

The AAC2016 Benchmark – Look-Ahead Control of Heavy Duty Trucks on Open Roads

Lars Eriksson* Anders Larsson** Andreas Thomasson*

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

** *Scania CV AB, Södertälje, Sweden*

Abstract: A benchmark problem for fuel efficient control of a truck with engine, driveline, and chassis models on a given mission with a road topography profile is formulated. The vehicle model is provided with open access to the vehicle model equations and parameters. It is compiled from model components validated in previous research projects and the result is a non-linear model that contains mixed continuous and discrete control variables. The driving scenario is provided as 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.

© 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

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

1. INTRODUCTION

Reducing fuel consumption while fulfilling the increasing needs for transportation of people and goods is an important societal challenge. One interesting development trend is the availability of driving mission information, with planning using GPS and road profile combined with the increase in computing power. Aiming at meeting these ends and supporting the community in the development process, a model and benchmark problem for fuel efficient control of a long haulage heavy duty truck on an open road with a given road profile is compiled, presented and provided as an open system to promote development. The basic vehicle model includes: diesel engine, clutch, transmission and vehicle. In conjunction with this there is a controller template and a driving mission with road profile.

The scenario is that the truck is being loaded to a given total vehicle mass m and the mission is to drive a given distance s_{tot} over a road with a given position dependent slope $\alpha(s)$. The goal is then to develop a cruise (vehicle speed) controller that gives as low fuel consumption as possible over the driving mission. One can think about both on-line solutions 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. Initially the vehicle can be at stand still or already driving when the driver presses the cruise controller button.

The main contribution is the compilation and documentation of an open source model that is made available for the scientific community. Experimental equipment for building and validating system models is not always available to everyone doing research on control and optimization methods and this contributes by making a fairly complex model for a relevant problem available for the scientific community. The modeling process can also be time consuming and make up a large part of a scientific project. Thereby this paper and the joint model can benefit the scientific community and speed up the pace of innovation.

1.1 Optimal Control Formulation

The task is to control the main vehicle actuators \mathbf{u} (gear request $u_{g,r}$ -, fuel injection m_f 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. Such 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 $v \leq 91$, smoke generation low $\lambda \geq 1.3$ and turbocharger healthy $n_t \leq 110\,000$ RPM.

$$\min \int_0^T \dot{m}_f dt \quad (1a)$$

$$s.t. \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (1b)$$

$$\mathbf{x}(0) = \mathbf{x}_0 \quad (1c)$$

$$\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \quad (1d)$$

$$\lambda(\mathbf{x}, \mathbf{u}) \geq 1.3 \quad (1e)$$

$$n_t \leq 110\,000 \text{ RPM} \quad (1f)$$

$$v \leq 91 \text{ km/h} \quad (1g)$$

$$s(T)/T \geq v_{des} \quad (1h)$$

* The development of this benchmark has been supported by the Vinnova Industry Excellence Center, Linköping Center for Sensor Informatics and Control, LINK-SIC.

The problem formulation shows that the fuel consumption is the main criterion but the constraints are also important for emissions and safe operation of the engine and vehicle. Other dimensions that will be evaluated beside the optimal control problem are the CPU memory and execution time and cloud computation time of the proposed solution.

2. COMPLETE SYSTEM MODEL

The top level of the benchmark model is shown in Figure 1 it consists of a controller and a vehicle model, with control signals u (gear, fuel injection, variable geometry turbine, egr valve, brake), and a sensor bus (vehicle velocity, road slope, distance driven, lambda, inlet manifold pressure, egr ratio, compressor air mass flow, turbo speed, engine speed).

The requirements for running the provided model are the following:

- Matlab/Simulink.
- Compiler and code generation for Simulink acceleration.
- Stateflow for gear controller. This can be removed if it is not needed in the presented solution.

Instructions for getting started:

- (1) Download and extract **TruckBenchmark.vXX.zip**, where XX is the version number. Currently the model is available by request only, later when the competition is over the files will be made available at both the AAC 2016 and vehicular systems web-sites: aac2016.isy.liu.se, www.fs.isy.liu.se/Software.
- (2) Point Matlab to the extracted TruckBenchmark directory and execute the script **RunMe.m**.
- (3) With the provided controller and settings in the benchmark download it takes around 150 seconds on standard laptop to simulate, after compilation.
- (4) Note that the provided example controller does not fulfill the maximum speed limit since it does not include a brake controller.
- (5) To start designing your own controller you open the **Controller.slx** file and in the menu “Diagram” you select “Unlock Library”.

LiU Scania Truck Model
AAC2016 Benchmark
Authors
Anders Larsson, Scania CV
Lars Eriksson, Vehicular Systems, LiU
Version 0.2 – 2015-07-09

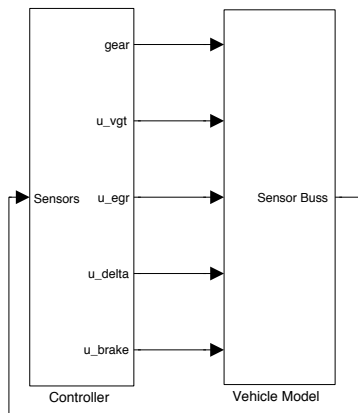


Fig. 1. Top view of the benchmark model showing the controller and vehicle model.

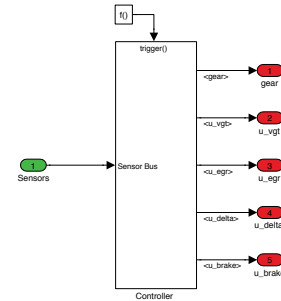


Fig. 2. Top level of the controller. At the top level there is the scheduling and at the lower level the simple controller. These are to be designed. Multiple timebases can be used.

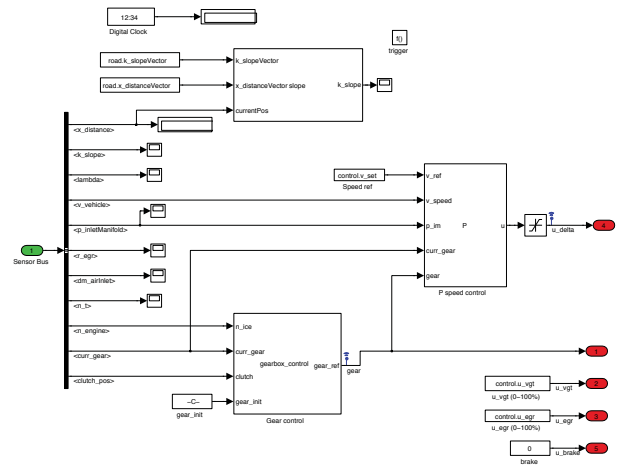


Fig. 3. Lower level of the controller. Here some simple controllers for one single time base have been implemented. The main task in the benchmark is to design these controllers.

3. CONTROLLER

We start the walk through of the models by going into the controller in Figure 1 as it is in here the main task for the benchmark problem is to be implemented. The controller is a link to the **Controller** block included in the **controller.slx** file. The controller has two levels and these are shown in Figures 2 and 3. In the example there is only one execution time base, 1 s, at the top level in Figure 2. This separation is done to illustrate that you can use several tasks to optimize CPU performance for example use a slower planning task with more heavy computations and a faster low level actuation controller.

The input to the controller is the sensor bus that contains the following signals (vehicle velocity v km/h, road slope α rad, distance driven s m, air fuel ratio λ -, inlet manifold pressure p_{im} Pa, egr ratio x_{egr} -, compressor air mass flow W_c kg/s, turbo speed n_t RPM, engine speed n_e RPM, actual gear u_g -, and clutch position -). The outputs from the controller are the main actuators u (gear request $u_{g,r}$ -, fuel injection u_δ mg/stroke, variable geometry turbine u_{vgt} %, egr valve u_{egr} %, brake u_b %).

The provided low level controller is not sophisticated and it only executes at sample time of 1 s. It has a simple gear changing logic that sends a gear request to the Transmission Control Unit TCU, after a request is sent it

waits until the new gear is engaged and the clutch released before a new gear change is allowed. As speed control a simple P-speed control is used to give the amount of fuel injected this is complemented with a simple smoke limiter to avoid too low λ :s. The EGR is closed and the VGT is set to a fixed position. The original controller provided with the benchmark download does not fulfill all necessary benchmark constraints, as there is no braking to prevent over speeding. Furthermore the road slope is not used either in the controller, so the included code is just a stub.

4. VEHICLE MODEL

The complete vehicle, shown in Figure 4, contains five modules: Engine, Clutch and Transmission, Transmission Control Unit (TCU), Chassi 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).

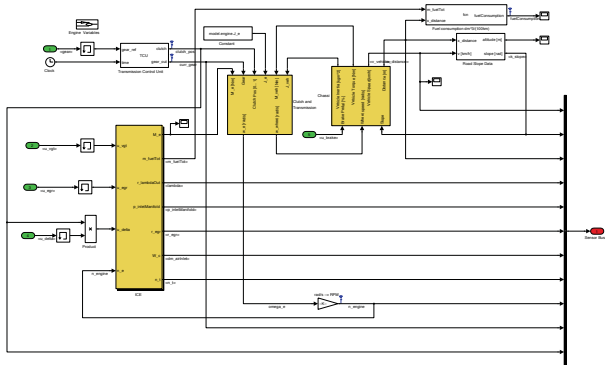


Fig. 4. Vehicle model with its five basic modules: Engine, Clutch and Transmission, Transmission Control Unit (TCU), Chassi, Road Slope.

	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	-
			W_c Compressor flow	kg/s
			n_t Turbo speed	rpm
Transmission	M_e Engine Torque	Nm	ω_e Engine Speed	rad/s
	M_w Wheel Torque	Nm	ω_w Wheel Speed	rad/s
	J_e Engine Inertia	kg m ²		
	J_w Vehicle Inertia	kg m ²		
	u_g Current gear	-		
	u_{cl} Clutch Position	0-1		
TCU	$u_{g,r}$ Requested gear	-	u_{cl} Clutch Position	0-1
	t time	s	u_g Current gear	-
Chassi	u_b Brake control	%	J_w Vehicle inertia	kg m ²
	ω_w Wheel speed	rad/s	M_w Wheel Torque	Nm
	α Road slope	rad	v Vehicle velocity	km/h
			s Travelled distance	m

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

4.1 Engine

The Diesel engine model with EGR and VGT is shown in Figure 5. It is based on the open source model LiU-Diesel developed in Wahlström and Eriksson (2011) that describes 12.7 liter, 6-cylinder diesel truck engine with EGR and VGT, it was 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 the cylinders there is an internal fuel limiter, calibrated so that it approximates the torque characteristics of the 400 hp engine. See Figure 6 for torque and power characteristics together with the Specific Fuel Consumption (SFC) characteristics.

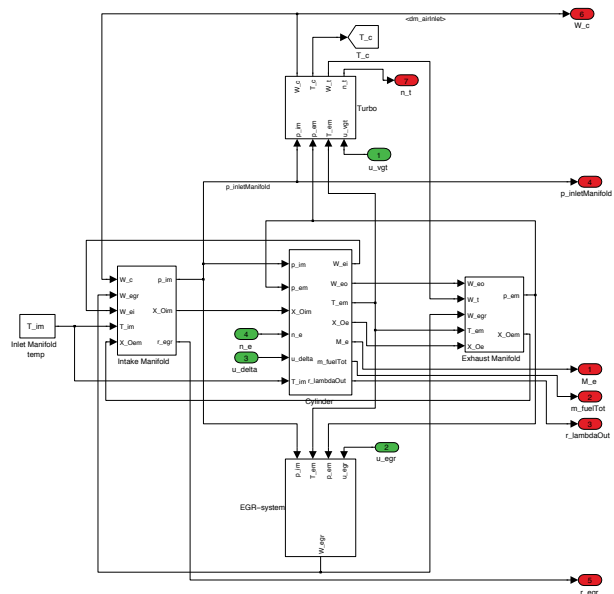


Fig. 5. The engine model with its five components: Cylinder, Intake manifold, Exhaust manifold, Turbo, and EGR System.

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 Eriksson and Nielsen (2014).

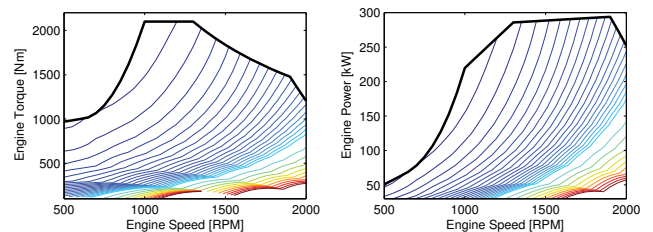


Fig. 6. The torque and power characteristics of the diesel engine model together with contours for the SFC. The SFC artifacts at low loads are from data grid generation and plotting.

$$\frac{d}{dt} p_{im} = \frac{R_a T_{im}}{V_{im}} (\dot{m}_c + \dot{m}_{egr} - \dot{m}_{ei}) \quad (2a)$$

$$\frac{d}{dt} p_{em} = \frac{R_e T_{em}}{V_{em}} (\dot{m}_{eo} - \dot{m}_t - \dot{m}_{egr}) \quad (2b)$$

$$\frac{d}{dt} X_{Oim} = \frac{R_a T_{im}}{p_{im} V_{im}} ((X_{Oem} - X_{Oim}) \dot{m}_{egr} + (X_{Oc} - X_{Oim}) \dot{m}_c) \quad (2c)$$

$$\frac{d}{dt} X_{Oem} = \frac{R_e T_{em}}{p_{em} V_{em}} (X_{Oe} - X_{Oem}) \dot{m}_{eo} \quad (2d)$$

where there are submodels for the different flows. The states for the oxygen fraction in the intake and exhaust manifold 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 ω_t , that is derived from Newton's second law

$$\frac{d}{dt} \omega_t = \frac{\dot{W}_t \eta_m - \dot{W}_c}{J_{tc} \omega_t} \quad (3)$$

where J_t 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 total mass flow \dot{m}_{ei} from the intake manifold into the cylinders is modeled using the volumetric efficiency η_{vol} (Heywood, 1988)

$$\dot{m}_{ei} = \frac{\eta_{vol}(p_{im}, N) p_{im} N V_d}{120 R_a T_{im}} \quad (4)$$

where p_{im} and T_{im} are the pressure and temperature in the intake manifold, N is the engine speed and V_d is the displaced volume. The volumetric efficiency is in its turn modeled as

$$\eta_{vol}(p_{im}, N) = c_{vol1} \sqrt{p_{im}} + c_{vol2} \sqrt{N} + c_{vol3} \quad (5)$$

The fuel mass flow \dot{m}_f into the cylinders is controlled by u_δ , which gives the injected mass of fuel in mg per cycle and cylinder

$$\dot{m}_f = \frac{10^{-6}}{120} u_\delta N n_{cyl} \quad (6)$$

where n_{cyl} is the number of cylinders. The mass flow \dot{m}_{eo} out from the cylinder is given by the mass balance as

$$\dot{m}_{eo} = \dot{m}_f + \dot{m}_{ei} \quad (7)$$

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

$$\lambda_O = \frac{\dot{m}_{ei} X_{Oim}}{\dot{m}_f (O/F)_s} \quad (8)$$

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.

4.2 Transmission with Clutch

The clutch and transmission shown in Figure 7 is modeled as an ideal transmission with stiff driveline, and friction clutch in between the two rotating systems. This system enables the engine to interact with the vehicle, it can for example during clutch slip or decoupling allow them to rotate independently or at clutch lock-up be locked

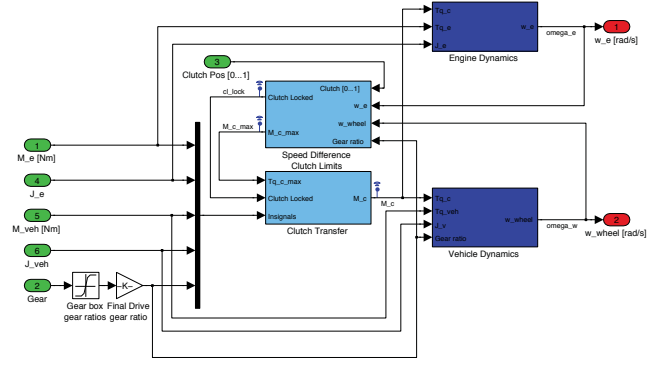


Fig. 7. Clutch and transmission model with its five components.

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 i_g connects the incoming and outgoing rotational velocities and torques

$$i_g = \omega_v / \omega_w, \quad i_g = T_w / T_v \quad (9)$$

where subscript v corresponds to the side connected to the clutch and w the wheel side. A mass with inertia I_w connected to the wheel side of the gears can be transferred to an *equivalent inertia* at the clutch side through

$$I_v = I_w / i_g^2 \quad (10)$$

The clutch position u_{cl} is connected to the clutch discs and produces a force that presses the clutch discs together, $F_c(u_{cl})$. Here $u_{cl}(t) \in [0, 1]$ and 0 means separated clutch and 1 means full force on the clutch discs. This force influences the maximum torque that the clutch can transfer $M_{c,k}(F_c(u_{cl}(t)))$. Here a simple linear model from the input u_{cl} is used

$$M_{c,k}(F_c(u_{cl}(t))) = M_{max,k} * u_{cl}(t) \quad (11)$$

In the expression above the kinetic friction $M_{c,k}$ is modeled, i.e. the clutch torque when the clutch disks are slipping. However, when the clutch is locked the clutch can transfer a higher (static) torque. The quotient between the static and dynamic friction is modeled as a constant, where

$$\frac{M_{c,s}(t)}{M_{c,k}(t)} = c_{s,k} \geq 1 \quad (12)$$

The maximum static friction torque is thus

$$M_{c,s}(F_c(u_{cl}(t))) = M_{max,k} c_{s,k} u_{cl}(t) \quad (13)$$

in the current Simulink implementation $c_{s,k} = 1$. For the clutch system there are two cases Clutch Slip and Clutch Lock that needs to be handled.

Clutch Slip - Two Mass System: The system has two masses that rotate with speeds independent of each other but their respective movements are connected to each other through to the transferred clutch torque. In this state the systems surrounding the clutch have two degrees of freedom and the equations that govern their rotations are

$$J_e \frac{d\omega_e}{dt} = M_e - M_c \quad (14a)$$

$$J_v \frac{d\omega_v}{dt} = M_c - M_v \quad (14b)$$

In slipping condition the torque transferred through the clutch is

$$M_{c,s} = M_{c,k}(f_c(u_{cl})) \operatorname{sgn}(\omega_e - \omega_v) \quad (15)$$

where sgn is the signum function. The equations above are valid when $\omega_v \neq \omega_e$.

Clutch Lock - One Mass System: The clutch disc locks the two rotating systems to each other and the two masses rotate with identical speed. Under these conditions the rotation of the surrounding two masses have only one degree of freedom and the equations that govern the rotations are

$$\begin{aligned} (J_e + J_v) \frac{d\omega_e}{dt} &= M_e - M_v \\ \omega_v &= \omega_e \end{aligned} \quad (16)$$

When the clutch is locked the torque transferred in the clutch $M_{c,l}$ depends on the applied torques and the inertias. The transferred torque is the driving torque minus the inertia effect of the first mass

$$M_{c,l} = M_e - J_e \frac{d\omega_e}{dt}$$

Inserting Equation (16) and manipulating gives the clutch torque

$$M_{c,l} = \frac{M_e J_v + M_v J_e}{J_e + J_v} \quad (17)$$

This torque is compared to the maximum static friction (13) to determine when clutch break up occurs.

In the Simulink implementation the system is simulated using the two state system (14) with two versions of the clutch transmitted torque depending on if the clutch is locked (17) or slipping (15). See Eriksson (2001) for more details about the lock-up and break apart logic.

4.3 Transmission Control Unit (TCU)

The transmission control unit (TCU) shown in top left of Figure 4 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 Figure 8 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 0.5 s, then a speed synchronization phase is assumed to last 1 s. Finally the clutch and fuel injection are ramped up again for 0.5 s. In the current model implementation there is no speed synchronization implemented, fuel is only cut-off over the period.

4.4 Chassi

The vehicle chassi model is shown in Figure 9, it contains the three submodels, air drag, rolling resistance, and road slope and interacts with the transmission and clutch model. The wheel is assumed to follow the rolling condition meaning that the vehicle velocity and wheel speed are connected algebraically, i.e. $v = r_w \omega_w$. 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:

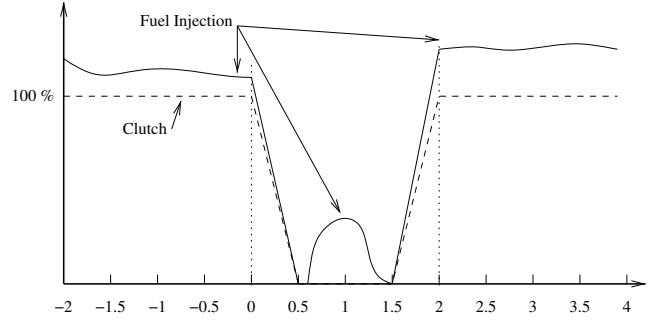


Fig. 8. Sketch of a profile for clutch and fuel injection during a gear change with. The total time for the gear change is 2 seconds. In the implemented model there is no speed synchronization so the fuel injection is kept at 0 from 0.5 s to 1.5 s after a commanded gear change.

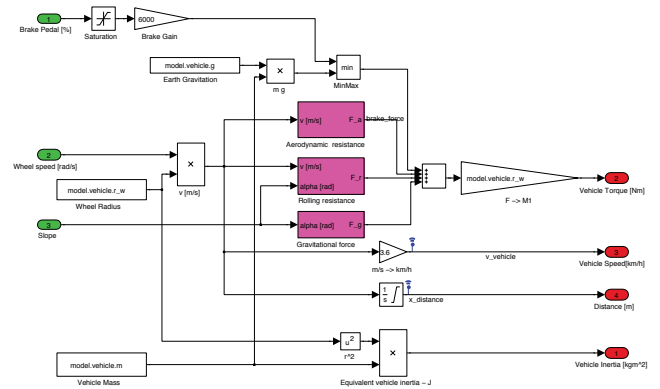


Fig. 9. Chassi model with the three main forces that act on the vehicle and give rise to accelerations or decelerations.

$$F_w = F_r + F_a + F_g \quad (18a)$$

$$F_r = m g \cos(\alpha)(c_{r,0} + c_{r,1} v) \quad (18b)$$

$$F_a = \frac{1}{2} c_w A \rho v^2 \quad (18c)$$

$$F_g = m g \sin(\alpha) \quad (18d)$$

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 $M_w = F_w/r_w$. 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, see Section 4.2. So the clutch provides the wheel speed ω_w for the Chassi model and the Chassi returns the loading torque at the wheels $M_w = F_w/r_w$.

5. SCENARIO DEFINITION

The scenario definition is given as a Matlab workspace variable in the format of a structure named `scenaro` it is pre-loaded before the `runSim.m` script is called. The contents of the `scenaro` structure is:

- `v_init` Initial speed of the vehicle $v(0)$ km/h.
- `gear_init` Initial gear number.
- `vehicleMass` Vehicle mass m kg.
- `v_set` Cruise controller set speed v_{set} km/h.

- `h_0` Altitude of the starting point $h(0)$ m.
- `road` Structure containing grid points and slopes
 - `distanceVector` grid points in of the driving distance in m.
 - `k_slopeVector` the road slope at the grid points in rad.

The provided scenario is from the Swedish E4 highway from Södertälje to Norrköping a distance of approx 120 km. The route is marked on the map in Figure 10.

6. BENCHMARK TASK

The benchmark task is to implement a controller that minimizes the fuel consumption over a driving mission, with the option of using map with road slope data. It is possible to use the model parameters but it is not allowed to modify the Vehicle model or its parameters.

6.1 Constraints

The optimal control problem with constraints were described in Section 1 in Equations (1a)-(1h).

Problem settings for the scenario will be setup outside and provided as a struct in the Matlab workspace called `scenario` to the script `runMe.m`. In the scenario we have desired average vehicle velocity (cruise controller setpoint) v_{set} , the driving profile as a vector of positions and road slopes, the vehicle mass m that can vary between 7000 kg (empty) and 60000 kg loaded to the legal limit. This is known by controller so it can do its planning. The scenario also contains an initial altitude, it is mostly included for completeness so that the altitude profile can be plotted.

Some cases Additional points will be gained by providing a **fallback mode** that handles the case if the GPS is not available or if map data is not available for the mission. In such a case the workspace variable defining the scenario will indicate this by providing `scenario.road=NaN`. In this case the vehicle will report instantaneous road slope information and position during the simulation execution but no look-ahead data will be provided.

The scenario start speed v_{start} might be 0 km/h (e.g. starting from a distribution centre or road crossing), or any arbitrary X km/h $X \in [0, 90]$ enabling the scenario that the driver activates the controller during driving. A starting gear is also provided in the scenario. In both scenarios we allow cloud computing.

6.2 What to deliver?

The solution to the benchmark is to be provided in a single zip-file.

- The controller to be delivered must be a block called `Controller` that is a part of a Simulink library called `controller.mdl` or `controller.slx`.
- A `runSim.m` file that does your controller pre-computations and calls `sim('TruckBenchmark')`.
- All your needed support files with data and scripts.
- Submit all needed things in a ZIP-file.

Criteria	Weight
Fuel consumption	60
Cloud computing time	5
Controller execution time	15
Controller memory usage	10
Fall back	10
Industrial ranking	20
Academic ranking	10
Longer trip time	DNQ
Smoke limiter	DNQ
Turbo overspeeding	DNQ

Table 2. Criteria for evaluations, relating the problem definition and constraints. DNQ means a hard constraint if these are not fulfilled then the controller Does Not Qualify.

7. BENCHMARK EVALUATION

The benchmark will be evaluated on the given Södertälje-Norrköping road profile. Robustness of the provided solutions will be evaluated using some other road profiles and vehicle weights that will be secret but supplied in the scenario description.

The evaluation will be set up like this:

- (1) Your provided solution with library, functions, and data will be stored in a directory.
- (2) The directory will be added to the Matlab path.
- (3) A scenario will be constructed providing the scenario definition, according to the possibilities outlined in Section 6.
- (4) Your file `runSim.m` will be called and it should call `sim('TruckBenchmark')` that will be available in our path. Our model `TruckBenchmark.mdl` will use your block `Controller` from your library `TruckBenchmark.mdl`.

The criteria and their importance for the ranking are specified in Table 2. There are two subjective rankings where the industrial members of the jury will provide a ranking of the solutions and the academic members will provide a ranking of the solutions.

7.1 Evaluation procedure

To qualify at all the solution must fulfill the hard constraints (trip time, smoke limiter, turbo overspeeding). When that is assessed each solution will be graded and related to the other solutions through the following procedure, inspired by the Eurovision Song contest: Lets say that there are 9 solutions contributed. The winner in a category will get 12 points, the second will get 10 points, the third 8 points, the fourth 7 points the fifth 6 points and so on down to 1 point for the last position. This highlights that it is beneficial to win a certain category. If the number of contributed solutions are higher than the maximum points will be increased so that no one will get 0 points.

For each category the points will be multiplied by the weight described in Table 2 and the winner will be the one gets the highest sum of products between points and weight. The weights sum up to 130 points so it is not percent but since all get access to the same number of points we stick to the numbers given. If by any chance two

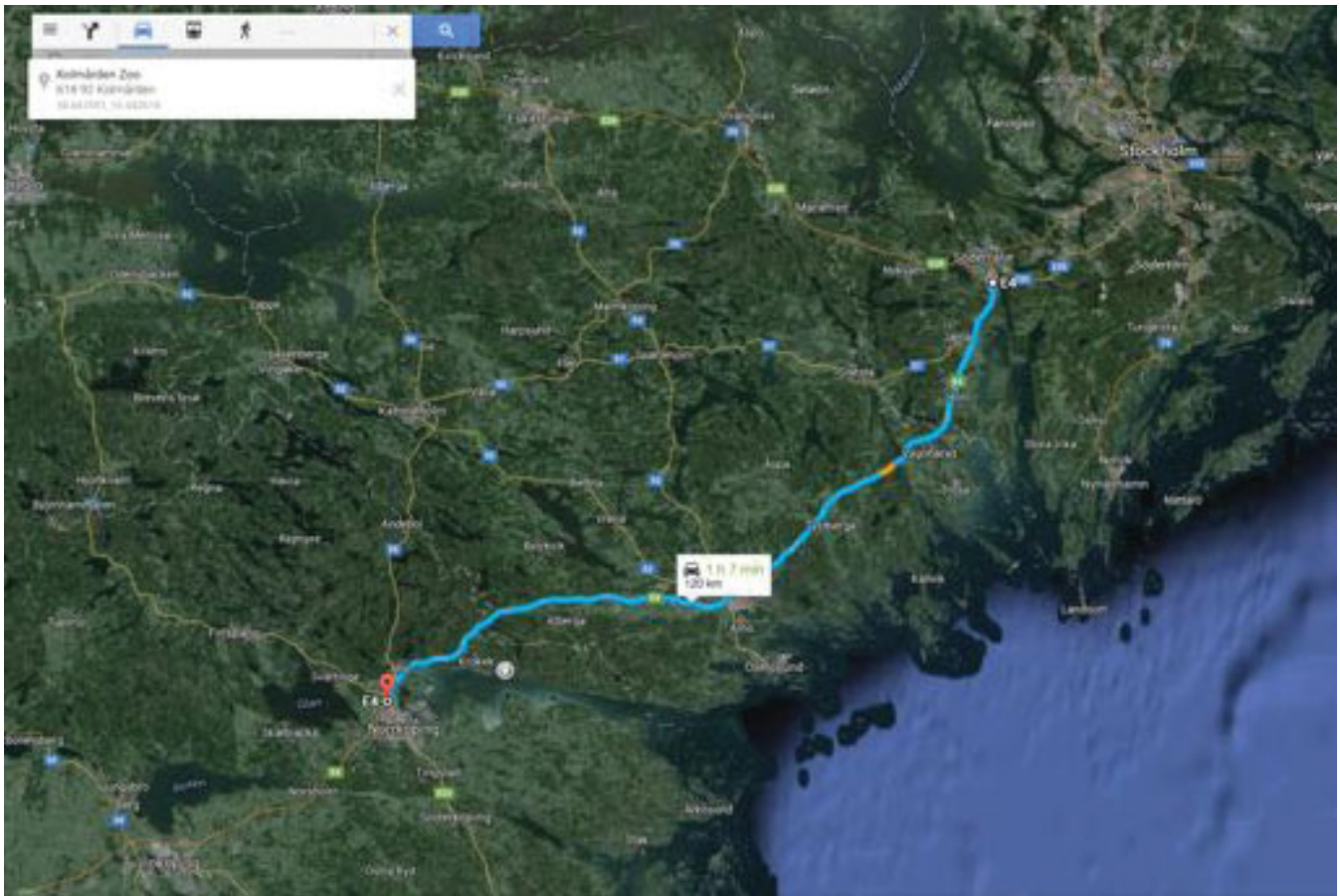


Fig. 10. Map showing the location of the road in the benchmark problem also marked is the AAC2016 conference venue near the Kolmården Zoo.

solutions end up with the same sum of weighted points the division line will be taken from the subjective industrial ranking, this to highlight the high industrial interest in the problem. The winning solution will be announced at the AAC 2016 symposium, we can only guarantee that the award to the winning team will be the honor of winning the contest (but we are actively searching for sponsors).

8. CONCLUSION

A benchmark problem for look ahead control of a heavy duty truck on an open road has been defined. The benchmark template consists of a complete vehicle model complemented with a driving mission with road slope data. The task is to design a fuel efficient controller for a given driving mission while several criteria have to be fulfilled. Some weights for the evaluation of the controller performance are also provided.

ACKNOWLEDGEMENTS

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

REFERENCES

- Eriksson, L. (2001). Simulation of a vehicle in longitudinal motion with clutch engagement and release. In *3rd IFAC Workshop "Advances in Automotive Control" Preprints*. Karlsruhe, Germany.
- Eriksson, L. and Nielsen, L. (2014). *Modeling and Control of Engines and Drivelines*. John Wiley & Sons.
- Heywood, J.B. (1988). *Internal Combustion Engine Fundamentals*. McGraw-Hill series in mechanical engineering. McGraw-Hill.
- Ivarsson, M., Åslund, J., and Nielsen, L. (2010). Impact of AMT gear-shifting on fuel optimal look ahead control. In *SAE World Congress*, number 2010-01-0370 in SAE Technical Paper Series. Detroit, MI, USA.
- Myklebust, A. (2014). *Dry Clutch Modeling, Estimation, and Control*. Ph.D. thesis, Linköping University.
- Myklebust, A. and 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*. Tokyo, Japan.
- 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. and 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.