Look-ahead controls of heavy duty trucks on open roads - six benchmark solutions

Lars Eriksson, Andreas Thomasson, Kristoffer Ekberg, Alberto Reig, Mark Eifert, Fabrizio Donatantonio, Antonio DAmato, Ivan Arsie, Cesare Pianese, Pavel Otta, Manne Held, Ulrich Voegelé and Christian Endisch

The self-archived postprint version of this journal article is available at Linköping University Institutional Repository (DiVA):

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-154674>

N.B.: When citing this work, cite the original publication.

Eriksson, L., Thomasson, A., Ekberg, K., Reig, A., Eifert, M., Donatantonio, F., DAmato, A., Arsie, I., Pianese, C., Otta, P., Held, M., Voegelé, U., Endisch, C., (2019), Look-ahead controls of heavy duty trucks on open roads - six benchmark solutions, *Control Engineering Practice*, 83, 45-66.
<https://doi.org/10.1016/j.conengprac.2018.10.014>

Original publication available at:

<https://doi.org/10.1016/j.conengprac.2018.10.014>

Copyright: Elsevier

<http://www.elsevier.com/>

Look-Ahead Controls of Heavy Duty Trucks on Open Roads - Six Benchmark Solutions

Lars Eriksson^a, Andreas Thomasson^a, Kristoffer Ekberg^a, Alberto Reig^b, Mark Eifert^c, Fabrizio Donatantonio, Antonio D'Amato, Ivan Arsie, Cesare Pianese^d, Pavel Otta^e, Manne Held^f, Ulrich Vögele, Christian Endisch^g

^a *Vehicular Systems, Dept. Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden (e-mail: [lars.eriksson/andreas.thomasson/kristoffer.ekberg]@liu.se).*

^b *Universidad Politécnica de Valencia, 46022 Valencia, Spain (alreiber@mot.upv.es).*

^c *Ford Research and Innovation Center, Süsterfeldstraße 200, 52072 Aachen, Germany (e-mail: meifert@ford.com)*

^d *Energy and Propulsion Systems Laboratory - Dept. of Industrial Engineering - University of Salerno Via Giovanni Paolo II 132, 84084 Fisciano (SA), Italy (e-mail: fadonatantonio@unisa.it)*

^e *Dept. Control Engineering, Czech Technical University in Prague, Czech Republic (e-mail: ottapav1@fel.cvut.cz).*

^f *Department of Automatic Control, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: manneh@kth.se).*

^g *Technische Hochschule Ingolstadt, Germany (e-mail: ulrich.voegel@thi.de).*

Abstract

A benchmark problem for fuel efficient control of a truck on a given road profile has been formulated and solved. Six different solution strategies utilizing varying degrees of off-line and on-line computations are described and compared. A vehicle model is used to benchmark the solutions on different driving missions. The vehicle model was presented at the IFAC AAC 2016 symposium and is compiled from model components validated in previous research projects. The driving scenario is provided as a road slope profile and a desired trip time. The problem to solve is a combination of engine-, driveline- and vehicle-control while fulfilling demands on emissions, driving time, legislative speed, and engine protections. The strength of this publication is the collection of all six different solutions in one paper. This paper is intended to provide a starting point for practicing engineers or researchers who work with optimal and/or model based vehicle control.

Keywords: Engine model, Driveline model, Vehicle model, Control design

---

1. Introduction

Today's truck manufacturers have to continuously improve fuel economy and reduce pollutant emissions to meet both customer and emission legislation demands. With the possibility to use GPS data and route information together with increasing computational power, optimal control could be a powerful tool to meet these demands. As an aid for control development in this area, a benchmark problem was developed and provided to the community prior to the Advances in Automotive Control (AAC) Symposium 2016. The benchmark contains a validated vehicle model connected to a configurable controller where the user can implement and evaluate their own control strategies on a driving mission. The task is to find the fuel optimal controller for driving the truck on a given route with a known topography, while meeting emission legislations. The benchmark includes the possibility to perform pre-computations based on the known driving mission and vehicle specifications and to calibrate the cruise controller.

During AAC 2016 six participants provided their solutions to the benchmark problem. These solutions are collected and compared in this paper according to the criteria that are provided in the benchmark description Eriksson et al. (2016).

1.1. Contributions

The main contribution of this paper is the comparison of the solutions to the complex optimal control problem posed as a benchmark problem at the AAC2016 Symposium, and the conclusions that can be drawn from that. The paper provides a compilation of the six different solutions that were submitted to the benchmark, specified in Eriksson et al. (2016). This can serve as a starting point for practicing engineers or researchers who work with optimal and/or model based vehicle control. Even though there is a single problem formulation the provided solutions show a big variation in solution approaches, with varying degrees of off- or on-line computations, as well as other strategies to determine and control the vehicle actuators.

1.2. Content of paper

The paper has the following outline. Section 3 describes the vehicle model in detail and Section 5 summarizes the optimal control problem formulation. In the subsequent Sections 6 to 11 the solutions introduced in the list below are described in more detail.

1. The proposed approach is based on an off-line solution to the optimal control problem. Optimization is performed with a direct collocation method to transcribe

the optimal control problem into a non-linear programming problem. The optimal solution is applied to the vehicle with a closed-loop controller.

2. A set of controllers were developed as a design exercise for the vehicle cruise control and lower-level engine control. The controllers were designed and calibrated to minimize fuel consumption while satisfying the constraints defined in Section 5 and expressed in (4d)-(4h). They define a minimum average speed (4h), a maximum instantaneous speed of 91 km/h (4g), a maximum turbo speed (4f), and a limitation to the smoke level in the exhaust (4e). Control of the transmission was not considered in the exercise.
3. The solution proposed consists of a two-level control structure. The higher level, using road data, engine and vehicle parameters, computes the optimal speed and gearing profiles through an off-line dual-stage dynamic programming algorithm. The lower level, operating in real-time, computes suitable powertrain control signals assuring the tracking of both reference speed and gear trajectories, while respecting the constraints imposed on engine operation. A parametric analysis on the engine model is provisionally needed in order to obtain the engine steady-state maps used in both levels.
4. The basic idea is to highly simplify the problem and optimize it off-line in a reasonable time. To do so, the off-line optimization is even split into two sub-problems. The first one must be solved once for each new type of vehicle (or load). The other one must be solved once for each new vehicle route. Then the sub-optimal solution is used as additional source of information in an on-line diminishing horizon control in the real-time controller.
5. The main idea of this solution is to maximize the distance during which the vehicle has neutral gear engaged, since using the neutral gear does not consume any fuel. This is done by an offline algorithm that uses the road elevation data in order to identify start and end positions for the use of neutral gear during downslopes. These positions and a speed trajectory is then used on-line during the driving mission by a simple controller.
6. An online approximate multi-criteria dynamic programming controller that iteratively approximates pareto-optimal solutions for time and fuel consumption over a given prediction horizon.

In Section 12 the different solutions are compared and discussed, followed by conclusions and discussion about future activities in Section 13 and 14.

2. Related research

Fuel optimal control of road vehicles have received increasing interest in the last decades in the strive towards lower fuel consumption and lower emissions. An early work

by Schwarzkopf and Leipnik (1977) formulates a minimal fuel-problem and provides explicit solutions for constant road slopes. More recent publications study specific road profiles include Chang and Morlok (2005); Fröberg et al. (2006); Fröberg and Nielsen (2007). For general road profiles different numerical techniques have been adopted based on dynamic programming. Early works include Hooker et al. (1983); Hooker (1988); Monastyrsky and Golownykh (1993) for passenger cars, and work focusing on heavy trucks are found in for example Lattemann et al. (2004); Neiss et al. (2004); Terwen et al. (2004); Hellström et al. (2006). These consider cruise-control by adding a quadratic penalty on deviations from cruise-speed rather than strictly fuel optimal control given a certain road segment and desired trip time. Further development by including trip time in the objective is adopted in Huang et al. (2008); Passenberg et al. (2009); Hellström et al. (2009, 2010a). To manage computation time in an on-board system the latter papers also employs receding horizon control (RHC), where a finite horizon problem is solved in each time step of the controller considering only a given time interval ahead. For the RHC method it is crucial how the horizon length and residual cost are selected, a problem treated in Hellström et al. (2010b).

3. Vehicle Model

The complete vehicle model contains five modules: Engine, Clutch and Transmission, Transmission Control Unit (TCU), Chassis and Road Slope. The inputs and outputs to the different subsystems are shown in Table 1, and the individual subsystems will be described summarily in the sections that follow. The foundation for the vehicle model are results from previous joint research projects where the scientific results are collected in PhD theses and papers. The engine model comes from Wahlström (2009) while the vehicle model is extracted from Myklebust (2014) and the clutch with its lock and release logic comes from the paper Eriksson (2001). This vehicle model was published in the benchmark description Eriksson et al. (2016), in Section 4, and a shorter description is republished here for completeness of the paper.

3.1. Engine

The Diesel engine model with EGR and VGT is based on the open source model LiU-Diesel developed in Wahlström and Eriksson (2011) that describes a 12.7 liter, 6-cylinder diesel truck engine with EGR and VGT. It is tuned to experimental data from Scania and validated in the above mentioned paper. The model contains five subsystems: Cylinders, Turbo (Turbine & Compressor), Intake manifold, Exhaust manifold, and EGR system. The inputs and outputs to the engine model are summarized in Table 1. Most variables are directly from the above mentioned paper with the addition of m_f which is the integrated fuel flow giving the total consumed fuel up to the given time. Inside

Table 1: Inputs and outputs to the different submodels in the vehicle.

	Input	unit	Output	unit
Engine	u_{vgt} VGT opening	0-100%	M_e Engine Torque	Nm
	u_{egr} EGR valve	0-100%	m_f Total fuel mass	kg
	u_δ Fuel injection	mg/stroke	λ Air fuel ratio	-
	n_e Engine speed	RPM	p_{im} Intake pressure	Pa
			x_{egr} EGR fraction	-
			$\dot{m}_c$ Compressor flow	kg/s
Transmission			n_t Turbo speed	rpm
	M_e Engine Torque	Nm	w_e Engine Speed	rad/s
	M_w Wheel Torque	Nm	w_w Wheel Speed	rad/s
	J_e Engine Inertia	kg m ²		
	J_w Vehicle Inertia	kg m ²		
	u_g Current gear	-		
TCU	u_{cl} Clutch Position	0-1		
	t Requested gear time	s		
Chassis	u_b Brake control	%	J_w Vehicle inertia	kg m ²
	w_w Wheel speed	rad/s	M_w Wheel Torque	Nm
	α Road slope	rad	v Vehicle velocity	km/h
			s Traveled distance	m

the cylinders there is an internal fuel limiter, calibrated so that it approximates the torque characteristics of the 400 hp engine. See Fig. 1 for torque and power characteristics together with the Specific Fuel Consumption (SFC) characteristics.

The engine model contains pressure and oxygen fraction states for the intake and exhaust manifold pressures that come from the mass balance and isothermal model in Eriksson and Nielsen (2014)

$$\frac{d}{dt} p_{im} = \frac{R_a T_{im}}{V_{im}} (\dot{m}_c + \dot{m}_{egr} - \dot{m}_{ei}) \quad (1a)$$

$$\frac{d}{dt} p_{em} = \frac{R_e T_{em}}{V_{em}} (\dot{m}_{eo} - \dot{m}_t - \dot{m}_{egr}) \quad (1b)$$

$$\frac{d}{dt} X_{Oim} = \frac{R_a T_{im}}{p_{im} V_{im}} ((X_{Oem} - X_{Oim}) \dot{m}_{egr} + (X_{Oc} - X_{Oim}) \dot{m}_c) \quad (1c)$$

$$\frac{d}{dt} X_{Oem} = \frac{R_e T_{em}}{p_{em} V_{em}} (X_{Oe} - X_{Oem}) \dot{m}_{eo} \quad (1d)$$

$$\frac{d}{dt} w_t = \frac{1}{J_{tc} w_t} (\dot{W}_t \eta_m - \dot{W}_c) \quad (1e)$$

where there are submodels for the different mass flows ($\dot{m}_x$). The states for the oxygen fraction in the intake and exhaust manifold (X_{Oim} and X_{Oem}) become important if EGR is used, since they track the fresh and burned gas fractions in the system. The fifth state is the turbo shaft speed w_t , that is derived from Newton's second law where J_{tc} is the turbo inertia, $\dot{W}_t$ is the power delivered by the turbine, $\dot{W}_c$ is the power required to drive the compressor, and η_m is the mechanical efficiency in the turbocharger. More information about the flow and turbo submodels is found in Wahlström and Eriksson (2011).

The oxygen to fuel ratio λ_O in the cylinder is defined as

$$\lambda_O = \frac{\dot{m}_{ei} X_{Oim}}{\dot{m}_f (O/F)_s} \quad (2)$$

Figure 1: The torque and power characteristics of the diesel engine model together with contours for the SFC. The SFC artifacts at low loads are from data grid generation and plotting.

where $(O/F)_s$ is the stoichiometric relation between the oxygen and fuel masses. The oxygen to fuel ratio is equivalent to the normalized air fuel ratio λ and it is an important constraint for the controller to avoid smoke generation in the benchmark.

3.2. Transmission with Clutch

The clutch and transmission are modeled as an ideal transmission with a stiff driveline, and a friction clutch in between the two rotating systems. This system enables the engine to interact with the vehicle. For example during clutch slip or decoupling it can allow them to rotate independently or at clutch lock-up be locked together. Therefore the clutch contains the states for the angular velocities of the engine and wheel. The gear box and final drive are modeled as rigid, friction free, and mass less elements that are lumped together into one set of gears. The cogwheels in the transmission change the torque and rotational speed. The gear ratio connects the incoming and outgoing rotational velocities and torques. In the Simulink implementation the system is simulated using the two state system with two versions of the clutch transmitted torque depending on if the clutch is locked or slipping. See Eriksson (2001) for more details about the lock-up and break apart logic.

3.3. Transmission Control Unit (TCU)

The transmission control unit (TCU) is a state machine that takes a commanded gear from the driver or cruise controller and performs a gearshift event that is inspired by Ivarsson et al. (2010). The procedure is illustrated in Fig. 2 and has the following events: when a gear shift is commanded the clutch and engine torque (fuel injection) are ramped down from the desired torque to 0 for 0.5s, then a speed synchronization phase is assumed to last 1s. Finally the clutch and fuel injection are ramped up again for 0.5s. In the current model implementation there is no speed synchronization implemented, fuel is only cut-off over the period.

3.4. Chassis

The vehicle chassis model contains the three submodels, air drag, rolling resistance, and road slope and interacts with the transmission and clutch model.

Figure 2: Sketch of a profile for clutch and fuel injection during a gear change. The total time for the gear change is 2 seconds. In the model there is no speed synchronization so the fuel injection is kept at 0 from 0.5 s to 1.5 s after a commanded gear change.

The wheel is assumed to follow the rolling condition meaning that the vehicle velocity and wheel speed are connected algebraically, i.e. $v = r_w w_w$. The basic model is the force/torque balance in Newtons second law that with the rolling condition is translated to an equivalent rotating system. The forces acting on the vehicle are:

$$F_w = F_r + F_a + F_g \quad (3a)$$

$$F_r = m g \cos(\alpha)(c_{r,0} + c_{r,1} v^2) \quad (3b)$$

$$F_a = \frac{1}{2} c_w A \rho v^2 \quad (3c)$$

$$F_g = m g \sin(\alpha) \quad (3d)$$

The vehicle parameters are extracted from Myklebust and Eriksson (2013) where a vehicle model is validated using driving data. Through the rolling condition these can be translated to a wheel torque. Newtons second law for the vehicle acceleration and deceleration with the torque balance and its state for the equivalent rotating system is placed in the clutch model. The clutch provides the wheel speed for the Chassis model and the Chassis returns the loading torque at the wheels.

4. Benchmark Scenario

The scenario which is used to benchmark the solutions in this publication is displayed in Fig. 3. The route is a highway section between the cities Södertälje and Norrköping in Sweden, which contains both steep uphill and downhill slopes. Only longitudinal dynamics is considered, the curvature of the road is not taken into account in the benchmark. One criteria for deciding if a solution is valid or not, is the total trip time. The solutions are demanded to complete the driving mission within a certain time, which sets the minimum requirement of $v_{des} = 80$ km/h. The truck mass is 40 000 kg, which is neither empty (7 000 kg) nor fully loaded (60 000 kg). In the AAC Benchmark other road profiles, unknown to the participants, were tested to evaluate the robustness of the solutions.

Figure 3: The height curve from a highway section between Södertälje and Norrköping.

5. Optimal Control Formulation

The task is to control the main vehicle actuators $\mathbf{u}$ (gear request $u_{g,r}$ -, fuel injection u_δ mg/stroke, variable geometry turbine u_{vgt} %, egr valve u_{egr} %, brake u_b %), provided a driving mission, v_{start} , $\alpha(s)$, v_{des} , m , s_{tot} and vehicle sensors. It is required that the average speed over the distance is the same or higher than the desired average speed v_{des} , while keeping legal limits for speed $v \leq 91$ km/h, smoke generation low $\lambda \geq 1.3$ and the turbocharger healthy $n_t \leq 110\,000$ RPM.

$$\min \int_0^T \dot{m}_f dt \quad (4a)$$

$$s.t. \quad \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) \quad (4b)$$

$$\mathbf{x}(0) = \mathbf{x}_0 \quad (4c)$$

$$\mathbf{u}_{min} \leq \mathbf{u} \leq \mathbf{u}_{max} \quad (4d)$$

$$\lambda(\mathbf{x}, \mathbf{u}) \geq 1.3 \quad (4e)$$

$$n_t \leq 110000 \text{ rpm} \quad (4f)$$

$$v \leq 91 \text{ km/h} \quad (4g)$$

$$s(T)/T \geq v_{des} \quad (4h)$$

The problem formulation shows that the fuel consumption is the main criterion but the constraints are important for emissions and safe operation of the engine and vehicle. The solution can be a mix of both on-line computations that executes and does the optimization in real-time, and cloud solutions that analyzes the problem and downloads a controller calibration that a more light-weight controller follows in real-time. In the evaluation of the solutions, fuel consumption, scenario robustness, together with the cloud and on-line computations are considered.

6. Solution, first participant

Author: Alberto Reig

Affiliation: Universidad Politcnica de Valencia, 46022 Valencia, Spain.

Contact: alreiber@mot.upv.es

This approach to the benchmark problem is based on solving the optimal control problem (OCP) formulated in Section 5 with optimal control theory methods. The vehicle model described in Section 3 (and in Eriksson et al. (2016)) is used as a control-oriented model, with some minor simplifications due to specific requisites of the optimization method and computational limitations.

A direct method (DM) is used to find the optimal solution to the OCP. The reason of this choice is based on the capabilities of this family of methods for handling large non-linear OCPs with a high number of actuators and states with a reasonable computation time Betts (2010).

The optimal trajectories of the actuators are calculated off-line with the DM. These trajectories are initially applied to the engine as a baseline solution. However, as the control model (with minor simplifications) and the simulation models are slightly different, an on-line monitor must be placed to avoid any constraint violation during the mission.

6.1. Optimization method

The OCP is reformulated in distance domain ξ since the disturbance –track slope– is based in location:

$$\min \int_0^\Xi \tilde{m}_f d\xi \quad (5a)$$

$$s.t. \quad \frac{\partial}{\partial \xi} \mathbf{x} = \tilde{f}(\mathbf{x}, \mathbf{u}) \quad (5b)$$

$$\int_0^\Xi 1/v d\xi \leq \Xi/v_{des} \quad (5c)$$

where $\tilde{m}_f = m_f/v$ and $\tilde{f} = f/v$ since $v = d\xi/dt$, and $\Xi = \int_0^T v dt$. Constraints (4d) to (4g) remain the same.

The applied DM consists in discretizing OCP’s ordinary differential equations (ODE) with a collocation method, then transcribing the problem into an associated large but sparse non-linear programming (NLP) problem and finally solving this last problem with one of the numerous NLP algorithms as described in Betts (2010).

The chosen collocation scheme is the Euler method because a first order method offers the highest sparsity grade in the NLP problem –a sparse NLP requires less memory and computational power–. The main drawback of Euler method is its accuracy; however, no significant differences lie between Euler and Runge-Kutta methods within a small interval. Therefore, with the Euler method, states and actuators are discretized as:

$$\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N, \mathbf{x}_{N+1}] \quad (6a)$$

$$\mathbf{u} = [\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1}, \mathbf{u}_N] \quad (6b)$$

and ODEs are approximated as:

$$\tilde{f}\left(\frac{\mathbf{x}_i + \mathbf{x}_{i+1}}{2}, \mathbf{u}_i\right) = \frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta\xi} \quad (7)$$

which are no longer ODEs but algebraic constraints relating contiguous terms of states and actuators. The cost index is also approximated as:

$$\min \left\{ \sum_{i=0}^N \tilde{m}_f\left(\frac{\mathbf{x}_i + \mathbf{x}_{i+1}}{2}, \mathbf{u}_i\right) \Delta\xi \right\} \quad (8)$$

The transcription of the constraints is:

$$\mathbf{x}_0 = x_0 \quad (9a)$$

$$\mathbf{u}_{\min} \leq \mathbf{u}_i \leq \mathbf{u}_{\max} \quad (9b)$$

$$\lambda(\mathbf{x}, \mathbf{u}) \geq 1.3 \quad (9c)$$

$$\mathbf{x}_{\min} \leq \mathbf{x}_i \leq \mathbf{x}_{\max} \quad (9d)$$

$$\sum_{i=0}^N 1/v \Delta\xi \leq \Xi/v_{des} \quad (9e)$$

Note that constraints (4f) and (4g) are synthesized in (9d), since both turbocharger speed and vehicle speed are states of the problem.

Therefore, the associated NLP problem defined in (7), (8) and (9) can be constructed from the original OCP, where dynamics of the model have been substituted with $N \cdot N_x$ algebraic constraints, where N_x is the number of states of the problem. This new problem is no longer a dynamic problem but a static one that can be solved with any algorithm for NLP. The resulting problem is larger than the original OCP but it is also extremely sparse.

To address the NLP problem the open source IPOPT solver (see Wchter and Biegler (2006)) has been used. Its algorithm is based on the interior point method and is especially efficient with large and sparse problems such as the current one. Cost index, constraints, gradient and Jacobian are given to the solver. The structure of the Jacobian is shown in Fig. 4 as a reference of the level of sparsity of the problem.

The use of the gradient and Jacobian function to solve the NLP problem requires a continuous and differentiable model. This requisite somehow limits the applicability of the method when facing a discrete model. This is the case of the vehicle’s gearbox, whose setting is inherently an integer variable. To handle this situation, and avoid a heavy mixed-integer NLP problem, the *integer constraint* is relaxed, i.e. the fact that a variable is integer is ignored and it is assumed as a continuous quantity. While this may impact the quality of the solution (it has not been further investigated), the high number of different gear ratios (14 gears) makes the discrete quantity $u_{g,r}$ close to a continuous variable. Hence, the NLP has been divided in three sequential optimizations:

1. Optimize gearbox, brakes and speed with a simplified engine (equivalent engine map) and *integer constraint* relaxation. The resulting gearbox setting is later rounded to integer values.

Figure 4: Non-zero elements of the Jacobian matrix; the number of non-zero elements is $N_{nz} = 855799$. Gray dotted lines divide the matrix in quadrants for individuals state, actuator or original OCP constraint and ODE, which are each made of N algebraic constraints. Red lines separate the different differentials in the Jacobian.

Table 2: Variables that are assumed as states, actuators and disturbances and the type of model used at each optimization step that the original OCP is divided into. Note that $\bar{u}_{g,r}$ is the continuous version of $u_{g,r}$.

Step	States	Actuators	Disturbances	Model
#1	v	$\dot{m}_f, u_b, \bar{u}_{g,r}$	α	engine map, detailed vehicle
#2	v	$\dot{m}_f, u_b$	$\alpha, u_{g,r}$	engine map, detailed vehicle
#3	$v, p_{im}, X_{Oim}, p_{em}, X_{Oem}, w_t$	$\dot{m}_f, u_{egr}, u_{vgt}$	$\alpha, u_{g,r}$	detailed engine and vehicle

- For the above given integer gearbox trajectory, brakes and speed are optimized with the simplified engine.
- With the previous gearbox setting and brakes trajectories, the engine actuators and vehicle speed are optimized with the complete and detailed model.

Table 2 summarizes in detail the variables that are assumed as actuators, states and disturbances at each step.

6.2. Model simplifications

Most of the model described in Section 3 has been used as the control-oriented model to address the OCP. However, the construction of the gradient and the Jacobian requires the model to be continuous and differentiable. Despite most of the original engine and vehicle models are differentiable, some sub-systems are represented with piecewise functions, saturations and discrete quantities. To fulfill the requisites

of the optimization method, all those discontinuities are approximated to analytic functions. In addition, several sub-models are simplified due to its high complexity and low impact on the behaviour of the complete vehicle.

6.2.1. Piecewise functions

Piecewise functions (or saturations) are represented analytically with a Heaviside step function approximation. Given a piecewise function defined as:

$$p(x) = \begin{cases} p_1 & x \leq \tilde{p} \\ p_2 & x > \tilde{p} \end{cases} \quad (10)$$

it may be approximated as:

$$p \approx (1 - H)p_1 + Hp_2 \quad (11)$$

with H an analytic approximation to the Heaviside function:

$$H = \frac{1}{2} + \frac{1}{2} \text{erf}(k(x - \tilde{p})) \quad (12)$$

6.2.2. Turbine c_m parameter

The turbine efficiency is modeled as a parabolic function of the blade speed ratio with a parameter c_m which is a piecewise function:

$$c_m = \begin{cases} c_{m1}(w_t - c_{m2})^{c_{m3}} & w_t \geq w_{t,lin} \\ \frac{c_{m,lin}(w_t - c_{m2})}{w_{t,lin} - c_{m2}} & w_t < w_{t,lin} \end{cases} \quad (13)$$

The following expression has been used as an analytic approximation replacing the above:

$$c_m \approx (c_{ma} + c_{mb} \exp(c_{mc}(w_t - c_{m2}))) \left(\frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{w_t - w_{t,lin} + c_{md}}{c_{md}} \right) \right) \quad (14)$$

The average approximation error is less than 0.13% for the complete operation range.

6.2.3. Fuel limiter

The internal fuel limiter in cylinder model is a piecewise function with 32 linear terms. In order to avoid such a non-differentiable function, the following analytic expression based on a Fourier approximation is used:

$$\begin{aligned} \dot{m}_{f,lim} = & c_{f1} \sin(c_{f2} n_e + c_{f3}) + c_{f4} \sin(c_{f5} n_e + c_{f6}) \\ & + c_{f7} \sin(c_{f8} n_e + c_{f9}) + c_{f10} \sin(c_{f11} n_e + c_{f12}) \\ & + c_{f13} \sin(c_{f14} n_e + c_{f15}) \end{aligned} \quad (15)$$

The average approximation error is less than 0.57% for the complete engine speed range.

6.2.4. EGR and VGT actuator dynamics

EGR and VGT actuators are originally modeled as a first order system with a pure delay. This involves additional states to the OCP with an increase in problem complexity. Due to the fast setting time (< 0.1 seconds) the dynamics of the actuators are considered negligible, and therefore they are not included in the control-oriented model.

6.2.5. Clutch

The clutch model in the benchmark model is complex and involves a significant number of states with non-differentiable equations. However, the clutching event lasts only two seconds (gear shiftings at the optimal solution are about 15, which means a 0.5% of the route) and is driven by an external controller so the optimal control only triggers the gear shift. Therefore, the clutch model has been replaced in the control-oriented model for a simple switch in the fueling rate: after a gear shift the fueling rate is set to zero during two seconds.

6.3. On-line application

The on-line implementation of the proposed optimal control strategy combines open-loop and closed-loop controllers: on one hand the actuators optimal trajectories are used as set-points for the control; on the other hand, several closed-loop safeguards are placed to avoid any eventual constraint violation and to provide a robust strategy.

- Speed limiter: cuts off the fuel injection as the vehicle speed approaches a given threshold (it was set to about 1 km/h below the road limit, but this parameter may be calibrated to other values).
- Distance closed-loop controller: a PID controller runs over the open-loop control ensuring that the optimal trajectory for the covered distance is fulfilled—to be on time at the destination—.
- Smoke limiter: a simple estimator predicts the maximum fuel injection to reach $\lambda = 1.3$ based on the available sensors (n_e , p_{im} , $\dot{m}_c$, x_{egr} and λ). The set-point fueling rate is saturated to this limit to avoid smoke generation.
- Emergency brakes: if vehicle speed reaches a threshold speed (in this case slightly less than 1 km/h), the necessary braking effort to keep constant speed is calculated with an inverted vehicle dynamics model and applied to the truck.

These safeguards are seldom active thanks to the high accuracy of the control-oriented model; the speed limiter actuates during 1.4% of the route, the distance controller during 3%, the smoke limiter during 1%, and emergency brakes are not used during the optimal control strategy.

6.4. Results

The application of the proposed optimal control strategy resulted in the speed and actuators trajectory shown in Fig. 23 where the results from the complete driving mission is displayed. Vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration. The OCP constraints are fulfilled along the complete route. Main values may be found in Table 3. The total fuel consumption over the test route is 33.48 kg (35.02 L/100 km).

Table 3: Main results of the proposed control strategy in the benchmark route.

Fuel consumption	33.48 kg
Average speed	80.07 > 80 km/h
Min λ	1.32 > 1.3
Max speed	90.85 < 91 km/h
Max turbo speed	88252 < 110000 rpm
Algorithm CPU time	851 s

It may be noted that both EGR and VGT actuator positions move around a tight range with few exceptions. This might be caused by a strong optimum pole around $u_{egr} = 45\%$ and $u_{vgt} = 40\%$ regardless of the engine operating point. However, they slightly move around those values with a policy that cannot be represented as a fixed calibration, see the bottom of Fig. 23 where no clear trend can be distinguished in actuators position.

Regarding gear number, it quickly shifts to the highest gear –14th–, holding until facing a very steep slope, where downshifting is necessary to keep the required speed (this is displayed in Fig. 23). Brakes are only actuated at the very end of the route where a steep downhill may overspeed the truck. The fueling rate follows the torque demand to reproduce the optimal speed trajectory.

There is also an interesting relation between the track altitude and the truck speed, maxima in speed correspond to minima in altitude. This is due to three different factors: first, the optimal solution transfers kinetic and potential energy minimizing the contribution of fuel to that process; second, depending on the grade of the slope, the truck might be unable to keep the speed –this situation is more pronounced as the vehicle mass increases–; and third, since the use of the brakes in a downhill jeopardizes the efficiency, the speed on top of a hill must be low enough to gain speed without overspeeding. Despite this behavior looks like a good rule to decide the vehicle speed at each hill, there is not a univocal relation between altitude and speed, so it is a question of local energy balance at each single hill.

6.5. Conclusions

An optimal control strategy as been proposed based on the solution of the OCP with a direct method. Due to the requirements of this method, several simplifications to the vehicle model have been performed, mainly to produce a fully continuous and differentiable model. The OCP has been splitted in several sequential optimizations to deal with the discrete nature of the gearbox which may impact on the optimality of the solution.

The optimal solution to the problem offers the optimal trajectories of the actuators. Despite the accuracy of the control-oriented model, an open-loop controller imposing these trajectories lacks of robustness. Therefore, several post-optimization closed-loop controllers have been added to work with the optimal trajectories as set-points to guarantee that constraints will be fulfilled and optimal

trajectories followed; however, these controllers are seldom active.

The optimal control strategy shows several interesting behaviors, especially the relation between vehicle speed and altitude, that looks difficult to reproduce with a fixed calibration or heuristic controllers.

7. Solution, second participant

Author: Mark Eifert

Affiliation: Ford Research and Innovation Center, Süsterfeldstraße 200, 52072 Aachen, Germany.

Contact: meifert@ford.com

A set of controllers were developed as a design exercise for the vehicle cruise control and lower-level engine control. The controllers were designed and calibrated to minimize fuel consumption while satisfying the constraints defined in Section 5 and expressed in (4d)-(4h). They define a minimum average speed (4h), a maximum instantaneous speed of 91 km/h (4g), a maximum turbo speed (4f), and a limitation to the smoke level in the exhaust (4e). Control of the transmission was not considered in the exercise.

7.1. Predictive Cruise Control

The cruise control consists of a predictive algorithm that determines the speed setpoint and an actuation strategy for the mechanical brakes. Both were conceived to minimize fuel consumption.

7.1.1. Speed Setpoint Control

The range of torques that the engine operates in is dependent on the speed setpoint. For example, choosing a high speed setpoint while the vehicle drives uphill, downhill or on flat terrain will raise the average output torques from the engine in the corresponding situation. Fuel economy over a drive in hilly terrain can be improved if a set of operating points in a given driving situation can be moved such that their efficiency is improved.

A simple speed setpoint control was devised that chooses a calibrated speed setpoint that is dependent on whether the vehicle is traveling uphill, downhill or on level terrain. The controller uses map-based road-slope data to determine the inclination of the vehicle, and that information is used to determine the speed setpoint v_{ref} :

$$v_{\text{ref}} = \begin{cases} UH_{\text{Speed}} & , \text{ slope} \geq 0 \\ DH_{\text{Speed}} & , \text{ otherwise} \end{cases} \quad (16)$$

Here slope is defined as positive when the vehicle is driving uphill. The optimal choice for the uphill and downhill speeds will be described in the calibration section.

7.1.2. Mechanical Brake Controller

The mechanical brake controller operates in parallel with the speed setpoint controller to maintain a vehicle speed under the maximum allowed 91 km/h defined in constraint (18g). It is a proportional controller that applies the brakes after the vehicle speed exceeds a threshold:

$$u_b = \begin{cases} K_b (v - V_{\text{Thresh}}) & , v > V_{\text{Thresh}} \\ 0 & , \text{ otherwise} \end{cases} \quad (17)$$

The threshold speed V_{Thresh} and the proportionality factor K_b were chosen such that the use of the brakes was minimized in order to maximize fuel economy. After some experimentation, a threshold speed of 85 km/h and a proportionality factor K_b of 1800 N hr/km were chosen.

7.2. Predictive Engine Control

The speed setpoint is achieved by a fueling control that determines the fuel mass flow to the engine. The engine's efficiency may be controlled by shifting it's operating point or through the use of the variable geometry turbine (VGT). Both the fueling and VGT controls are constrained by the smoke limit, which limits the fuel equivalence ratio λ to values greater than or equal to 1.3, which guarantees a Bosch smoke rating under 3 as described in Obert (1973). In order to realize this constraint, predictive controls using map-based information about the road ahead are implemented for both fueling and the VGT.

Control of exhaust gas recirculation (EGR) was available as well, but none was used, because NO_x production was not part of the design constraints. Hence u_{egr} was continuously set to zero.

7.2.1. Fueling Control

The fueling control regulates the fuel mass flow using a proportional controller that compares the speed setpoint v_{ref} with the measured vehicle speed v :

$$u_\delta = K_p (v_{\text{ref}} - v) \quad (18)$$

where u_δ is the fuel mass per stroke. The value of K_p was left unchanged from the original benchmark model.

The fuel mass-flow setpoint u_δ is purely a function of the proportional controller as described above if a tip-out of the throttle is not imminent or has not just occurred. However, a feed-forward element is added to the proportional controller in order to meet the engine-smoke constraint if a tip-out is predicted, or if the equivalence ratio λ approaches the critical threshold 1.3.

A tip-out is predicted to occur whenever the vehicle arrives at the top of a hill while the cruise control is activated. The imminent arrival at the top of a hill is detected by an algorithm that compares the slope of the road at the current position with the slope a fixed distance in the future. If the current slope is positive (indicating an uphill drive) and the slope a fixed distance M in the future is negative,

a “Top Of-Hill” condition is identified, and a corresponding flag F_{TOH} is set:

$$F_{TOH} = \begin{cases} 1, & s_0 > 0 \text{ AND } s_M < 0 \\ 0, & \text{otherwise} \end{cases} \quad (19)$$

Here s_0 represents the current road slope and s_M represents the slope a horizon distance M down the road. For the benchmark problem, a horizon distance M of 48 meters was used, however the value may be made variable and dependent on the vehicle speed in an actual series application.

The engine will smoke excessively if the air-fuel mixture becomes too rich. In steady-state operation, this is not a problem, because the p-controller is sufficient to maintain a fuel equivalence ratio $\lambda \gg 1.3$. In tip-out situations where the throttle is quickly closed, the fueling controller may react to a change in air flow too slowly and call for excessive fuel to be injected. For this reason, the fuel mass-flow setpoint u_δ that is calculated from the proportional expression above is reduced by an additional proportional term K_{TOH} if a hill top was identified in the current or last controller calculation loop:

$$u_\delta = K_p K_{TOH} (v_{ref} - v) \quad (20)$$

where $K_{TOH} < 1$.

If the equivalence ratio λ approaches the threshold 1.3, the fuel mass-flow setpoint u_δ is further reduced by a constant term K_λ regardless of the state of the “Top-Of-Hill” indicator. Hence the complete fueling law may be expressed as:

$$u_\delta = \begin{cases} K_p K_{TOH} (v_{ref} - v), & \lambda \geq \lambda_{Critical} \\ K_p K_{TOH} (v_{ref} - v) - K_\lambda, & \text{otherwise} \end{cases} \quad (21)$$

The variable $\lambda_{Critical}$ defines the threshold at which the equivalence ratio is considered critical. A value of 2 was used for this threshold without optimizing it.

7.2.2. Variable Geometry Turbine Control

The setting of the variable geometry turbine (VGT) is also controlled using the “Top-Of-Hill” indicator to allow it to be set as high as possible without excessive smoke being generated during throttle tip-out. When a “Top-Of-Hill” is indicated, a low VGT setting u_{vgt} is selected by the control algorithm. Otherwise, a higher setting is used:

$$u_{vgt} = \begin{cases} VGT_{TOH}, & F_{TOH} = 1 \\ VGT_0, & \text{otherwise} \end{cases} \quad (22)$$

where F_{TOH} designates the state of the “Top-Of-Hill” indicator defined above and VGT_{TOH} and VGT_0 represent the calibrated values for the VGT settings.

7.3. Parameter Calibration

The calibration parameters defining the setpoints for uphill and downhill speed along with the VGT settings represent a compromise between minimizing fuel consumption and maintaining a defined minimum average speed and fuel equivalence ratio λ . Determining the set of parameters that minimize fuel consumption while maintaining the constraints is a nonlinear, nonconvex optimization problem. Nonlinearities in the optimization problem occur especially during tip-out when the vehicle crests a hill or a gear-change occurs. During these situations, the constraints on maximal speed and smoke limit come close to being broken. The speed limit may be robustly controlled by a proportional controller, because the relation between braking force and speed is Lipschitz continuous (limited in how fast speed changes with respect to braking force) as described in Nocedal and Wright (2000). However, that is not the case with the equivalence ratio λ , which may exhibit discontinuities when the throttle closes combined with nonlinearities due to changes in combustion.

Because the constraint on maximum speed could be robustly controlled, the calibrations for the mechanical brake control were considered as system constraints to the optimization problem of choosing the cruise control and VGT control setpoints. It was recognized during initial simulations that the equivalence ratio was discontinuous at each tip-out and sunk to a value in the neighborhood of 1.3. In the route that was provided as part of the benchmark problem, 14 critical points were identified where this occurred. The optimization problem is to identify calibration parameters that do not allow the equivalence ratio to fall below 1.3 in a tip-out given that the equivalence ratio following the previous tip-out did not violate the constraint but was in the neighborhood. The nonlinearities of the system make it necessary for a solution by direct method to first evaluate incremental candidate solution steps between these critical points. During this initial evaluation of candidate calibration parameters, a safe initial state of 1.6 was assumed as the equivalence ratio after a successful tip-out where the constraint was not violated.

The optimization problem can be described mathematically as minimizing the cost function expressed by equation (18a) over a time interval $[0, T]$ such that constraints (18b) through (18h) are realized. The time interval is divided into $M+1$ subintervals with M intermediate points at discontinuities corresponding to tip-in conditions: $[t_1, t_2, \dots, t_M]$. If the safe initial state at the end of each interval is designated as $\lambda(t_i^-)$ and the initial state at the beginning of the next is designated as $\lambda(t_i^+)$, an additional constraint may be added:

$$\lambda(t_i^-) - \lambda(t_i^+) \leq \epsilon \quad (23)$$

where ϵ is chosen small enough such that the smoking constraint (18e) is never violated. A solution to this class of optimization problem may be generated by the direct multiple shooting method that was first described in Bock

and Plitt (1984). Like all direct methods, solutions to the optimization problem are generated incrementally. In the case of this parameter optimization problem, a set of candidate parameters are evaluated in a simulation of the vehicle model over a benchmark route. The resulting value of the cost function is calculated and constraint violations are flagged. The algorithm used for this problem is represented by the following pseudocode:

1. Define interval start times from initial simulation:
 $[t_1, t_2, \dots, t_M]$
2. Define starting Lambda for each interval $\lambda(t_i^+) = 1.6$ and $\epsilon = 0.3$
3. Define initial parameter set:
 $P = [UH_{Speed}, DH_{Speed}, VGT_0, VGT_{TOH}]$
4. Define MinFuel=0; IterationCount=0; MinStep=1
5. Search for optimal parameter set
  - (a) Identify set of valid parameters
    - i. Run simulation over each M+1 interval to check for validity of parameters
    - ii. If violations in Speed(t), $\lambda(t)$ or Average Speed occur, perturb parameter set P and repeat previous step; otherwise goto next step
  - (b) Investigate optimality
    - i. Run simulation over entire driving mission and calculate fuel use over entire driving mission: Fuel(T)
    - ii. Calculate change in fuel use:
 $\Delta Fuel = \text{MinFuel} - \text{Fuel}(T)$
    - iii. If $\Delta Fuel < 0$, define MinFuel=Fuel(T)
    - iv. Identify optimum parameter set
      - A. If $\Delta Fuel < \text{MinStep}$:
Optimal Parameter Set is P
      - B. Otherwise continue search
        - Perturb parameter set P proportionally to $\Delta Fuel$
        - Return to step 5a

The parameter optimization algorithm converged after 115 simulations of the entire benchmark driving mission. The resulting parameters were used to compare this solution against the others. In Fig. 24 the results from the complete driving mission is displayed, where vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration.

7.4. Conclusion

Many facets of the predictive control strategy with optimized calibration were implemented to maintain the fuel equivalence ratio λ above 1.3 in order to limit engine smoking. The strategy did successfully accomplish this in the benchmark driving mission. A mirror image of that driving trajectory along with a third simulated drive were also successfully tested without smoke violations.

However, the fuel consumption that was simulated with this strategy was significantly higher than the other strategies. A significant improvement in fuel consumption may be made by expanding the algorithm to include gear choice.

A further improvement would be made by controlling fueling directly with respect to the efficiency of the operating point of the engine. The calibration process of the current strategy intrinsically compared the group of operating points of the engine while the vehicle was climbing or driving on a flat stretch of road with operating points encountered while the vehicle traveled downhill. The speed setpoint for the set of points with the lowest cost of mechanical work was increased through the calibration process in order to maximize the propulsion energy expended in that region. That allowed the speed setpoint for the set of points with the highest cost of mechanical work to be set lower, which minimized the propulsion energy expended there. Designing a controller that intrinsically appraises the cost of energy of individual operating points and the region around them allows for a more optimal scheduling of energy conversion, and this is reflected in the fuel consumption performance of many of the other control strategies that are presented here.

8. Solution, third participant

Authors:

Fabrizio Donatantonio, Antonio D'Amato, Ivan Arsie, Cesare Pianese.

Affiliation:

Energy and Propulsion Laboratory, Dept. of Industrial Engineering, University of Salerno, Via Giovanni Paolo II 132, 84084 Fisciano (SA), Italy.

Contact:

fadonatantonio@unisa.it

8.1. Introduction

In this solution, velocity reference profile and shifting schedule are computed off-line using elevation data along with vehicle and engine characteristics as shown in Fig. 5. Once the reference speed and gear schedule are computed, they are supplied to cruise, engine and powertrain control, operating online to promptly compute current engine and drivetrain control signals.

The computation of fuel optimal speed profile and gear schedule relies on dynamic programming as already proposed by Arsie et al. (2005), Hellström et al. (2009, 2010a) and Ozatay et al. (2014), but the most significant difference respect to previous works lays in the problem formulation. Rather than directly minimizing the fuel consumption, the original optimal control problem (4a)-(4h) is split in two sub-problems formulated in the space domain and solved in series through dynamic programming with the algorithm by Sundstrom and Guzzella (2009). The first sub-problem, detailed in Section 8.3, minimizes the amount of mechanical

Figure 5: Control hierarchy scheme, in double line, the measured signals from the powertrain.

energy required to accomplish the whole driving mission, subject to the assigned constraints; the output, supplied to the second optimization, is the optimal velocity profile to be followed. The second sub-problem finds the optimal shifting schedule minimizing the fuel consumption and guaranteeing the execution of the velocity profile previously computed. It has to be remarked that constraints on maximum n_T (4f) and minimum λ (4e) have not been taken into account in the higher level. Nevertheless, their fulfillment is assured by the engine control hereby developed and presented in Section 8.5. This methodology, though providing a sub-optimal solution, allows to keep the computational time low, thus minimizing the impact of optimization over trip time. The complete description and implementation of the solution is found in Donatantonio et al. (2018)

8.2. Parametric Analysis

Vehicle and powertrain characteristics are used by both the off-line optimization process (i.e. higher level) and online engine control (i.e. lower level) and thus, have to be known *a priori*. Vehicle characteristics are parameters and thus supposed to be known. Engine data and performance have to be made available either through a provisional parametric analysis to be carried out just once for any engine type or directly by the engine manufacturer. The parametric analysis is formulated as follows:

$$\min_{u_\delta, u_{vgt}, u_{egr}} sfc(n_e, M_e) \quad \forall n_e, M_e \quad (24a)$$

$$\text{where } M_e(n_e, u_\delta, u_{vgt}, u_{egr}) \quad (24b)$$

$$\text{s.t. } \lambda \geq 1.4 \quad (24c)$$

$$n_t \leq 100\,000 \text{ rpm} \quad (24d)$$

As expressed by (24a), the objective is to find the control signals of the engine actuators u_δ , u_{vgt} , and u_{egr} that obtains the lowest specific fuel consumption sfc , in any engine operating point, defined by n_e and M_e , in steady state conditions, while respecting constraints (24c), (24d). For the former and the latter, a slight safety margin has been taken into account respect to (4e) and (4f). This analysis allows to obtain maps for maximum torque curve

$M_{e,\max}(n_e)$, fuel flow $\dot{m}_f(n_e, M_e)$ along with $u_\delta(n_e, M_e)$, $u_{vgt}(n_e, M_e)$, $u_{egr}(n_e, M_e)$ and the corresponding $\lambda(n_e, M_e)$. These data are used in both the off-line optimization and on-line powertrain control.

8.3. Velocity Profile Optimization

With reference to Fig. 5, the first optimization sub-problem consists in minimizing the energy spent to drive along a given path. The optimal control problem, formulated in discrete form, in the space variable s and with kinetic energy K as state variable, can be expressed as follows:

$$\min_{F_{dr,1} \dots F_{dr,N_1}} \sum_{k=1}^{N_1} J_k(F_{dr,k}) \quad (25a)$$

$$J_k = \mu_1 F_{dr,k}^2 + \left| \frac{1}{v_{des}} - \sqrt{\frac{m}{2K_k}} \right| \mu_2 m \quad (25b)$$

$$\text{s.t. } K_{k+1} = K_k + \Delta s_1 [F_{dr,k} + F_{w,k}] \quad (25c)$$

$$K(0) = \frac{1}{2} m v_0^2 \quad (25d)$$

$$\frac{1}{2} m v_{\min}^2 \leq K_k \leq \frac{1}{2} m v_{\max}^2 \quad (25e)$$

$$F_{dr,k} \in \text{linspace}[-20\,000, F_{\max,k}] \quad (25f)$$

For this problem the control variable is the driving force at the wheels, denoted with F_{dr} . The optimal control objective is expressed by (25a) as sum of the N_1 step costs J_k , calculated through (25b), to drive along each one of the N_1 space segments of length Δs_1 into which the whole track length S is discretized. In (25b) μ_1 and μ_2 are constant weights. The resulting optimal control is the sequence of the N_1 control signals $F_{dr,k}$ satisfying (25a). The state equation, is reported in (25c), discretized through the Euler forward method, along with its initial condition (25d). The state K is double bounded by minimum and maximum kinetic energy corresponding to either legal or desired speed limits as in (25e).

$$F_{\max} = \frac{M_{e,\max}(n_e) i_{\text{final}} i_g(u_g)}{r_w} \quad (26a)$$

$$\text{where } u_g \in \Gamma : n_e(K, u_g) \in [400, 2000] \text{ rpm} \quad (26b)$$

$$n_e(K, u_g) = \sqrt{\frac{2K}{m}} \frac{i_g(u_g) i_{\text{final}}}{r_w} \quad (26c)$$

The control signal $F_{dr,k}$ have to belong to the discrete feasible set (25f) double bounded by minimum and maximum driving force values. The former has been set arbitrarily, the latter corresponds to the maximum achievable driving force at current speed (26a), in any gear keeping engine speed n_e in the feasible range as formally described by (26b). The maximum torque value is computed through linear interpolation of maximum engine torque as a function of n_e (26c), obtained as described in Section 8.2. The outputs of this optimization process are the kinetic energy and driving force profiles minimizing (25a) that will be

both used in the second optimization stage. The reference speed profile is then directly derived from the kinetic energy profile obtained.

8.4. Gear Shifting Schedule Optimization

In the second optimization stage, the objective is to find the gear schedule allowing to follow the velocity reference computed in the previous optimization while keeping the engine in the best operating conditions and, thus, with low fuel consumption. This second optimization problem, is formulated as follows:

$$\min_{\gamma_1 \dots \gamma_{N_2}} \sum_{k=1}^{N_2} J'_k(\gamma_k, u_{sh,k}) \quad (27a)$$

$$J'_k = \frac{\dot{m}_f(M_{e,k}, n_{e,k}) \Delta s_2}{v_k} + (\beta_1 + \beta_2) |u_{sh,k}| \quad (27b)$$

$$\text{where} \quad \begin{cases} \beta_2 = 0, & \text{if } \alpha_k \leq 0 \\ \beta_2 = \beta_2, & \text{if } \alpha_k > 0 \end{cases} \quad (27c)$$

$$\text{s.t.} \quad u_{g,k+1} = u_{g,k} + u_{sh,k} \quad (27d)$$

$$u_{sh,k} = [-2; -1; 0; +1; +2] \quad (27e)$$

For this problem, the state variable is the engaged gear u_g , evolving according to (27d) and the control variable is the signed command shifting u_{sh} . The optimal control objective (27a) is the sum of N_2 step costs J'_k (27b). The first term of J'_k is the fuel mass required to drive on the k -th segment of road, calculated by using a steady state map of fuel flow $\dot{m}_f(M_e, n_e)$ as a function of engine torque and speed. Since velocity and driving force profiles are calculated in the previous optimization process, by using (26a) and (26c), it can be seen that the only degree of freedom is the engaged gear u_g . The second term of J'_k penalizes the shifting maneuver that is further penalized on uphill segments through the coefficient β_2 (27c). The admissible set of maneuvers, according to (27e) are single and double up/downshift, along with gear hold. Again, the engine has to be kept within its feasible operating speed range, this entails that condition (26b) still applies.

An accurate shifting schedule shows to have a significant impact on fuel consumption. A second optimization is performed on a finer grid in comparison of the first, with a larger number of steps N_2 . The result of this process is a fuel optimal gear shifting schedule for the reference velocity v_{ref} and driving force F_{dr} profiles calculated in the previous optimization.

8.5. Powertrain Control

This section addresses to the computation of the powertrain control signals. The velocity profile and gear schedule, computed off-line as described in Sections 8.3 and 8.4, play the role of reference trajectories and in the following paragraphs will be denoted as v_{ref} and $u_{g,ref}$ whereas current vehicle speed and engaged gear as v and u_g .

8.5.1. Delay Recovery System

The second term of the cost function J_k (25b) addresses only partially the constraint on minimum average speed (4h). The speed reference therefore requires an on-line correction that consider the objective average speed v_{des} .

$$v_{ref} = v_{ref} + \Delta v \quad (28a)$$

$$\Delta v = K_v \left(v_{des} - \frac{1}{T} \int_0^T v \, dt \right) \quad (28b)$$

In practical terms, the speed reference is raised by a term Δv (28b), proportional, through the constant gain K_v , to the difference between v_{des} and average speed at time T .

8.5.2. Torque Estimation

In order to generate the powertrain control signals, an engine required torque estimate $\widehat{M}_{e,req}$ has to be computed.

$$\widehat{M}_{e,req} = \frac{(F_r + F_a + F_g) r_w}{i_g(u_g) i_{final}} + K_{p,v}(v_{ref} - v) \quad (29)$$

Equation (29) shows how the required torque estimate is made by two components: road load and a term proportional to the residual of current speed respect to the reference one, through the constant gain $K_{p,v}$.

8.5.3. VGT/EGR command

For any engine operating point, defined by engine speed and torque, the steady-state maps built as shown in Section 8.2, provide a command baseline for both u_{egr} and u_{vgt} . The former is used as is to generate the u_{egr} command, the latter is corrected in those engine transients resulting in a value of λ lower than the reference value, obtained through the steady state map $\lambda_{ref}(n_e, M_e)$ as follows.

$$u_{vgt} = \begin{cases} u_{vgt}(n_e, M_e) - \Delta u_{vgt} & \text{with } \lambda < \lambda_{ref} \\ u_{vgt}(n_e, M_e) & \text{with } \lambda \geq \lambda_{ref} \end{cases} \quad (30a)$$

$$\Delta u_{vgt} = K_{p,vgt}(\lambda_{ref} - \lambda) + K_{i,vgt} \int \lambda_{ref} - \lambda \, dt \quad (30b)$$

The correction (30b) consists in a proportional and an integral term, with constant gains $K_{p,vgt}$ and $K_{i,vgt}$ of the residual between current value of λ and its reference. The correction is aimed at slightly closing the VGT during sharp tip-in transients in order to speed up the turbocharger and thus increase air flow.

8.5.4. u_δ command

As for the other engine actuators, the injected fuel mass baseline is supplied by the steady state map $u_\delta(n_e, M_e)$. In order to avoid infringement of constraint on λ , the amount of injected fuel has to be dynamically limited considering the actual engine intake air mass flow. This latter is estimated by applying the filling-emptying at the intake line, from the compressor to the intake valve, whose overall volume is V_{im} , as shown in (31a). Once the $\dot{m}_{ei}$ has been estimated, the maximum fuel flow that is possible

to inject without infringing the constraint on minimum λ can be evaluated and the maximum u_δ command can be computed (31c).

$$\dot{m}_{ei} = \dot{m}_c - \frac{V_{im}}{R_a T_{im}} \frac{dp_{im}}{dt} \quad (31a)$$

$$\dot{m}_{f,max} = \frac{\dot{m}_{ei} X_{Oim}}{\lambda_{min} (A/F)_s X_{Oc}} \quad (31b)$$

$$u_{\delta,max} = \frac{120 \dot{m}_{f,max}}{n_{cyl} n_e} \quad (31c)$$

It has to be remarked that any variable involved in the calculation of $u_{\delta,max}$ is either measured or constant, excluding the intake manifold oxygen concentration X_{Oim} that is unknown. This variable is affected by the effective EGR fraction. It has been observed, for this engine, that operating points where critical λ values are attained have either zero or very low EGR valve opening commands. Therefore, X_{Oim} has been assumed equal to ambient air oxygen mass concentration (i.e. $X_{Oim} = 0.23$).

8.5.5. Freewheeling

To further increase fuel economy, the opportunity of coasting in neutral gear on downhill segments has been explored. This maneuver is known as freewheeling.

$$\frac{(F_r + F_a + F_g) r_w}{i_g(u_{g,ref}) i_{final}} \leq M_{e,FW} \quad (32)$$

This operation mode is treated quite simply, triggering neutral gear use when condition (32) holds, that is, when road load at the engine crankshaft falls below the threshold value $M_{e,FW}$. When the condition stops holding, freewheeling is released and thus reference gear is engaged again.

8.6. Fallback mode

When no look-ahead information is available, the fallback mode is entered. In this case, the velocity set-point is adapted to the instantaneous road according to:

$$v_{ref,k} = \begin{cases} \frac{v_{set}}{1 + K_{uh} \alpha(s)} & \alpha(s) > 0 \\ \frac{v_{set}}{1 + K_{dh} \alpha(s)} & \alpha(s) < 0 \end{cases} \quad (33)$$

The two coefficients K_{uh} and K_{dh} are scaling constants, and are set up such that it results $K_{uh} > K_{dh}$. The gear shifting logic, presented in D'Amato et al. (2017), implements a set of quantitative rules that can be briefly synthesized in: i) avoiding excessive shifting ii) choosing the right gear to deal with road load iii) respecting engine operating constraints iv) keeping the engine speed in the maximum efficiency rpm range. Freewheeling capability and engine control strategy remain unchanged when fallback mode is active.

8.7. Results

In Fig. 25 the results from the complete driving mission is displayed (known road information), where vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration.

8.8. Conclusions

The sequential solution of the optimal control problem presented allows reducing the number of decision variables and thus computational effort. The final solution in terms of reference speed and gear schedule allows reducing fuel consumption and keep computational time low enough for real life application. Constraints on maximum turbocharged speed have not been directly addressed since it did not show to be critical. On the other hand, The constraint on minimum λ has been addressed both in the development of the steady state maps used in engine control, adopting some safety margin, and in the engine control design phase. The constraint on maximum travel time has been taken into account in the definition of the cost function of the second optimization sub-problem and is also handled on-line with a reference speed, delay-based, correction. The presented fallback mode, implementing a smart shifting strategy and adapting speed set-point to instantaneous slope, allows reducing fuel consumption also in cases where no look-ahead data is available.

9. Solution, fourth participant

Author: Pavel Otta

Affiliation: Dept. Control Engineering, Czech Technical University in Prague, Czech Republic.

Contact: ottapav1@fel.cvut.cz

The benchmark problem complexity can be tackled by an off-line pre-computation giving a nearly optimal nominal solution and an on-line controller handling transients, disturbances and the off-line model discrepancy. The off-line optimizer discretizes the track to segments of constant slope and a sequence of speed and actuators that minimize fuel consumption on the whole track while satisfying all constraints. Using the pre-optimized speed and actuators profiles may lead to better performance of an on-line NMPC controller.

9.1. Reference Trajectory Optimization (off-line)

This section describes the off-line calculations that provide a reference trajectory for an on-line controller.

9.1.1. Track Description

For sake of simplification, the track is discretized (Fig. 6). The discrete values are equal to the average slope along each discretization interval. For further purpose, i -th track segment is an uphill if $i \in \mathcal{U}$ or downhill if $i \in \mathcal{D}$. The decision can be made using a threshold (as shown in Fig. 6) or either a smarter way. Note that the threshold cannot be chosen arbitrary, it must hold that a vehicle steady state exists for all $i \in \mathcal{U}$.

Figure 6: Detail of Sodertalje-Norrköping track discretized in 240m length intervals. At the top, red crosses denote uphill segments and blue circles denote downhill segments. At the bottom, dashed line denotes a threshold.

9.1.2. Truck Description

Multivariable polynomials can describe the vehicle steady-state behavior. Particularly interesting are steady-state values of truck speed v , fuel flow $\dot{m}_f$, oxygen flow $\dot{m}_o$, engine speed n_e , and turbo speed n_t . These values are captured on a grid of constant $\mathbf{u} = (u_{egr}, u_{vgt}, u_\delta)^T$ in range $[0, 100\%]$ each, α in range $[-0.02, 0.05 \text{ rad}]$, n_g in range $[1, 14]$. The break u_b is neglected at this moment. The situation is sketched on the principal model in Fig. 7.

For instance, the steady-state values of v captured at the $\{v, \alpha, n_g\}$ -grid can be interpolated by multivariable polynomial $v(\mathbf{u}, \alpha, n_g)$. Similarly, a polynomials $W_f(\mathbf{u}, \alpha, n_g)$, $\dot{m}_o(\mathbf{u}, \alpha, n_g)$, $n_e(\mathbf{u}, \alpha, n_g)$, $n_t(\mathbf{u}, \alpha, n_g)$ have been obtained. In the following section, these polynomials are used to find control inputs that minimize fuel consumption.

9.1.3. Fuel Consumption Minimization

One of the challenging issue of the benchmark is scheduling of the gear n_g —an integer variable that makes an optimization problem hard. To deal with n_g the following off-line optimization for every single combination of v, α, n_g is performed

$$\min_{\mathbf{u}} \quad \tilde{m}_f(\mathbf{u}, \alpha, n_g) \quad (34a)$$

$$\text{s.t.} \quad v(\mathbf{u}, \alpha, n_g) = v \quad (34b)$$

$$450 \text{ RPM} \leq n_e(\mathbf{u}, \alpha, n_g) \leq 2000 \text{ RPM} \quad (34c)$$

$$1.3 \dot{m}_f(\mathbf{u}, \alpha, n_g)(O/F)_s \leq \dot{m}_o(\mathbf{u}, \alpha, n_g) \quad (34d)$$

$$n_t(\mathbf{u}, \alpha, n_g) \leq 110000 \text{ RPM} \quad (34e)$$

$$\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max}, \quad (34f)$$

where $\tilde{m}_f(\mathbf{u}, \alpha, n_g) = \dot{m}_f(\mathbf{u}, \alpha, n_g)/v$ is fuel consumption per kilometers. Note that (34c) prevents engine over-speeding, (34d) is rewritten smoke generation limit (recall

Figure 7: Principal truck model composed of Internal Combustion Engine (ICE), gear box, and vehicle dynamics.

$\lambda = \frac{\dot{m}_o}{\dot{m}_f} \frac{1}{(O/F)_s}$), and (34f) provide the physical limitations of the actuators.

All manipulated variables $\mathbf{u}$ that minimize fuel consumption at any point of $\{v, \alpha, n_g\}$ -grid are known at this moment (Fig. 8). Further notice, optimal gear n_g^* can be determined — it is such a gear n_g (of all feasible gears) for which the fuel consumption is minimal for given v, α . Let the multivariable polynomial interpolating these grid points be denoted

$$\tilde{m}_f^*(v, \alpha) = \tilde{m}_f(v, \alpha, n_g^*). \quad (35)$$

Note that low resolution of the grid may lead to a situation where neighboring points vary a lot in the value of n_g^* — it would correspond to rough shifting. In the case, correction should be made (one can formulate an MILP considering smooth shifting that minimizes deviation from n_g^* for example).

Note that the solution of problem (34) depends on a specific vehicle — it has to be recalculated for each type of vehicle and load.

It has to be mentioned that vehicle mass m is treated as a constant but it should rather be involved as an additional variable in the multivariable polynomial description.

9.1.4. Minimization Criterion

There are two things that should be considered when performing fuel minimization over the mission: 1) fuel consumption at steady state speed; 2) fuel consumed by speeding up the vehicle. Let's consider stage cost at one track element to be

$$l(\tilde{v}_i) = \tilde{m}_f^*(\tilde{v}_i, \tilde{\alpha}_i) + \rho \max(0, \frac{1}{2} m [\tilde{v}_{i+1}^2 - \tilde{v}_i^2]), \quad (36)$$

where $\tilde{m}_f(\tilde{v}_i, \tilde{\alpha}_i)$ is steady-state consumption per kilometer (35) for speed $\tilde{v}_i$ (will be described further in the next paragraph), and slope $\tilde{\alpha}_i$ at the i -th track element is given. Coefficient $\rho = \frac{1}{\text{FHV} \cdot \text{CE}}$, where FHV is *fuel heating value* and CE is *combustion efficiency*, provides unit conversion from energy to fuel consumption. It is convenient to replace *max* function by *soft maximum* (see Cook (2011) for details) in the criterion.

Figure 8: Optimal costs of (34) — each dot denotes minimal value of $\tilde{m}_f$ at $\{v, \alpha, n_g\}$ -grid. Gears n_g are denoted by colors.

9.1.5. Vehicle Dynamics

Vehicle dynamics can be modeled by a double integrator described in the time domain as

$$\begin{aligned} \frac{ds(t)}{dt} &= v(t), \quad s(t_0) = x_0, \quad s(t_f) = x_f \\ \frac{dv(t)}{dt} &= \frac{1}{m}F(t), \quad v(t_0) = v_0, \end{aligned}$$

where initial position x_0 and final position x_f as well as initial speed v_0 are given. This model is then transformed into the spatial domain

$$\begin{aligned} \frac{d\tilde{t}(x)}{dx} &= \frac{1}{\tilde{v}(x)}, \quad \tilde{t}(x_0) = t_0 \\ \frac{d\tilde{v}(x)}{dx} &= \frac{1}{m}\tilde{F}(x)\frac{1}{\tilde{v}(x)}, \quad \tilde{v}(x_0) = \tilde{v}_0, \end{aligned} \quad (37)$$

where the transformed variables are time $\tilde{t}(x) = s^{-1}(x) = t$, speed $\tilde{v}(x) = v(\tilde{t}(x))$ and force $\tilde{F}(x) = F(\tilde{t}(x))$ at a particular position x . The parameter x is from the interval $[x_0, x_f]$.

Since the use of vehicle dynamics is limited to the case of freewheeling only, resulting force is $\tilde{F}(x) = (\tilde{F}_a - \tilde{F}_r - \tilde{F}_g - \tilde{F}_b)(x)$, i.e. sum of aerodynamic, frictional, gravitational and brake forces acting at position x . $\tilde{F}_r$ and $\tilde{F}_g$ depend on a given data, the roadway slope $\tilde{\alpha}(x)$ related to distance from the beginning of the track. Therefore, the information about roadway slope can be employed in the optimization directly.

9.1.6. Optimal Control Problem Statement

A simple Euler method can be used to discretize vehicle dynamics (37). Then the optimization problem is formed using stage cost (36) as

$$\begin{aligned} \min_{\tilde{v}} \quad & \sum_{i=0}^N l(\tilde{v}_i), \quad i \in \mathcal{U} \\ \text{s.t.} \quad & \tilde{t}_{i+1} = \tilde{t}_i + \frac{1}{\tilde{v}_i}, \quad \tilde{t}_0 = t_{\text{init}} \\ & \tilde{v}_{i+1} = \tilde{v}_i + \frac{1}{m}\tilde{F}_i\frac{1}{\tilde{v}_i}, \quad \tilde{v}_0 = \tilde{v}_{\text{init}} \\ & \underline{v} \leq \tilde{v}_i \leq \bar{v}, \quad i \in \mathcal{D}, \end{aligned} \quad (38)$$

where N is the total number of track segments, t_{init} is an initial time, $\tilde{v}_{\text{init}}$ is an initial speed, $\underline{v}$ is a lower and $\bar{v}$ is an upper speed limit.

Note that downhill segments with zero fuel consumption do not influence cost function and the speed is determined only by the constraints, namely the dynamics and speed coupling between neighboring segments.

In contrast with (34), optimization (38) additionally relates to a specific roadway — it has to be recalculated for each new route.

9.2. Reference Trajectory Tracking (on-line)

Although no such controller was implemented during the competition, the ambitions of this solution is described in this section.

The solution of (38) brings a near-optimal speed and control trajectories. When disturbances or model discrepancies cause a deviation from those reference trajectories, the solution of (38) is no longer valid, and the optimization of the problem with a new initial condition has to be solved all over again. However, if the deviation is small enough, *neighboring extremals* (NE) (Bryson (1975)) can be employed to find control update rather than repeatedly solve the optimization problem.

A two-stage nonlinear MPC control with diminishing horizon is designed. The first stage is a cost of the prediction horizon and the second stage is a cost of the rest of the diminishing horizon where control is assumed to be unconstrained. When using NE, the second stage can be formulated in term of terminal cost. As a consequence, a short prediction horizon can be chosen. Therefore the problem can be solved faster while preserving near the same control performance.

9.3. Results

In Fig. 26 the results from the complete driving mission is displayed (known road information), where vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration.

9.4. Conclusions

A systematic solution to the benchmark problem has been introduced. The problem complexity is tackled by the off-line optimization. The benefit of the presented approach is that even the off-line optimization is split into two problems. The first one (34) must be computed only once for a particular truck. The second one (38) must be computed for each new route. The pre-optimized solution can be exploited in an on-line predictive controller design. Not all described features — on-line predictive controller in particular — have been implemented during the competition. Nevertheless, taking advantage of the predictive control strategy is probably necessary to utilize the off-line optimization.

10. Solution, fifth participant

Author: Manne Held

Affiliation: Department of Automatic Control, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden.

Contact: manneh@kth.se

This solution is split into two parts. The first part is a pre-computation running before the vehicle starts and the second part contains the on-line controller that is run as the vehicle travels along the road. Each of these two parts is described in the following two subsections. The third subsection treats the control of VGT, EGR, gear and brake and the fourth and last subsection shows some results.

10.1. Pre-computations

The external forces acting on the vehicle are according to (3) air resistance, rolling resistance and gravitational force. The rolling resistance is mainly speed independent and these losses can therefore not be decreased. The gravitational force does not result in any loss of energy, rather a conversion between kinetic and potential energy. The losses due to air resistance and part of the rolling resistance losses are caused by a force proportional to the squared velocity. In order to minimize these losses, the vehicle should keep as equal velocity as possible throughout the driving mission. Because of the great weight of a heavy duty truck in combination with altitude changes of the road, it is not possible to keep constant speed at some parts of the road i.e. where the upslopes are too steep. There may also be sections of steep downslopes where the truck will accelerate even without any injection of fuel. Such sections are generally very beneficial during a fuel saving mission. By early experiments with the vehicle model, it was found that the vehicle does not consume any fuel at all if the neutral gear is selected. This corresponds to being able to shut off the engine while driving. Engaging neutral gear will from this point be referred to as *coasting*. The steep upslopes and downslopes will constitute the foundation of the derived solution presented in this section. At or around the steep upslopes, the truck will use maximum power. At or around steep downslopes, the truck will coast. For the remaining distance, the truck will keep constant speed.

The result of the pre-computation is a speed trajectory, a time trajectory indicating the expected elapsed time as a function of position and positions for start and end of sections where coasting and full power should be performed. The pre-computation is an iterative procedure that will end when the final time of the calculated trajectory T is 0-10 s less than the maximum allowed time $t_{\max} = s_{\text{tot}}/v_{\text{des}}$. The pre-computation starts by finding an initial guess for the speed trajectory. This is done by dividing the total distance with the allowed time. The first guess of the speed trajectory will thus be a constant value $\bar{v}$ that should be held everywhere. By using the model of resistive forces (3) with $F_w = F_w(x, v)$ for position x and velocity v , each

Figure 9: Sections considered as upslopes, downslopes and flat.

position x is defined as either flat, upslope or downslope. A section containing the position x is defined as upslope if

$$F_w(x, \bar{v}) > F_{\max}(\bar{v}), \quad (39)$$

defined as downslope if

$$F_w(x, \bar{v}) < 0, \quad (40)$$

and defined as flat if

$$0 \leq F_w(x, \bar{v}) \leq F_{\max}(\bar{v}), \quad (41)$$

where $F_{\max}(v)$ is the maximum tractive force given by a look-up table derived beforehand by experimenting with the model. For a specific velocity $\bar{v}$, the different sections can be seen in Fig. 9 with flat sections in blue, upslopes in red and downslopes in green.

By first solving a simplified version of the optimal control problem using dynamic programming, it was found that the approach described so far resembled the optimal solution, in terms of the speed profile. However, coasting should not only be performed in the downslope sections. It should start before the downslope in order to lower the velocity below $\bar{v}$. When entering the downslope, the vehicle will start accelerating and somewhere in the middle of the downslope, the velocity reaches and passes $\bar{v}$. At some distance after the end of the downslope, the vehicle has lost velocity such that it again drives at $\bar{v}$ which can now be held constant. The same procedure is valid for upslopes. Maximum power starts before the upslope and stops when the vehicle again has managed to reach $\bar{v}$. The calculations are done by matlab *ode45* in order to make the velocity profiles symmetric around $\bar{v}$ during the slopes. This is done by first calculating a midpoint in terms of energy loss/gain in the slope and solving the differential equation for the velocity forward and backward from this point. A resulting speed profile can be seen in Fig. 10 for a downslope. If the

Figure 10: Speed profile during coasting in a downslope.

upper speed limit is reached, the calculations are performed again in order to reach slightly below the speed restriction exactly at the end of the downslope. An example of this can be seen in Fig. 11. For very long or steep downslopes, the vehicle might need to lower the speed ahead of the downslope to unacceptable low values. A minimum value $v_{\min}$ is therefore introduced and set to $v_{\min} = 45$ km/h after testing different values.

The corresponding calculations during downslopes are done for upslopes with the resulting speed profile having an inverted shape from the one during coasting. When integrating the velocity using *ode45* in upslopes, one or many gear shifts may occur. From the gear controller, which will be discussed later, the velocities for when the truck will need to lower the gear is known. When any of these velocities is reached, the input force is set to zero for a distance corresponding to 2s. The distance between the point where the velocity leaves $\bar{v}$ until it again reaches it is considered either a coasting or a full power section. The speed trajectory is given by the constant speed $\bar{v}$ and the results from solving the differential equations during coasting and full power. If the sections for coasting and full power coincides by sections of the same type, superposition of the individual speed trajectories is done. If the sections for coasting and full power coincides by sections of the other type, then the more extreme in terms of deviation from $\bar{v}$ is chosen. A resulting speed trajectory can be seen in Fig. 12. Red color indicates sections of full power, green sections of coasting and blue sections where constant speed is kept. The strategy plotted on a height curve can be seen in Fig. 13.

Given the speed trajectory, the total trip time T and the margin to the maximum allowed time $t_{\text{margin}} = t_{\text{max}} - T$ is calculated. In order to finish in time, $t_{\text{margin}} \geq 0$ must hold and in order not to waste fuel by driving faster than necessary, $t_{\text{margin}} \leq 10$ is set as the upper bound. If these conditions do not hold, $\bar{v}$ is recalculated from the distance

Figure 11: Speed profile for coasting in the case it cannot be performed symmetric around $\bar{v}$ due to the speed restriction.

and time during which constant speed should be held. The full procedure of the pre-computation is summarized in the pseudo-code below:

```

Initialize  $\bar{v} = s_{\text{tot}}/t_{\text{max}}$ 
for  $i = 1 : 10$ 
    Find sections of upslopes and downslopes.
    Calculate sections of full power and coasting.
    Calculate trip time  $T$ .
    if  $0 \leq t_{\text{margin}} \leq 10$ 
        break.
    else
        Let  $t_{\text{dyn}}$  and  $s_{\text{dyn}}$  be time and distance for coasting and full power.
        Set  $\bar{v} = \frac{s_{\text{tot}} - s_{\text{dyn}}}{t_{\text{tot}} - t_{\text{dyn}}}$ .
    end
end

```

Since convergence is not guaranteed, the iteration will stop after ten iterations and use the trajectory calculated so far in case the stopping requirement is not fulfilled. The reason why this iterative procedure is needed is that the sections considered as upslopes or downslopes depend on the previous guess on the constant speed $\bar{v}$, since the resistive force and maximum force is velocity dependent. Also, even if the algorithm tries to create symmetric deviations during coasting and full power, there can be significant time losses during these sections.

10.2. On-line controller

The aim of the on-line controller is to follow the pre-computed speed trajectory and initiate coasting at the pre-computed sections and order maximum power at the

Figure 12: Speed trajectory calculated by the pre-computation. This trajectory is used as reference for the on-line controller.

corresponding sections. Since there is already a default limiter on the fuel injection, the controller itself does not have to limit the amount of fuel except for making sure not to cause too low values on λ . The limit on the fuel injection u_δ is set as a function of the air mass flow through the compressor $\dot{m}_c$ and the engine speed n_e as in Wahlström et al. (2005). For this specific vehicle the limit is

$$u_\delta \leq \frac{10.2 \cdot 10^5 \dot{m}_c}{n_e}. \quad (42)$$

The on-line controller is designed to strive for finishing $\Delta_t = 5$ s ahead of the time limit $t_{\max}$. This is done by adjusting the velocity in the sections where neither coasting nor full power is desired. From the pre-computation, the expected time of finish T is given. By stating that the vehicle should finish Δ_t ahead of the time limit, the amount of time the vehicle should be ahead of the computed trajectory is easily calculated. The fuel injection at these sections is given by

$$u_\delta = K_p v_{\text{err}} + K_t \Delta_{\text{tr}} + F_w / r_{\text{FF}} \quad (43)$$

where v_{err} is the difference between the velocity from the trajectory and the current velocity in km/h and $K_p = 50$ is a constant for this proportional gain. The variable Δ_{tr} is the difference between the position from the trajectory and the current position and $K_t = 1$ is a constant for this integral gain. The constant r_{FF} is the ratio between fuel and force which is empirically estimated to $r_{\text{FF}} = 40$ and this last term can therefore be seen as a feed forward part proportional to the resistive forces. The fuel injection may however not exceed the value from (42). On sections of full power, u_δ is given by setting equality in (42) and on sections for coasting $u_\delta = 0$.

Figure 13: Sections for coasting, full power and constant velocity.

10.3. Other control signals

10.3.1. EGR and VGT

Both the EGR and the VGT signal are kept at constant values during the full driving mission. To reach the conclusion not to change them dynamically, the model was decoupled from the driving mission and experiments were performed on the model of the vehicle alone. In steps of five percentage points in the span 0-100%, the EGR and VGT signals were varied for different values on the engine speed and the fuel injection in the typical working range of these signals. A typical plot showing the output torque from the engine as a function of EGR and VGT for the engine speed $n_e = 1100$ rpm and fuel injection $u_\delta = 210$ mg/stroke can be seen in Fig. 14. It can be seen that for VGT equal to 50%, higher values do not yield any significant increase in output torque. The EGR values for higher VGT does not have much influence on the torque. These observations were also valid for other values on the engine speed and fuel injection. With these motivations, the VGT was set to 50% and the EGR to 0%.

10.3.2. Gear and brake

The default gear selection was used. The brake was only used in case the velocity was less than 2km/h from the speed limit and the vehicle was accelerating.

10.4. Results

The results from a simulation with the default problem formulation can be seen in Fig. 15, where the resulting speed is plotted together with the trajectory from the pre-computation. The two trajectories are mostly similar with a few exceptions. During the interval $t = 1600$ -1900 s, the vehicle is ahead of the pre-computed trajectory and therefore drives slower than $\bar{v}$. During the interval $t = 2800$ -3200 s, the vehicle is behind the pre-computed trajectory and therefore drives faster than $\bar{v}$.

Figure 14: Output torque as function of EGR and VGT for a specific engine speed and fuel injection

Figure 15: Speed trajectory from simulation compared to speed trajectory from the pre-computation.

In Fig. 27 the results from the complete driving mission is displayed (known road information), where vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration.

10.5. Conclusion

This solution provides a straightforward and fairly intuitive controller for the benchmark problem. The focus is on minimizing the fuel consumption by engaging neutral gear during downslopes. There are, however, still room for improvements of the algorithm for pre-computing the speed trajectory. The algorithm should be made more robust. It was found during the evaluation that if the vehicle is heavily loaded or if the road is very hilly, the algorithm might fail to find a solution. The reason is that during such circumstances, the vehicle will lose too much time during upslopes and during coasting in downslopes that cannot be compensated for during the flat sections. The constant

speed during flat areas would have to be higher than the maximum allowed, i.e., $\bar{v} > 91$ km/h. A second improvement could be to increase the distance of coasting even more. This could be done by increasing the speed ahead of downslopes and by allowing the vehicle to lose speed below $\bar{v}$ directly after a downslope. Whether or not the fuel saved by increasing the coasting distance is greater than the fuel lost due to increased air resistance by this method is however not clear.

11. Solution, sixth participant

Author: Ulrich Vögele

Affiliation: Technische Hochschule Ingolstadt, Germany.

Contact: ulrich.voegel@thi.de

11.1. Introduction

Dynamic Programming is often proposed to improve the fuel economy of vehicles, by automating their longitudinal behavior. Examples of such controllers are given in Radke (2013); Themann et al. (2014b,a); Vögele and Endisch (2015). A property of Dynamic Programming is that it can find the globally optimal solution for a scenario. However, the computation time can be very long and a lot of research was done about efficient implementations of Dynamic Programming so that it can be used to control vehicles on-line. The inclusion of additional optimization criteria, besides fuel consumption, further increases the computational complexity. Wahl et al. (2013) take time aspects into account by using the weighted sum of fuel consumption and trip time as cost-function. Because this cost function is still a scalar it doesn't increase the computational complexity. However, the selection of an appropriate weighting factor to fulfill constraint (4h) and minimizing the fuel consumption may be difficult.

Therefore, a controller is developed that simultaneously respects time and fuel consumption by using a multidimensional cost-function. In order to reduce computation time and memory usage an approximation scheme similar to the one described in Dabia et al. (2010) is used.

The structure of the controller is presented in the following. For each component a brief description of its purpose and function is given. After that the control algorithm is presented in more detail. Then the fallback mode, allowing the controller to operate even in case no predictive slope data is given, is presented.

11.2. Controller Structure

The controller consists of a low-level controller, directly controlling the EGR-valve u_{egr} , VGT u_{vgt} , and the fuel injection u_{δ} , based on current sensor readings, a desired engine torque $M_{e,set}$ and the current gear u_g .

The desired engine torque $M_{e,set}$ is provided by a velocity controller, that limits the commanded torque $M_{e,ref}$ and actuates the brake u_b to satisfy constraint (4g).

Requested gear $u_{g,r}$ and the commanded torque $M_{e,ref}$ are provided by the Predictive Controller, which implements a predictive velocity control of the truck, such that a given route is driven within a certain time using the least possible amount of fuel. The complete structure is shown in Fig. 16.

Figure 16: Structure of the controller.

11.2.1. Low-Level Controller

The Low-Level Controller directly commands the EGR-valve u_{egr} , the variable turbo geometry (VGT) u_{vgt} , and the fuel injection u_{δ} . Primary goals of the Low-Level Controller are to prevent the engine from emitting smoke by regulating the λ -Value to a value above 1.3, constraint (4e), and to prevent the turbo from overspeeding, constraint (4f). Secondary goal is to control the engine such that the demanded torque is provided with the highest efficiency possible. The structure of the low-level controller is depicted in Fig. 17.

This structure is similar to Wahlstrom et al. (2010), where two PID controller for u_{egr} and u_{vgt} were used. Because NOx emissions were not part of the benchmark evaluation a simplified control using only one PID controller with

$$u_{vgt} = -\text{PID}(\lambda_{O, \text{set}} - \lambda_O). \quad (44)$$

Having a similar influence on λ_O , u_{egr} is set to

$$u_{egr} = u_{vgt} - 20. \quad (45)$$

To minimize pump losses $\lambda_{O, \text{set}}$ is set so that the maximum allowed fuel rate $u_{\delta, \text{max}}$, without violating constraint (4e), equals the demanded fuel rate $u_{\delta, \text{unlim}}$. According to (2), rewriting the fuel mass flow as a function of demanded fuel

Figure 17: Structure of the Low-Level Controller.

Figure 18: Comparison of maximal possible fuel rate $u_{\delta, \text{max}}$ and demanded fuel rate $u_{\delta, \text{unlim}}$ (upper graph) for the given control of u_{vgt} (lower graph).

rate, the reference value $\lambda_{O, \text{set}}$ is calculated by

$$\lambda_{O, \text{set}} = \frac{120}{10^{-6}} \frac{\dot{m}_{ei} X_{Oim}}{u_{\delta, \text{unlim}} n_e n_{cyl} (O/F)_s}. \quad (46)$$

Fig. 18 shows the u_{vgt} control together with the demanded fuel rate $u_{\delta, \text{unlim}}$ and $u_{\delta, \text{max}}$ for a part of the test scenario, including three gear up-shifts at 35 s, 61 s and 67 s.

The feed-forward fuel rate $u_{\delta, \text{ff}}$ is directly calculated from the desired torque $M_{e, \text{set}}$ using (26)-(31) from Wahlström and Eriksson (2011). To prevent unnecessary fuel injection during gear shifts and while the clutch is not synchronized, $u_{\delta, \text{ff}}$ is cut off during those times, forming $u_{\delta, \text{unlim}}$. The fuel limiter finally calculates u_{δ} and the set-point for the VGT-control while respecting constraint (4e).

11.2.2. Velocity Controller

The Velocity Controller prevents overspeeding within the next second by limiting the commanded torque $M_{e, \text{ref}}$. However, negative slopes may require an engagement of the brakes beside torque reduction to satisfy constraint (4g).

11.2.3. Predictive Controller

The Predictive Controller consists of a Slope Provider, an Optimizer, Gear Selection and Torque Control, see Fig. 19. The Slope Provider provides the slope α_{slope} for the future route at the distances s_{slope} .

Based on the future slope profile and the current state of the vehicle, given by its velocity v and gear u_g , and the time t already traveled a speed- and gear-profile $\mathbf{v}_{\text{optim}}$ and $\mathbf{u}_{g, \text{optim}}$ is calculated. The first control inputs are then applied to the vehicle in the Gear Selection and Torque Control subsystems. The Optimizer then calculates a new speed- and gear-profile based on the consecutive state of the vehicle from which again only the first controls are applied, and so on.

11.3. Control Algorithm

The Optimizer uses a multi-objective approximate dynamic programming scheme to calculate the next gear and velocity profile for minimizing fuel consumption as described in Vögele and Endisch (2016).

Figure 19: Structure of the Predictive Controller.

Given the current vehicle state $\mathbf{x}$, consisting of its velocity v and the current gear u_g

$$\mathbf{x} = \begin{pmatrix} v \\ u_g \end{pmatrix}, \quad (47)$$

and predictive slope data the Optimizer tries to minimize the fuel consumption by calculating the optimal velocity profile $\mathbf{v}_{\text{optim}}$ and gear-shift strategy $\mathbf{u}_{g,\text{optim}}$ over a certain prediction horizon. This requires a discretization of the route with a step size of Δs , so that N steps constitute the prediction horizon.

Let $\mathbf{X}_i$ be the set of possible states at prediction step i . By variation of the velocity for each element in $\mathbf{X}_i$ by $\Delta \mathbf{v}$ and then variation of the gear for each of these elements by $\Delta \mathbf{u}_g$ the set of possible states $\mathbf{X}_{i+1,\text{full}}$ is constructed. In addition for each element in $\mathbf{X}_i$ coasting by switching in neutral gear shall be considered, where the final velocity starting from v_i is given by $f_{\text{free}}(v_i)$. Calculation of the set $\mathbf{X}_{i+1,\text{full}}$ from $\mathbf{X}_i$ is denoted as

$$\mathbf{X}_{i+1,\text{full}} = \text{step}(\mathbf{X}_i). \quad (48)$$

The set $\mathbf{X}_{i,\text{full}}$ can contain states that violate the given constraints, because they have either a too high or too low velocity (4g), try to engage a non-feasible gear (4d), or need too much time (4h). These states can be removed, giving the set of feasible states $\mathbf{X}_{i,\text{feas}}$, denoted as

$$\mathbf{X}_{i,\text{feas}} = \text{feasible}(\mathbf{X}_{i,\text{full}}). \quad (49)$$

Using the multi-objective cost function $\mathbf{J}$ with

$$\mathbf{J} = \begin{pmatrix} m_f \\ t \end{pmatrix} = \sum_{k=0}^{N-1} \begin{pmatrix} \Delta m_f \\ \Delta t \end{pmatrix} \quad (50)$$

a possible velocity and gear profile is optimal if there exists no other solution with smaller fuel consumption m_f and trip time t .

An optimal solution itself has to consist of optimal solutions to the sub-problems, see Bellman's Principle of Optimality. Therefore, without loss of optimality, all states in $\mathbf{X}_{i,\text{feas}}$ not contributing to an optimal solution can be removed

$$\mathbf{X}_{i,\text{optim}} = \text{optim}(\mathbf{X}_{i,\text{feas}}). \quad (51)$$

Figure 20: Reduction of the set of optimal solutions $\mathbf{X}_{i,\text{optim}}$ (all dots) to a reduced set $\mathbf{X}_{i,\text{red}}$ (blue dots).

Finally the set of optimal solutions $\mathbf{X}_{i,\text{optim}}$ is reduced by down-sampling to a certain number of solutions similar to Dabia et al. (2010). This is performed because otherwise the set of optimal solutions $\mathbf{X}_{i,\text{optim}}$ would grow exponentially with each iteration which would lead to high computation time and memory demand. Therefore, each dimension in the cost-function $\mathbf{J}$ is divided into n_{red} equidistant windows between the minimal and maximal costs, as depicted in Fig. 20. Then for each window, only one solution is kept. The more windows are used, the more the solution converges towards the globally optimal solution together with an increase of computation time and memory usage. Hereafter, the reduction is described by

$$\mathbf{X}_{i,\text{red}} = \text{reduce}(\mathbf{X}_{i,\text{optim}}). \quad (52)$$

This is performed until the number of prediction steps N is reached. From the obtained set of solutions the one with the lowest fuel consumption x_{optim} is then selected and its affiliated control inputs are applied. The overall algorithm is given in Algorithm 1.

Algorithm 1. Optimizer

- 1: $\mathbf{X}_0 \leftarrow \mathbf{x}$
- 2: **for** $i = 0$ to $N-1$ **do**
- 3: $\mathbf{X}_{\text{full},i+1} \leftarrow \text{step}(\mathbf{X}_i)$
- 4: $\mathbf{X}_{\text{feas},i+1} \leftarrow \text{feasible}(\mathbf{X}_{i+1,\text{full}})$
- 5: $\mathbf{X}_{\text{optim},i+1} \leftarrow \text{optim}(\mathbf{X}_{i+1,\text{feas}})$
- 6: $\mathbf{X}_{i+1} \leftarrow \text{reduce}(\mathbf{X}_{i+1,\text{optim}})$
- 7: **end for**
- 8: $x_{\text{optim}} \leftarrow \min_{m_f} \mathbf{X}_N$

To account for model inaccuracies, control disturbances and a prediction horizon that can be smaller than the overall route, the optimization is performed again after a distance of Δs and the control inputs are updated.

Figure 21: Structure of the neural network used for slope prediction.

11.4. Fallback Mode

When slope information is provided for the route the Slope Provider simply passes this information as two vectors α_{slope} and s_{slope} containing slope values and distance of each slope value to the Optimizer. In case no slope information is provided, the Slope Provider predicts the slope for the next N steps using a neural network, which has been trained on the route provided with the benchmark. The neural network has been trained using back-propagation with a conjugate gradient descent method. The input of the neural network are the current slope $\alpha(s)$ and the slope at 100 m before the current position $\alpha(s - 100)$, so that a change in slope over the route can be taken into account. The neural network then predicts the future slope over the prediction horizon N in steps of Δs , so that it has a structure as depicted in Fig. 21.

The mean absolute slope error of the neural network (NN) is depicted in Fig. 22. Compared to a constant prediction (Const), assuming a constant slope over the prediction horizon, a linear return to zero (Lin) prediction, assuming the slope will eventually return to zero and just assuming zero slope (Zero), the neural network has the lowest prediction error over a prediction horizon of 1 km. Beyond that distance the neural network has a similar performance as the zero slope prediction. The prediction horizon of the Optimizer is therefore limited to 1 km during fallback mode.

11.5. Results

In Fig. 28 the results from the complete driving mission is displayed (known road information), where vehicle speed, gear selections, turbocharger speed and EGR and VGT controls are compared with the original ECU configuration.

11.6. Conclusion

The presented method is able to provide control inputs for a truck that satisfy the given constraints while minimizing fuel consumption. The optimization uses an approximation scheme that allows a trade-off between optimality of the obtained control inputs and computation time and memory usage. When no slope information is available

for the route, the future slope profile is estimated in a fallback mode using a neural network. This allows to use the same optimization algorithms, whether predictive data is available or not.

12. Benchmark of the Solutions

The different solutions are benchmarked and compared to each other, the results are displayed in Table 4. The results of the benchmark competition at the AAC conference (Advances in Automotive Control 2016, Kolmården) are not represented by only the values in Table 4. In addition several test routes and a weighted scoring system was used. Even though there are six solutions to the same problem, the participants have very different solution strategies. All the solutions comply with the constraints on the optimization problem in (4), all solutions complete the driving mission in time and managed to control the smoke limiter and not over speeding the turbocharger. Not all the solutions can handle Fallback mode (fail-safe mode if GPS signal is missing). Some solutions have larger requirement on cloud computing time t_{Cloud} and some online computing time t_{Online} . The online computing time is the time it takes to simulate the benchmark route, while the cloud computing time is the pre-computational time needed to setup the solution strategy before the simulation is started. The main benchmark variable is the fuel consumption ϕ_{fuel} , since the main goal of the task was to minimize the fuel consumption.

The results from the simulated solutions are displayed in Fig. 23-28. In the figures, the vehicle speed, altitude, gear, turbocharger rotational speed and control signals for EGR and VGT are depicted.

It is evident in the speed profiles that solution two and four have resulted in too high average speed, which punishes the fuel consumption of these solutions. Comparing the 'Gear'-trajectories of Fig. 25, 27 and 28, reveals that these three solutions utilizes roll out, by decoupling the clutch when there is no need for traction torque from the

Figure 22: Mean absolute error of the neural network (NN) used for slope prediction, compared to a constant (Const), linear (Lin) and zero (Zero) prediction.

Table 4: Results from the six solution strategies on the Södertälje-Norrköping route. Mean fuel consumption, $\bar{\phi}_{\text{fuel}}$ (l/100km), cloud computation time, t_{Cloud} (s), online computation time, t_{Online} (s), and fallback mode for loss of GPS signal. (The computational times are recorded while simulating the solutions on a 2.7 GHz Dual-core Intel i5 computer with 8gb ram).

Participant	1	2	3	4	5	6
$\bar{\phi}_{\text{fuel}}$	35.02	43.20	34.88	37.27	33.89	34.36
t_{Cloud}	465	0.003	84	91	22	0.151
t_{Online}	358	118	1713	153	91	4607
<i>Fallback</i>	No	Yes	Yes	No	No	Yes

combustion engine. These three have the lowest fuel consumption on the test track Södertälje-Norrköping, but the difference to solution one is very small. Solution one on the other hand stands out with much smoother control action than the other solutions, and possibly could have improved on the fuel consumption by utilizing roll out in down hill.

Comparing the EGR and VGT control signals in Fig. 23-28, reveals that four of the solutions actively control the EGR and VGT valve, while two specify constant values. All six solutions reduce the need for downshifting when approaching uphill sections as opposed to the reference strategy, which performs several down shifts.

Two solutions differ significantly from the other four when comparing the fuel control signal, u_{δ} . Solution one as previously mentioned has a smoother control signal, while the variance in the fueling for solution two is higher compared to solution three to six. Solution two is also the one with the highest fuel consumption, and this is probably a contributing factor that should be avoided.

From an computational time perspective, at least solution three and six would need refinement and optimization before implementation in an on-board controller. Although the driving mission is around 5400s long, an on-board system does not have the performance as the computer producing the result in Table 4. The cloud computational time would probably not be a problem, possibly with the exception of solution one, but a cloud computer for a real system could have significantly better performance than the test computer for the benchmark.

13. Summary and Conclusions

The paper presents six different solutions to the benchmark problem, of minimizing the fuel consumption of a truck, given a maximum trip time and road profile.

The participants were offered the possibility of preforming offline pre-computations, called cloud computing since it could be performed on a separate powerful hardware outside of the real time controller, given the known driving mission including road profile, vehicle weight and desired mean trip velocity.

The results of the pre-computations together with vehicle sensor feedback and road profile from GPS are then used by the online controller. A fallback mode to account for loss

of GPS signal or absence of road profile data for the given driving mission could also be included, and gave merits in the evaluation.

Despite the specific problem formulation, six very different solutions were proposed, which highlights that there are many ways to approach the solution of a complex problem. Optimal control was used to some extent by most participants, mainly in the offline computations but one participant choose to do online optimization using dynamical programming. In the offline pre-computation phase it was used to find an optimal velocity trajectory and to find optimal parameters for an online controller. The online computation time is an indication of how viable the strategy would be for real time implementation. Three of the solutions handle the fallback mode, when no road data is available, without violating any of the constraints.

14. Future Benchmark Activities

The benchmark and the proposed solutions are published and available for researchers and engineers, so that it can act as an inspiration for other researchers and practicing engineers. The benchmark model and the problem description is available for download for free at: <http://www.fs.isy.liu.se/Software>. The hope is that this availability of the benchmark model and problem with the six solutions described here will have an impact on the research and education within the Automotive field.

Researchers and practicing engineers are encouraged to analyze the solutions and their respective merits and use this as inspiration in their development of new even more efficient algorithms. This benchmark on planning of driving missions will hopefully be beneficial for the development of planning algorithms for future fuel efficient autonomous trucks.

We are promoting the development and usage of benchmark problems in future IFAC Events, in particular the model is freely available and can be modified or extended with more details or other aspects can be included in the problem. It is the hope that the benchmark can contribute to the development of an energy efficient and sustainable transportation system.

Figure 23: First participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Figure 24: Second participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Figure 25: Third participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Figure 26: Fourth participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Figure 27: Fifth participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Figure 28: Sixth participant solution. Visualization of the variables: vehicle speed, altitude, gear selection, turbocharger rotational speed, control signals u_{egr} and u_{vgt} and the injected amount of fuel are plotted over the driven distance. The gray path in each plot is the reference controller, which was distributed to the participants. The blue path is the participant solution.

Acknowledgments

The model is developed in collaboration between Linköping University and Scania CV AB. This work was funded by the Vinnova Industrial Excellence Center "Linköping Center for Sensor Informatics and Control" LINK-SIC <http://linksic-isy.liu.se/>.

References

- Arsie, I., Graziosi, M., Pianese, C., Rizzo, G., Sorrentino, M., 2005. Control strategy optimization for hybrid electric vehicles via provisional load estimate. *Review of Automotive Engineering* 26 (3), 341–348.
- Betts, J. T., 2010. Practical methods for optimal control and estimation using nonlinear programming. SIAM.
- Bock, H. G., Plitt, K. J., 1984. A multiple shooting algorithm for direct solution of optimal control problems. In: *Proceedings of the 9th IFAC World Congress*. Vol. 17. Budapest, Hungary, pp. 1603–1608.
- Bryson, Arthur Earl; Ho, Y.-C., 1975. *Applied optimal control: optimization, estimation and control*. CRC Press.
- Chang, D. J., Morlok, E. K., 2005. Vehicle speed profiles to minimize work and fuel consumption. *Journal of Transportation Engineering* 131 (3), 173 – 182.
- Cook, J. D., 2011. Basic properties of the soft maximum. *Collection of Biostatistics Research Archive*.
- Dabia, S., Van Woensel, T., De Kok, A., 2010. A dynamic programming approach to multi-objective time-dependent capacitated single vehicle routing problems with time windows. Tech. rep., Working Paper 313, Eindhoven University of Technology, School of Industrial Engineering, The Netherlands.
- D’Amato, A., Donatantonio, F., Arsie, I., Pianese, C., 2017. Development of a cruise controller based on current road load information with integrated control of variable velocity set-point and gear shifting. In: *SAE World Congress*. No. 2017-01-0089 in SAE Technical Paper Series. Detroit, MI, USA.
- Donatantonio, F., D’Amato, A., Arsie, I., Pianese, C., 2018. A multi-layer control hierarchy for heavy duty vehicles with off-line dual stage dynamic programming optimization. *Transportation Research Part C: Emerging Technologies* 92, 486 – 503.
- Eriksson, L., 2001. Simulation of a vehicle in longitudinal motion with clutch engagement and release. In: *3rd IFAC Workshop "Advances in Automotive Control" Preprints*. Vol. 34. Karlsruhe, Germany, pp. 65–70.
- Eriksson, L., Larsson, A., Thomasson, A., 2016. The AAC2016 Benchmark – Look-Ahead Control of Heavy Duty Trucks on Open Roads. In: *Proc. of the IFAC Symposium on Advances in Automotive Control*. Vol. 49. Kolmården, Sweden, pp. 121–127.
- Eriksson, L., Nielsen, L., 2014. *Modeling and Control of Engines and Drivelines*. John Wiley & Sons.
- Fröberg, A., Hellström, E., Nielsen, L., April 2006. Explicit fuel optimal speed profiles for heavy trucks on a set of topographic road profiles. In: *SAE World Congress*. No. 2006-01-0008 in SAE Technical Paper Series. Detroit, MI, USA.
- Fröberg, A., Nielsen, L., 2007. Optimal fuel and gear ratio control for heavy trucks with piece wise affine engine characteristics. *IFAC Proceedings Volumes* 40 (10), 335–342.
- Hellström, E., Åslund, J., Nielsen, L., 2010a. Design of an efficient algorithm for fuel-optimal look-ahead control. *Control Engineering Practice* 18 (11), 1318–1327.
- Hellström, E., Åslund, J., Nielsen, L., 2010b. Horizon length and fuel equivalents for fuel-optimal look-ahead control. In: *6th IFAC Symposium on Advances in Automotive Control*. Munich, Germany.
- Hellström, E., Fröberg, A., Nielsen, L., 2006. A real-time fuel-optimal cruise controller for heavy trucks using road topography information. In: *SAE World Congress*. No. 2006-01-0008. Detroit, MI, USA.
- Hellström, E., Ivarsson, M., Åslund, J., Nielsen, L., 2009. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. *Control Engineering Practice* 17 (2), 245–254.
- Hooker, J., 1988. Optimal driving for single-vehicle fuel economy. *Transportation Research Part A: General* 22 (3), 183–201.
- Hooker, J., Rose, A. B., Roberts, G., 1983. Optimal control of automobiles for fuel economy. *Transportation Science* 17 (2), 146–167.
- Huang, W., Bevely, D. M., Schnick, S., Li, X., 2008. Using 3d road geometry to optimize heavy truck fuel efficiency. In: *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, pp. 334–339.
- Ivarsson, M., Åslund, J., Nielsen, L., April 2010. Impact of AMT gear-shifting on fuel optimal look ahead control. In: *SAE World Congress*. No. 2010-01-0370 in SAE Technical Paper Series. Detroit, MI, USA.
- Lattemann, F., Neiss, K., Terwen, S., Connolly, T., Oct 2004. The predictive cruise control - a system to reduce fuel consumption of heavy duty trucks. In: *SAE Commercial Vehicle Engineering Congress & Exhibition*. No. 2006-01-0008 in SAE Technical Paper Series.
- Monastyrsky, V., Golownykh, I., 1993. Rapid computation of optimal control for vehicles. *Transportation Research Part B: Methodological* 27 (3), 219–227.
- Myklebust, A., 2014. Dry clutch modeling, estimation, and control. Ph.D. thesis, Linköping University.
- Myklebust, A., Eriksson, L., 2013. The effect of thermal expansion in a dry clutch on launch control. In: *AAC’13 – 7th IFAC Symposium on Advances in Automotive Control*. Vol. 46. Tokyo, Japan, pp. 458–463.
- Neiss, K., Terwen, S., Connolly, T., 2004. Predictive speed control for a motor vehicle. US Patent Pub. No. US 68359, 2004.
- Nocedal, J., Wright, S. J., 2000. *Numerical Optimization*. Springer.
- Obert, E. F., 1973. *Internal Combustion Engines and Air Pollution*. Harper & Row.
- Ozatay, E., Onori, S., Wollaege, J., Ozguner, U., Rizzoni, G., Filev, D., Michelini, J., Di Cairano, S., 2014. Cloud-based velocity profile optimization for everyday driving: A dynamic-programming-based solution. *IEEE Transactions on Intelligent Transportation Systems* 15 (6), 2491–2505.
- Passenberg, B., Kock, P., Stursberg, O., 2009. Combined time and fuel optimal driving of trucks based on a hybrid model. In: *European Control Conference, Budapest*. pp. 1–6.
- Radke, T., 2013. *Energieoptimale Längsführung von Kraftfahrzeugen durch Einsatz vorausschauender Fahrstrategien*. Vol. 19. KIT Scientific Publishing.
- Schwarzkopf, A., Leipnik, R., 1977. Control of highway vehicles for minimum fuel consumption over varying terrain. *Transportation Research* 11 (4), 279–286.
- Sundstrom, O., Guzzella, L., 2009. A generic dynamic programming matlab function. *Proceedings of the IEEE International Conference on Control Applications*, 1625–1630.
- Terwen, S., Back, M., Krebs, V., 2004. Predictive powertrain control for heavy duty trucks. *IFAC Proceedings Volumes* 37 (22), 105–110.
- Themann, P., Bock, J., Eckstein, L., 2014a. Optimisation of energy efficiency based on average driving behaviour and driver’s preferences for automated driving. *IET intelligent transport systems* 9 (1), 50–58.
- Themann, P., Krajewski, R., Eckstein, L., 2014b. Discrete dynamic optimization in automated driving systems to improve energy efficiency in cooperative networks. In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, pp. 370–375.
- Vögele, U., Endisch, C., 2016. Predictive vehicle velocity control using dynamic traffic information. In: *SAE World Congress*. No. 2016-01-0121 in SAE Technical Paper Series. Detroit, MI, USA.
- Vögele, U., Endisch, C., 2015. Potential analysis of predictive optimal velocity control for electric and conventional vehicles. In: *2015 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, pp. 196–201.
- Wächter, A., 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.

- Wahl, H.-G., Bauer, K.-L., Gauterin, F., Holzäpfel, M., 2013. A real-time capable enhanced dynamic programming approach for predictive optimal cruise control in hybrid electric vehicles. In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). IEEE, pp. 1662–1667.
- Wahlström, J., 2009. Control of EGR and VGT for emission control and pumping work minimization in diesel engines. Ph.D. thesis, Linköping University.
- Wahlström, J., Eriksson, L., 2011. Modelling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics. Proceedings of the Institution of Mechanical Engineers, Part D, Journal of Automobile Engineering 225 (7), 960–986.
- Wahlstrom, J., Eriksson, L., Nielsen, L., 2010. Egr-vgt control and tuning for pumping work minimization and emission control. IEEE Transactions on Control Systems Technology 18 (4), 993–1003.
- Wahlström, J., Eriksson, L., Nielsen, L., Pettersson, M., 2005. Pid Controllers and Their Tuning for EGR and VGT Control in Diesel Engines. In: 16th IFAC World Congress. Vol. 38. pp. 212–217.

Table A.5: The AAC2016 Truck Benchmark results, the bottom line (Total Score) displays the total score for each participant.

Cycle	Criterion	Alberto Reig	Mark Eifert	Fabrizio Donatantonio	Pavel Otta	Manne Held	Vögele Ulrich
1	Fuel Consumption	480	300	420	360	720	600
	Cloud computing	30	60	35	30	40	50
	Controller execution	105	150	90	120	180	75
	Fallback	0	120	120	0	0	120
	Industrial ranking	120	100	240	140	200	160
	Academia ranking	120	60	100	50	70	80
	Vehicle Weight min	35	0	50	0	60	40
	Vehicle Weight max	50	0	60	0	0	0
2	Fuel Consumption	420	300	480	360	600	720
	Cloud computing	30	60	35	25	40	50
	Controller execution	105	150	90	120	180	75
	DNQ	60	0	60	60	60	60
3	Fuel Consumption	0	420	600	480	0	720
	Cloud computing	0	60	35	40	0	50
	Controller execution	0	180	120	150	0	105
	DNQ	0	40	60	40	0	50
Total Score		1555	2000	2595	1975	2150	2955

Appendix A. Ranking from AAC 2016 Benchmark Contest

This section contains the results of the Benchmark Competition held at AAC 2016. The goal of the competition was to complete a driving mission within a specified time, while fulfilling the problem constraints (see further Eriksson et al. (2016) for more information about the benchmark problem). A vehicle model was distributed to the participants, where the vehicle’s controller was free for the participants to change. A known road profile, specifying road slope and distance, was handed out along with the model to give the participants a case which they could use to test their control strategies. The different solutions were scored at a set of different criteria with different weight, and the results are displayed in Table A.5. The different solutions were tested on three different cycles, one known to the participants, one which were the reverse known cycle, and one unknown long drive cycle (10 times longer driving distance than the known cycle). Two solutions did not comply with the long cycle, therefore they were not awarded any points for Cycle 3 in the table.