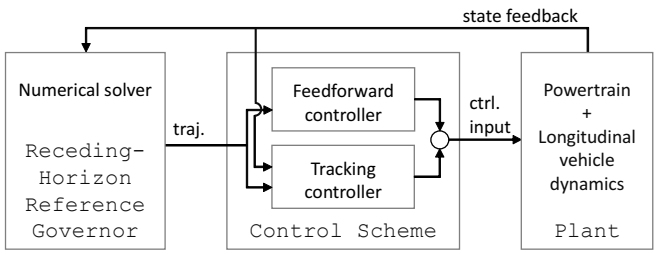

Fig. 1. Overview of control structure.

In the higher-level reference governor that is executed in a low average sampling rate, the numerical solver repeats the optimization calculation according to updated information on the receding horizon. The lower-level tracking control, running in a fast sampling rate, adopts a two-degrees-of-freedom scheme that keeps track of the optimization solution while suppressing disturbances and errors.

2. PROBLEM FORMULATION

The task of ecological longitudinal vehicle guidance is to minimize energy consumption, for a defined travel distance, by a connected and automated vehicle (CAV). A CAV is defined as a vehicle provided with look-ahead information and fully governed by the controller on board.

2.1 Longitudinal Vehicle Dynamics

The vehicle's movement is a complex model that includes nonlinearities in tires and steering. The majority of energy consumption occurs in the longitudinal movement of road vehicles, so we can focus on the longitudinal dynamics. The longitudinal vehicle dynamics are depicted in Fig. 2.

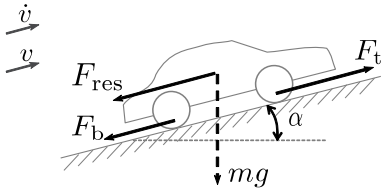

Fig. 2. Vehicle longitudinal dynamics

The longitudinal vehicle model can be described in the following equations (Guzzella and Sciarretta, 2007):

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{v} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} \hat{m}^{-1} \cdot \left(T_{\mathrm{m}} \gamma \eta^{\mathrm{sgn}(T_{\mathrm{m}})} - F_{\mathrm{b}} - F_{\mathrm{res}} \right) \\ v \end{bmatrix}, \tag{1}$$

where s and v are the traveled distance and speed of the vehicle, respecively, and $\hat{m}$ is the total inertia, including the vehicle mass and equivalent mass of all rotating parts. The motor torque T_{m} multiplied by the transmission ratio γ and the efficiency η comprises the traction force F_{t}. The mechanical brake force and resistance force are denoted by F_{b} and F_{res}, respectively. F_{res} is the sum of the wheel rolling friction, the aerodynamic drag, and the uphill force. Thus, F_{res} is expressed as a quadratic polynomial of v with coefficients c_0, c_1, and c_2,

$$F_{\mathrm{res}}(v) = c_0 + c_1\, v + c_2\, v^2 + mg\, \sin\alpha, \tag{2}$$

where g is the gravitational acceleration, α is the road grade, and m denotes the vehicle mass (excluding the rotational inertia).

2.2 Optimal Control Problem Formulation

The optimal trajectory is generated by solving an eco-driving OCP. The OCP is formulated as follows:

$$\min_{u \in \mathcal{U}} \quad J = \int_{t_0}^{t_{\mathrm{f}}} \phi(\boldsymbol{x}, \boldsymbol{u})\, \mathrm{d}t \tag{3a}$$

subject to

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \tag{3b}$$
$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{3c}$$
$$\boldsymbol{g}(\boldsymbol{x}(t_{\mathrm{f}}), t_{\mathrm{f}}) = \boldsymbol{0}, \tag{3d}$$
$$\tilde{\boldsymbol{g}}(\boldsymbol{x}(t_i), t_i) = \boldsymbol{0}, \tag{3e}$$
$$\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}) \leq \boldsymbol{0}. \tag{3f}$$

The OCP is to minimize the overall energy consumption J that integrates the running cost ϕ from an initial time t_0 to a final time t_{f}, subject to the system dynamics (3b) given in (1). Boundary conditions at the initial and final time are stated in (3c) and (3d); interior-point boundary conditions are presented by (3e). Inequality conditions are summarized in (3f) (see Section 3.2). The running cost ϕ approximates the energy consumption rate with a quartic piecewise function of v and T_{m}, with the two groups of coefficients $c_{\{\mathrm{m,g}\},ij}$ associated with the motor ($T_{\mathrm{m}} \geq 0$) and generator ($T_{\mathrm{m}} < 0$) modes,

$$\phi(v, T_{\mathrm{m}}) = \sum_{i=0}^{2} \sum_{j=0}^{2} c_{\{\mathrm{m,g}\},ij}\, (\gamma v)^i\, T_{\mathrm{m}}^j.$$

The state vector is $\boldsymbol{x} = [v, s]^{\mathsf{T}}$, and the control vector is $\boldsymbol{u} = [T_{\mathrm{m}}, F_{\mathrm{b}}]^{\mathsf{T}}$.

We assume that before the passenger starts the driving mission, a certain route with a destination is selected, either by the passenger or by the navigation computer. That allows us to convert the continuous slope data obtained from a digital map into piecewise-constant profiles (Shen et al., 2018a). The converted slope profile and the speed limit are inherently piecewise constants. These two piecewise constant profiles are expressed in a discrete form:

slope: $\alpha(s) = \alpha_i$, for $s \in [s_{\mathrm{s},i}, s_{\mathrm{s},i+1})$,
speed limit: $v_{\mathrm{l}}(s) = v_{\mathrm{l},j}$, for $s \in [s_{\mathrm{l},j}, s_{\mathrm{l},j+1})$,

with the slope starting positions $s_{\mathrm{s},i} \in \mathcal{S}_{\mathrm{S}}$ and the speed limit starting positions $s_{\mathrm{l},j} \in \mathcal{S}_{\mathrm{L}}$. In addition, the positions of pointwise road traits such as stop signs $s \in \mathcal{S}_{\mathrm{P}}$ and traffic lights $s \in \mathcal{S}_{\mathrm{TL}}$ must be considered for route segmentation. Consequently, in a preprocess the route is divided into a series of segments at positions $s \in \mathcal{S}_{\mathrm{RS}} = (\mathcal{S}_{\mathrm{S}} \cup \mathcal{S}_{\mathrm{L}} \cup \mathcal{S}_{\mathrm{P}} \cup \mathcal{S}_{\mathrm{TL}})$.

3. RECEDING-HORIZON REFERENCE GOVERNOR

The structure of the reference governor is shown in Fig. 3. A *refresh trigger* is introduced to conditionally renew the OCP (3) in *previewer* and to re-calculate a new solution by *optimizer*.

3.1 Refresh Trigger

In contrast to a typical MPC speed control, such as those described in Passenberg et al. (2009) and Asadi and Vahidi (2011), the proposed reference governor refreshes the receding horizon not on a fixed sampling rate but on a

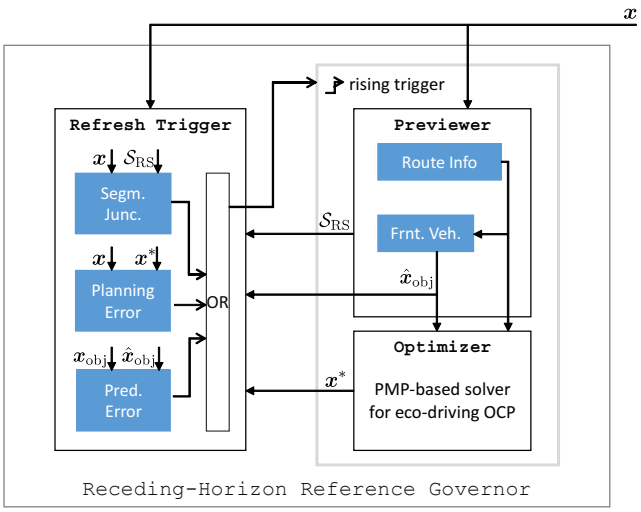

Fig. 3. Block diagram of the receding horizon reference governor.

non-regular, route-segment-based interval. In addition to triggering at segment junction points, a refresh request is triggered if the planning or prediction error is too large, as shown in the left block in Fig. 3. This refresh is to keep the OCP formulation accurate and maintain the system within a confidence band around the planned trajectory.

3.2 Previewer: Update Problem Formulation

The initial condition (3c) is updated with the current position and speed. The preview horizon of a fixed time length T_h moves along as the vehicle moves forward, as shown in Fig. 4. The horizon length l_h is thus a linear

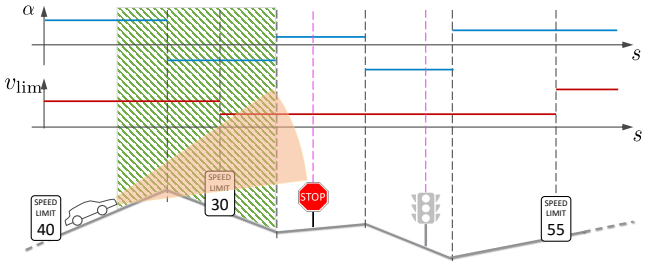

Fig. 4. Preview horizon of route information.

function of the initial vehicle speed v_0:

$$l_\mathrm{h} = T_\mathrm{h}\, v_0. \tag{4}$$

The maximum number of route segments within the preview horizon is limited to n_hs.[1] Let the vehicle position at the triggering of the renewed preview be on the i_rs-th segment, $s(t) \in [s_{\mathrm{rs},i_\mathrm{rs}-1}, s_{\mathrm{rs},i_\mathrm{rs}})$. In (3d), the final position $s(t_\mathrm{f}) = x_2(t_\mathrm{f})$ is definitive:

$$s(t_\mathrm{f}) = \min(l_\mathrm{h},\, s_{\mathrm{rs},i_\mathrm{rs}+n_\mathrm{hs}-1}). \tag{5}$$

The final speed $v(t_\mathrm{f})$ is left free; except for a stop sign, $v(t_\mathrm{f}) = 0$. In general, the final time t_f is free due to its ambiguity in the middle of the entire travel.

Control and state inequality constraints (3f) are detailed as follows:

[1] In this work, we set $n_\mathrm{hs} = 3$.

$$\boldsymbol{h} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \end{bmatrix} = \begin{bmatrix} T_\mathrm{m,min} - T_\mathrm{m} \\ T_\mathrm{m} - T_\mathrm{m,max} \\ 0 - F_\mathrm{b} \\ F_\mathrm{b} - F_\mathrm{b,max} \\ v - v_\mathrm{l}(s) \\ s - s_\mathrm{b}(t) \end{bmatrix} \leq 0. \tag{6}$$

Constraints on the control variables T_m and F_b resulting from physical limitations of the powertrain parts are represented by $h_{\{1,2,3,4\}}$. Constraints $h_{\{5,6\}}$ are constantly updated by the preview horizon. The speed limit v_l is enforced by h_5, and collisions are avoided by h_6, where the position-bound s_b takes into account the front vehicle position s_obj and the safety headway rule as a linear function of the front vehicle speed $\dot{s}_\mathrm{obj}$ (Ioannou and Chien, 1993),

$$s_\mathrm{b} = s_\mathrm{obj} - (c_{\mathrm{h},1}\dot{s}_\mathrm{obj} + c_{\mathrm{h},0}),$$

where $c_{\mathrm{h},\{0,1\}}$ are the coefficients of the safety headway rule. The front vehicle's movement in the future is uncertain. In this paper, we apply a well-established car-following model, the ideal driver model (IDM) (Treiber and Kesting, 2013, Ch. 11), to predict how the front vehicle reacts to the traffic-regulating devices that appear in the preview horizon. An example is shown in Fig. 5.

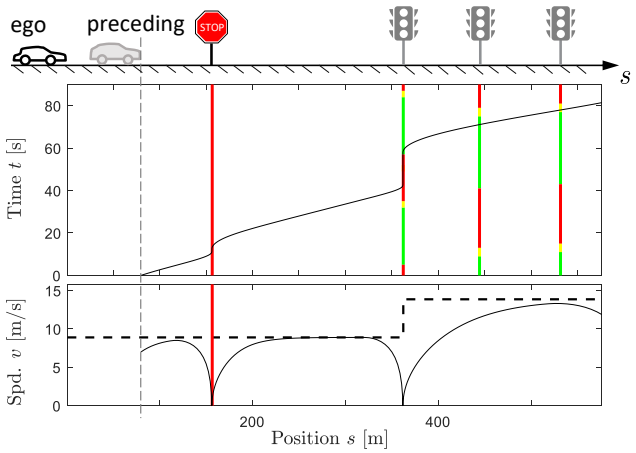

Fig. 5. Prediction of the front vehicle using IDM.

The condition (3e) is used to describe the interior-point constraints pursuant to the traffic-light-crossing strategy

$$\tilde{g}_i = s(t_{\mathrm{tl},i}) - s_{\mathrm{tl},i} = 0, \quad i = 1, ..., n_\mathrm{tlh}, \tag{7}$$

where n_tlh denotes the number of traffic lights within the preview horizon.[2] The crossing times $t_{\mathrm{tl},i}$ at each traffic light position $s_{\mathrm{tl},i} \in \mathcal{S}_\mathrm{P}$ are calculated by a near-optimal decision algorithm inspired by De Nunzio et al. (2013) each time the receding preview horizon is triggered.[3]

[2] Due to the maximum segment number of the preview horizon, $n_\mathrm{tlh} \leq n_\mathrm{hs} - 1$.

[3] The algorithm (1) determines the feasible s-t-region based on speed limits, (2) identifies the green windows within the feasible region and simplifies them as vertices, (3) finds the passing path with the lowest energy cost using Dijkstras's algorithm, and finally (4) refines the crossing times in the selected green windows by solving a static optimization problem. See De Nunzio et al. (2013) for more details.

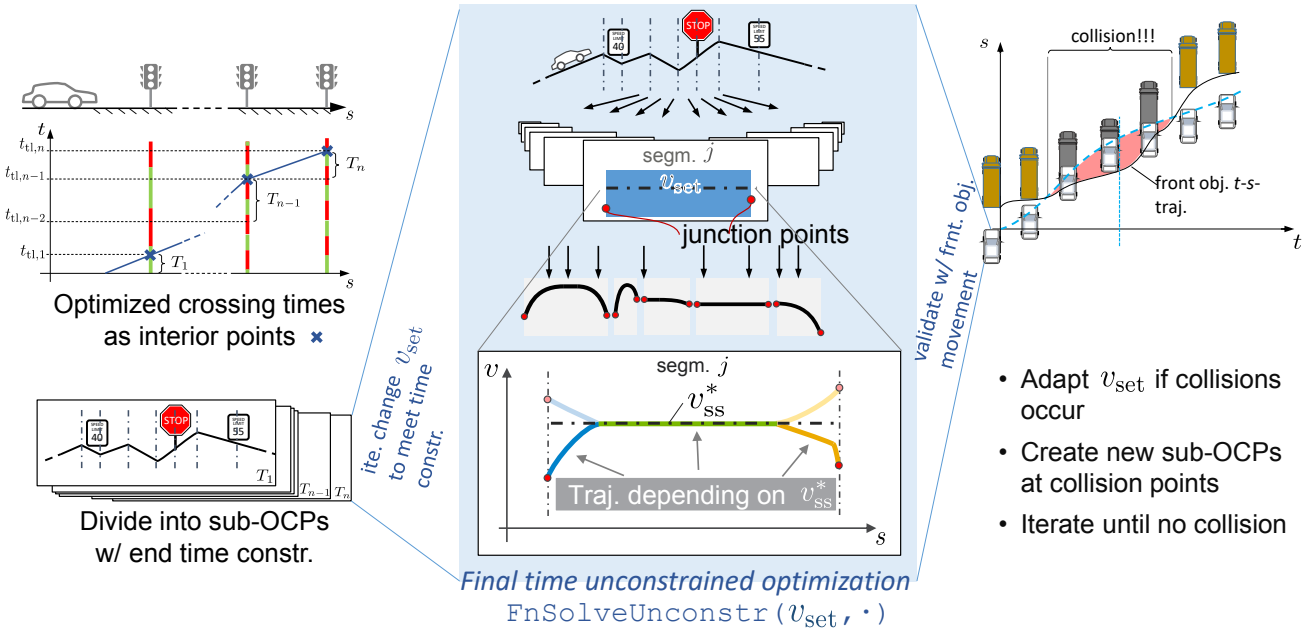

Fig. 6. Workflow of the proposed algorithm for solving a CAV eco-driving OCP.

3.3 Optimizer: Renew Solution Trajectories

In three previous papers (Shen et al., 2018a,b,c), the authors have presented the PMP-based algorithm to solve the OCP in the form of (3). PMP transforms the OCP (3) into the Hamiltonian,

$$H(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\lambda},\boldsymbol{\mu}) = \phi(\boldsymbol{x},\boldsymbol{u}) + \boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) + \boldsymbol{\mu}^{\mathsf{T}}\boldsymbol{h}(\boldsymbol{x},\boldsymbol{u}),$$

with an appended co-state $\boldsymbol{\lambda} = [\lambda_{\mathrm{v}}, \lambda_{\mathrm{s}}]^{\mathsf{T}}$ and a Lagrangian multiplier $\boldsymbol{\mu}$.

Analysis in Shen et al. (2018a,c) shows that the Hamiltonian H has a constant optimal value $\mathfrak{H}$ between interior points. It incentivizes us to divide the OCP (3) into sub-OCPs, whose connections are subject to the interior-point conditions. λ_{s} is inherently constant in each route segment (of a constant slope and constant speed limit). Thus, the local minimum condition in PMP,

$$\boldsymbol{u}^* = \arg\min_{\boldsymbol{u}\in\boldsymbol{\mathcal{U}}} H(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\lambda}),$$

leads to the control policy, which is a function of vehicle speed v parameterized with constants $\mathfrak{H}$ and λ_{s},

$$\boldsymbol{u}^* = \boldsymbol{\Upsilon}(v \,|\, \mathfrak{H}, \lambda_{\mathrm{s}}). \tag{8}$$

Through a steady-state analysis at a speed v_{ss}, that is, by letting $\dot{v}(\boldsymbol{u}^*) = \dot{v}[\boldsymbol{\Upsilon}(v_{\mathrm{ss}} \,|\, \mathfrak{H}, \lambda_{\mathrm{s}})] = 0$, a mapping is established, $\pi : \mathbb{R} \to \mathbb{R}^2 : v_{\mathrm{ss}} \mapsto [\mathfrak{H}, \lambda_{\mathrm{s}}]$ (see Shen et al., 2018b,c). It is obvious that this mapping π is dependant on the slope α as evident in (1) and (2). Let v_{set} be the v_{ss} for the general case $\alpha = 0$, which corresponds to a certain pair $[\mathfrak{H}, \lambda_{\mathrm{s}}]$ according to the mapping. Because $H = \mathfrak{H}$ remains constant (in each sub-OCP), the mapping π parameterized by α is used again to derive v_{ss} for each slope segment. In other words, once v_{set} is chosen, v_{ss} is determined for all of the route segments within the sub-OCP; and by extension, the state and control trajectories of acceleration, deceleration, and constant-speed for all of the route segments are determined. The unknown junction speeds between neighboring segments are found by satisfying the continuity in λ_{v}. The above process is packaged as a function in Matlab, `FnSolveUnconstr`$(v_{\mathrm{set}}, \boldsymbol{x}_0, \boldsymbol{x}_{\mathrm{f}})$, as illustrated in the middle column of Fig. 6. It can yield the optimal solution in one step with a desired v_{set} defined by the passenger.

To fulfill the interior-point constraints (7) due to traffic lights, within each sub-OCP, we adapt the input v_{set} of the function `FnSolveUnconstr()` iteratively to approach the required crossing time (see the first column of Fig. 6). The optimization of junction speeds between sub-OCPs (at traffic lights) is relaxed by assuming that they are the arithmetic means of the average speeds within the two adjacent route segments just before and after the traffic lights.[4] The values of the average speeds can be directly obtained from the calculation for the traffic light crossing strategy (7).

The resulting trajectories then must be validated by the inequality condition h_6 to ensure a safe distance to the front vehicle (in a predicted way).[5] The numeric method of solving the OCP (3) with an active h_6 is introduced in Shen et al. (2018c). As illustrated in the right column of Fig. 6, first a new interior point is created at the position where h_6 is most violated and the affected OCP (or sub-OCP) is split in two; then in the two newly created sub-OCPs, v_{set} is adapted in `FnSolveUnconstr()` to meet the interior time points. Then these steps are repeated until no violation is detected.

As a consequence of solving the OCP (3) through the algorithm described above, the trajectories of both states and co-states (including the Hamiltonian) are computed. In the next section, we explain how to utilize the two groups of trajectories.

[4] Because v is not concerned in (7), junction speeds should be co-optimized along with v_{set} for all sub-OCP together to reach a global optimum. The sub-optimal approach in this paper is adopted in light of computation efficiency for online-control.

[5] The rest of the inequality constraints $h_{\{1\sim5\}}$ are already considered in `FnSolveUnconstr()`.

4. CONTROL SCHEME

The output signals of the *receding-horizon reference governor* are the optimal trajectories of the vehicle speed v and the traveled distance s as well as of the parameter pair, $[\mathfrak{H}, \lambda_s]$. The control scheme adopts a two-degrees-of-freedom design as shown in Fig. 7. That is, an adaptive feedforward controller with the speed feedback and a tracking controller to compensate errors are both used to obtain the desired longitudinal dynamics.

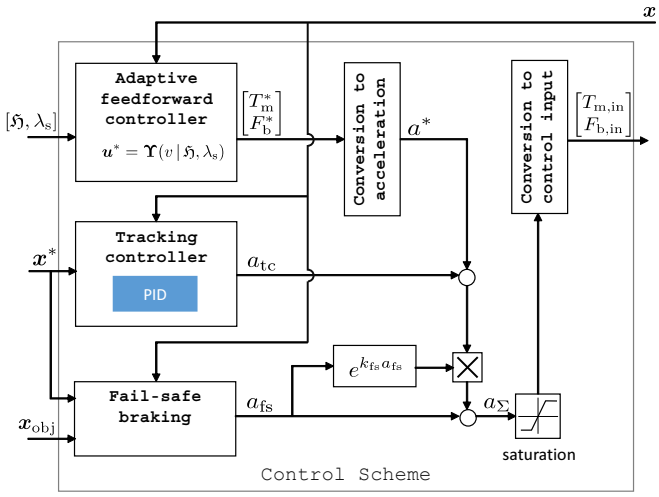

Fig. 7. Block diagram of the tracking control.

4.1 Adaptive Feedforward Controller

The feedforward controller is based on the parameterizable control policy $\boldsymbol{u}^*$, which is dependent on the vehicle speed v, as shown in (8). The parameter pair $[\mathfrak{H}, \lambda_s]$, as stated in Section 3.3, is piecewise constant and jumps only at route segment junctions. Thus, $[\mathfrak{H}, \lambda_s]$ is indexed with position marks. In the absence of modeling errors and disturbances, the feedforward controller should ensure exact tracking of the generated trajectory.

4.2 Tracking Controller

The tracking controller reduces the error between the current state and the reference state according to the generated optimal trajectory. A proportional-integral-derivative (PID) controller is applied in this paper to demonstrate the control structure. [6] The feedback is the sum of the error gains of acceleration, speed, and position, in reference to the state trajectory $\boldsymbol{x}^*$ in the time domain.

4.3 Fail-Safe Braking

Because of inevitable prediction errors and unpredictable disturbances, the reference-tracking controller may still fail to prevent the vehicle from violating safety-critical constraints, such as running red lights or colliding with the preceding vehicle. As a fail-safe for braking the vehicle to a full stop in extreme cases, we apply the intelligent braking strategy of IDM (Treiber and Kesting, 2013); this produces feedback a_{fs} according to the distance gap $(s_{obj} - s)$ between the vehicle and the front object, which could be either a vehicle or a red traffic light,

$$a_{fs} = \min\big(0,\, a_0\,(1 - z^2)\big), \tag{9a}$$

with

$$z = \frac{1}{s_{obj} - s}\left[p_{d0} + \max\left(0,\, v\,p_T + \frac{v\,(v - \dot{s}_{obj})}{2\sqrt{p_a p_b}}\right)\right]. \tag{9b}$$

The p_a, p_b, p_T, and p_{d0} are model parameters of IDM, representing comfortable acceleration, deceleration, time gap, and minimum distance gap, respectively. For a standing vehicle or a red light, $\dot{s}_{obj} = 0$.

The output a_{fs} is added to the sum of the tracking feedback a_{tc} and the feedforward portion a^*. Here, when the fail-safe braking comes into play, its dominance needs to be assured. Thus, the contribution of the sum $(a_{tc} + a^*)$ is throttled down by multiplying an exponential function of a_{fs}, namely $\exp(k_{fs}\, a_{fs})$, with a coefficient $k_{fs} > 0$. The resulting value a_Σ for the longitudinal acceleration request is then limited for comfort and safety reasons. The control demand fed to the powertrain is obtained by converting that adjusted acceleration request.

5. SIMULATION RESULTS

The control system presented herein was validated on RoadRunner, a multi-vehicle energy consumption and performance simulation platform (Kim et al., 2018). In RoadRuner, the emulated interaction interfaces that exist between vehicles and road infrastructure allow us to evaluate the merits of the proposed control in a simulation environment backed by real-world route data. For each single vehicle, RoadRunner deploys the high-fidelity powertrain and longitudinal vehicle dynamics models of Autonomie (Argonne National Laboratory, 2019), which provide us with an accurate estimation of the energy impact.

In this paper, we compare three control strategies: (1) a human driver model as baseline (BSL), (2) speed-only eco-driving (Eco-Spd), and (3) our proposed control (Eco-SpdPT). The human-driver BSL was developed and validated using on-road testing data (Han et al., 2020b). Eco-Spd uses an MPC framework to implement online closed-form optimization solutions that minimize acceleration energy subject to the state constraints imposed by speed limits and the preceding vehicle (Han et al., 2020a). [7]

Note that the strategies of horizon length choice on Eco-Spd and Eco-SpdPT are different: Eco-Spd previews a fixed-length distance and considers only one traffic light ahead, whereas Eco-SpdPT's preview horizon varies with vehicle speed and is dependent on the route segmentation, as shown in (4) and (5), thereby considering up to two traffic lights ahead. Moreover, the sampling rate of the MPC frames on Eco-Spd and Eco-SpdPT are very different: It is fixed at 10 ms with Eco-Spd, and it is floating with an average of about 1 s with Eco-SpdPT (see Section 3.1). Therefore, the results shown in the following are by no means meant for a performance contest but only as a quality control for the proposed concept.

[6] Of course, more sophisticated controllers like MPC could be implemented in its place.

[7] Aerodynamic drag, road grade, and powertrain factors are not considered.

5.1 Simulation Setup

A large-scale study was executed for the three control strategies on 33 real-world routes extracted from HERE maps, as shown in Fig. 8. In the study, the same electric

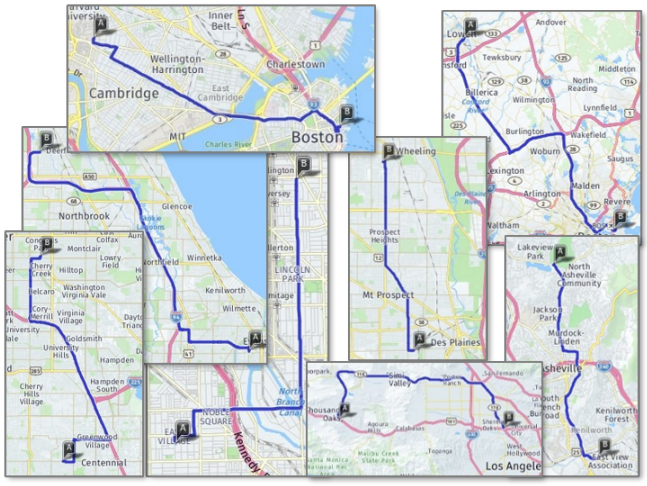

Fig. 8. Routes extracted from HERE digital maps: 6 urban, 8 suburban, 5 mixed (urban + suburban + highway), and 14 highway routes.

vehicle plant was used, the configuration of which is shown in Table 1. Each test case involved two vehicles, one

Table 1. Vehicle parameters

Parameter	Value	Unit
Total Mass	1784	kg
Motor Max. Power	123.9	kW
Motor Max. Torque	393.7	N m
Wheel Radius	0.301	m
Battery Energy	59.89	kW h
Gear Ratio	1.6	-
Final Drive Ratio	3.5	-

leading and one following. We named the vehicle position arrangement: LF_00, LF_01, LF_10, and LF_11, where L refers to leading, F to following, the digit 0 to BSL, and 1 to an advanced strategy, either Eco-Spd or Eco-SpdPT. Through permutation of the control strategy and the vehicle position (leading/following), there are in total 10 cases. [8]

The BSL and the Eco-Spd strategies both use the default rule-based supervisory controller for an electric vehicle with a single-speed gear transmission (in the Autonomie library). The acceleration/pedal-position signal output from BSL and Eco-Spd is interpreted by the supervisory controller and converted into detailed demands for the motor and the brake. On the other hand, the Eco-SpdPT strategy is a self-contained online implementation approach to vehicle speed and powertrain control. The error-compensated demand output for the motor torque and brake force is directly fed to the vehicle plant.

[8] The cases with different eco-driving strategies (Eco-Spd and Eco-SpdPT) in the leading and following vehicles are not considered.

5.2 Analysis of Results

The relative energy and travel time differences for the leading vehicle and the following vehicle are displayed in Figs. 9 and 10, respectively, where the comparisons are in reference to case LF_00. Fig. 9 is for analysis in a free-

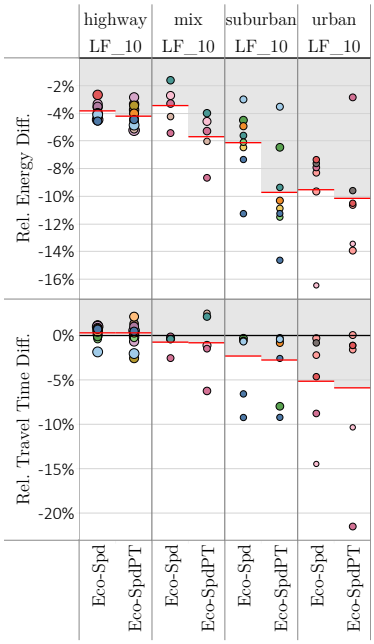

Fig. 9. Overview of relative energy and travel time differences for the *leading* vehicle.

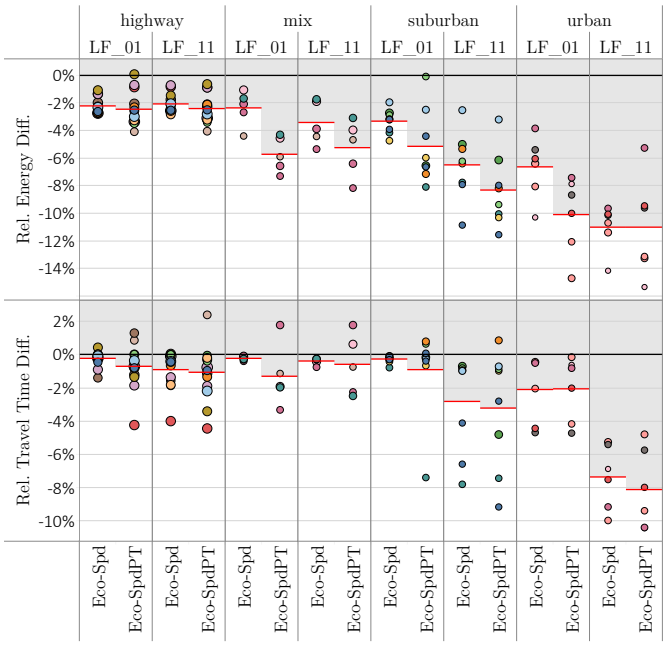

Fig. 10. Overview of relative energy and travel time differences for the *following* vehicle.

flow condition, and Fig. 10 is for analysis in a car-following condition. The circles in different colors represent different routes, and the red lines indicate the average value in each cell.

At first glance across the route type row from highway to urban in both Figs. 9 and 10, the results show a general trend that the energy-saving potential grows as the intersection occurrence density increases without sacrificing travel time.

In highway LF_10, where other constraint factors (e.g., the preceding vehicle) play no role, Eco-SpdPT outperforms Eco-Spd in terms of energy savings ($-4.21\,\%$ vs. $-3.83\,\%$) because it considers the system complexity (e.g., aerodynamic drag, road grade). The energy savings gap between Eco-SpdPT and Eco-Spd becomes larger in suburban routes as a result of different preview strategies. The preview length of Eco-SpdPT floats between about 100 and 2000 m depending on the speed in suburban routes, whereas Eco-Spd previews at maximum 250 m ahead and not beyond the first intersection in sight. The performance advantage of Eco-SpdPT over Eco-Spd decreases in urban routes due to the fact that shorter intersection distances lead to shorter preview length in Eco-SpdPT, which is comparable to the one in Eco-Spd.

We now take a closer look at an urban route in Atlanta. The simulation results of Eco-SpdPT on the leading and the following vehicles are displayed in Fig. 11 and Fig. 12, respectively.

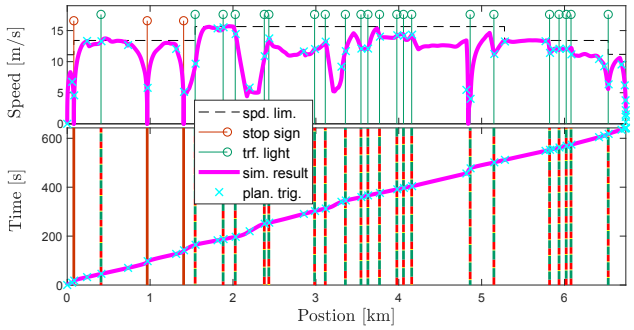

Fig. 11. Speed and time traces over position for the *leading* vehicle with Eco-SpdPT.

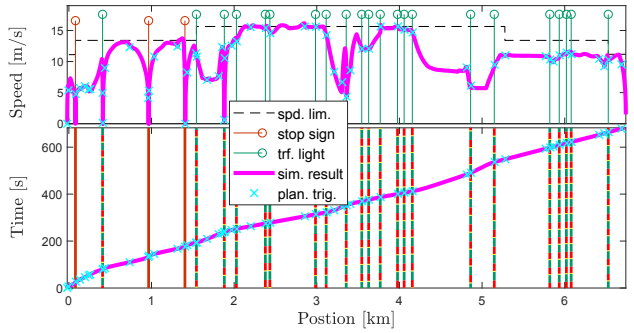

Fig. 12. Speed and time traces over position for the *following* vehicle with Eco-SpdPT.

The reference governor was triggered 47 times for the leading vehicle (in LF_10) and 66 times for the following vehicle (in LF_01), as marked by cyan ×s. The additional triggerings in LF_01 occurred between about 0 and 4 km, where the preceding vehicle (BSL) was within the preview horizon length. Because of errors in predicting the preceding vehicle's movement, it was necessary for the reference governor to be engaged more often and to renew the optimization (see Section 3.1).

To aid in evaluating the method's real-time capability, the statistics of the time intervals T_Δ between each triggering event and the execution time T_{exe} of the reference governor are displayed in Fig. 13. For LF_10, all T_{exe} are less than

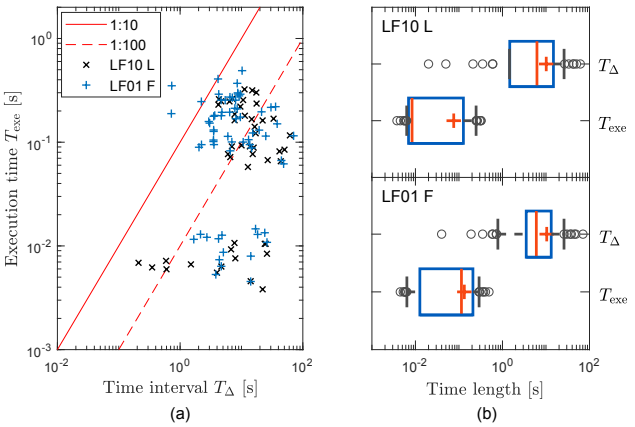

Fig. 13. Statistics of triggering intervals T_Δ and execution time T_{exe}. a) Scattered data points of $(T_\Delta, T_{\text{exe}})$. b) Box plots T_Δ of and T_{exe} in case LF_01 (above) and LF_10 (below).

1/10 of T_Δ; and about half of the T_{exe} are less than 1/100 of T_Δ. For LF_01, in 3 out of the 66 instances, T_{exe} are over 1/10 of T_Δ; in the worst case, $T_{\text{exe}} = 0.35\,\text{s}$ and $T_\Delta = 0.73\,\text{s}$. The extra iterations for collision avoidance in the numerical optimization (see Fig. 6) contributed to the additional amount of execution time. Overall, the proposed control concept could be considered as being real-time capable with tolerance to a few extreme cases.

6. CONCLUSIONS

Herein we have presented the design and strategies of an online-implementable, powertrain-aware eco-driving control approach. The proposed modular design separates the longitudinal vehicle guidance into two levels. In the higher-level reference governor at a low sampling rate, the eco-driving OCP is numerically solved. The lower-level tracking control at a high sampling rate compensates for disturbances and/or errors due to modeling inaccuracies. Through a large-scale simulation study, it was demonstrated that the two-level approach accommodates the time-consuming numerical optimization in the reference generation well, and the online implementation is real-time capable.

The control concept is applied to electric vehicles in this paper. Future work will expand its application to vehicles with other powertrain types. Of course, due to mode selections, the OCP becomes hybrid; thus, it requires further investigation to guarantee obtaining the global optimum in the solution in the planning level and to ensure robustness and stability in the lower tracking level.

ACKNOWLEDGMENTS

DOE Office of Energy Efficiency and Renewable Energy (EERE) manager David Anderson played an important role in establishing the project concept, advancing implementation, and providing ongoing guidance.

DISCLAIMER

REFERENCES

Argonne National Laboratory (2019). Autonomie - Home. URL `www.autonomie.net`.

Asadi, B. and Vahidi, A. (2011). Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time. *IEEE Transactions on Control Systems Technology*, 19(3), 707–714. doi:10.1109/TCST.2010.2047860.

De Nunzio, G., Canudas de Wit, C., Moulin, P., and Di Domenico, D. (2013). Eco-driving in urban traffic networks using traffic signal information. In *2013 IEEE 52nd Annual Conference on Decision and Control (CDC)*, 892–898. doi:10.1109/CDC.2013.6759995.

Dib, W., Chasse, A., Moulin, P., Sciarretta, A., and Corde, G. (2014). Optimal energy management for an electric vehicle in eco-driving applications. *Control Engineering Practice*, 29, 299–307.

Dollar, R.A. and Vahidi, A. (2018). Efficient and collision-free anticipative cruise control in randomly mixed strings. *IEEE Transactions on Intelligent Vehicles*, 3(4), 439–452. doi:10.1109/TIV.2018.2873895.

Guzzella, L. and Sciarretta, A. (2007). *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*. Springer, Berlin, 2nd edition.

Han, J., Karbowski, D., and Kim, N. (2020a). Closed-form solutions for a real-time energy-optimal and collision-free speed planning with limited information. To be presented at 2020 American Control Conference, Denver, CO, USA.

Han, J., Karbowski, D., Kim, N., and Rousseau, A. (2020b). Human driver modeling based on analytical optimal solutions: Stopping behaviors at the intersections. *ASME Letters in Dynamic Systems and Control*, 1(1). doi:10.1115/1.4046575.

Han, J., Sciarretta, A., Ojeda, L.L., De Nunzio, G., and Thibault, L. (2018). Safe- and eco-driving control for connected and automated electric vehicles using analytical state-constrained optimal solution. *IEEE Transactions on Intelligent Vehicles*. doi: 10.1109/TIV.2018.2804162.

Ioannou, P. and Chien, C. (1993). Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4), 657–672. doi:10.1109/25.260745.

Khalik, Z., Padilla, G., Romijn, T., and Donkers, M. (2018). Vehicle energy management with ecodriving: A sequential quadratic programming approach with dual decomposition. In *2018 Annual American Control Conference (ACC)*, 4002–4007. IEEE, Milwaukee, WI, USA. doi:10.23919/ACC.2018.8431544.

Kim, N., Karbowski, D., and Rousseau, A. (2018). A modeling framework for connectivity and automation co-simulation. In *WCX: SAE World Congress Experience*, 1–12. SAE International, Detroit, MI, USA.

Passenberg, B., Kock, P., and Stursberg, O. (2009). Combined time and fuel optimal driving of trucks based on a hybrid model. In *2009 European Control Conference (ECC)*, 4955–4960. IEEE, Budapest.

Schwickart, T., Voos, H., Hadji-Minaglou, J., and Darouach, M. (2014). A novel model-predictive cruise controller for electric vehicles and energy-efficient driving. In *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 1067–1072. IEEE, Besancon, France. doi:10.1109/AIM.2014.6878222.

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

Shen, D., Karbowski, D., and Rousseau, A. (2018a). Fuel efficient speed optimization for real-world highway cruising. In *WCX: SAE World Congress Experience*, 1–12. SAE International, Detroit, MI, USA.

Shen, D., Karbowski, D., and Rousseau, A. (2018b). Fuel-optimal periodic control of passenger cars in cruise based on Pontryagin's minimum principle. *IFAC-PapersOnLine*, 51(31), 813–820. doi:10.1016/j.ifacol.2018.10.123.

Shen, D., Karbowski, D., and Rousseau, A. (2018c). Highway eco-driving of an electric vehicle based on minimum principle. In *2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 1–8. IEEE, Chicago, IL, USA.

Treiber, M. and Kesting, A. (2013). *Traffic Flow Dynamics: Data, Models and Simulation*. Springer, Heidelberg, 1st edition.

Vahidi, A. and Sciarretta, A. (2018). Energy saving potentials of connected and automated vehicles. *Transportation Research Part C: Emerging Technologies*, 95, 822–843. doi:10.1016/j.trc.2018.09.001.

Xu, S., Li, S.E., Deng, K., Li, S., and Cheng, B. (2015). A unified pseudospectral computational framework for optimal control of road vehicles. *IEEE/ASME Transactions on Mechatronics*, 20(4), 1499–1510. doi: 10.1109/TMECH.2014.2360613.