

A Driver Model for Virtual Drivetrain Endurance Testing

Tilman Bunte
German Aerospace Center (DLR)
Institute of Robotics and Mechatronics, Germany
Tilman.Buente@dlr.de

Emmanuel Chrisofakis
Daimler AG
Stuttgart, Germany
emmanuel.chrisofakis@daimler.com

Abstract

Starting from an assumed vehicle path on a given road section we derive the formulae for the calculation of an appropriate reference speed profile tabled over road arc length. Together with a speed error feedback we thus emulate what a real driver does while driving and scheduling his actions on throttle and brake pedal. The resulting driver model may be used for automatic speed control in vehicle dynamics simulation. The application addressed here is software in the loop simulation for virtual drivetrain endurance testing at Daimler AG. A prototypical Modelica implementation was made at DLR and tested with a simple longitudinal vehicle dynamics model. Finally, we discuss the experiences with the reference speed profiles made in the industrial practice.

Keywords: *driver model; automatic gearbox testing; reference vehicle speed profile; software in the loop (SiL); virtual drivetrain endurance testing.*

idea is continued and re-engineered here to obtain an adequate complexity of the solution for the given problem.

According to the split subtask notion, the second subtask is accomplished by using feedback of the speed error to desired longitudinal vehicle acceleration. Based on the latter appropriate gas and brake pedal positions can be determined. Meaningful parameterization allows for assigning typical driver types like *cautious*, *normal*, *sportive* or *risky*. The driver model is implemented in Modelica; the speed profile calculation is done at initialization time.

The paper is organized as follows. The assumptions and the theoretic background of the speed profile generation are described in section 2. Section 3 is dedicated to the implementation of a speed feedback controller. Some simulation results with the model depicted in Figure 1 are shown in section 4. Experimental results from virtual automatic gearbox test runs are presented in section 5 including a report on one's experiences.

1 Introduction

For the virtual endurance test of automatic gearboxes realistic and repeatable load collectives are searched for. Therefore, a given road is assumed in terms of slope, crossfall, curvature, road adhesion coefficient, and speed limits along the path of the road centerline. Adequate input signals for throttle and brake pedal are needed to drive a total vehicle model along the road while imitating realistic driver behavior. The driver model task means providing suitable pedal position signals. In our approach, the driver model is split into two sequential subtasks. Firstly, a vehicle speed profile along the road arc length is calculated regarding the road conditions with a sufficient preview. This reference speed profile is supposed to approximate a speed profile which a driver (usually unconsciously) forms in his mind yielding a set point for subsequent speed control. Speed profiles have already been used in the context of various driver speed assistance systems such as [1], [2]. The

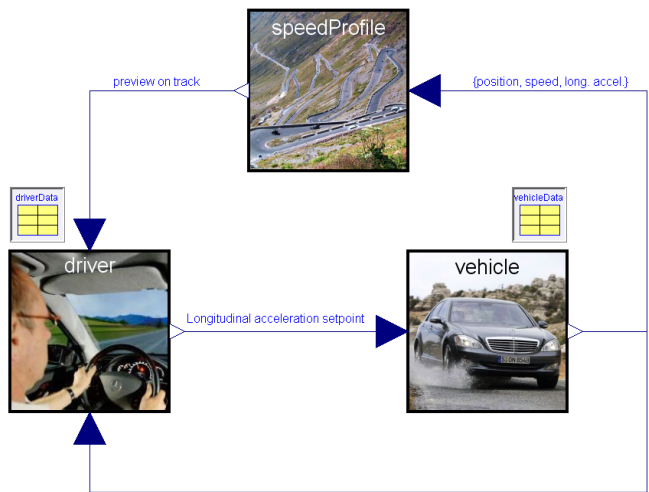


Figure 1: Total Modelica model for driver model evaluation

2 Speed profile generation

The derivation of a reference speed profile is according to the following conception: At the end of a preview horizon the vehicle should come to a standstill. The preview horizon may be set arbitrarily a certain distance ahead or e.g. formed by the end of the current visual range, the next road junction, and/or by an obstacle. On the way to this stop the vehicle's speed is scheduled to be maximal, however, such that all traffic regulations and physical limitations are met with certain margins. These include the limited lateral acceleration in curves, reduced deceleration capability while downhill or curve riding due to combined longitudinal/lateral tire forces, speed limits, and so on. The margins are adapted according to the driver type. In summary: Like real drivers do, the speed is scheduled virtually along the reverse path starting from a limitation arising ahead.

2.1 Assumptions

The road definition is assumed to be given in terms of slope $\partial z(s)/\partial s$, crossfall $\partial z(s)/\partial w$, curvature $\rho(s)$, road adhesion coefficient $\mu(s)$, and speed limits as functions of a single parameter being the road arc length s . The variable's dependencies of s (also denoted *position* in the sequel) are omitted in the formulae as from now.

The road position $[x, y, z]^T$ and the heading angle φ belonging to any value of the arc length s and any lateral displacement w from the road centerline can be calculated by numeric integration based on the following ordinary differential equations

$$\frac{\partial \varphi}{\partial s} = \rho, \quad \frac{\partial x}{\partial s} = \cos(\varphi), \quad \frac{\partial y}{\partial s} = \sin(\varphi) \quad (1)$$

and adequate start conditions. Road slope and crossfall should be so small such that errors from linearization of associated trigonometric functions are negligible. A basic supposition adopted here is that the total horizontal force $|F_{sw}|$ transmitted between the collectivity of all tires and the road is limited isotropically,

$$|F_{sw}| = \sqrt{F_s^2 + F_w^2} \leq \mu \cdot m \cdot g, \quad (2)$$

where g is the gravitational acceleration, F_s is the force in travel direction, and F_w represents the lateral force.

The disposition of the driver to utilize the physical force limits in longitudinal or lateral direction is reflected by the driver behavioral parameters κ_s and κ_w , respectively, each with $0 \leq \kappa_{s,w} \leq 1$ and specific

values depending on the driver type. Evolving from (2), the expression

$$\frac{\sqrt{\left(\frac{F_s}{\kappa_s}\right)^2 + \left(\frac{F_w}{\kappa_w}\right)^2}}{\mu \cdot m \cdot g} \leq 1 \quad (3)$$

is denoted the *driver related degree of utilization of force transmission quota* which usually is persistently changing while driving. The associated inequality is the fundamental relation [1] later used for the calculation of speed profiles.

2.2 Forces acting on the vehicle

For the calculation of the reference speed profile the vehicle is considered a point mass. Therefore, vehicle dynamics such as yaw, roll, pitch and heave motion plus their effect on the tire forces are neglected. The speed of the point mass vehicle is $v = ds/dt$, the longitudinal force may be expressed as

$$F_s = m \left(\dot{v} + \lambda \cdot v \cdot |v| + g \cdot \left(k_R(v) - \frac{\partial z}{\partial s} \right) \right) \quad (4)$$

where $k_R(v)$ is the vehicle's rolling resistance coefficient and

$$\lambda = \frac{\rho_L c_w A}{2 \cdot m} \quad (5)$$

is a parameter related to aerodynamic drag defined only for abbreviation of math terms. Here, ρ_L is the air density, c_w is the drag coefficient, A is the face surface, and m is the total mass of the vehicle.

The lateral force is

$$F_w = m \left(\rho \cdot v^2 + \frac{\partial z}{\partial w} \cdot g \right). \quad (6)$$

Note that the point mass assumption does not hold for highly dynamic manoeuvres which may result from *risky* driver behaviour. In this case, it can not be guaranteed that the real vehicle would still be able to follow the speed profile.

2.3 Constraints on longitudinal dynamics

The calculation of reference speed profiles is determined by a set of constraints on the longitudinal dynamics of the vehicle which are presented below.

Static speed limits

An upper static bound on the speed is obtained when solving (3) for v after inserting (6) and $F_s = 0$:

$$v \leq \sqrt{\frac{\kappa_w \cdot \mu \cdot g}{|\rho|} - \frac{g}{\rho} \cdot \frac{\partial z}{\partial w}} \quad (7)$$

This is the local maximum speed without making skidding off the road in a curve. Or, to be more precise, the portion of it the driver is accepting.

Another bound reflects the collectivity of all conceivable speed limitations such as legal speed limits, deliberate speed reduction or any other arbitrary speed constraint:

$$v \leq \frac{\kappa_f}{\kappa_v} \cdot v_{speedlimit} \quad (8)$$

The behavioral parameter κ_f reflects the driver's disposition towards this constraint category. The value $\kappa_f = 1.1$ means that the driver is ready to excess speed limits by ten percent. The parameter κ_v in (8) will be cancelled later (see (23)) and is of no relevance here. The static upper speed limit, in summary, is the smaller of the two limits calculated by (7) and (8):

$$v \leq v_{max,stat}$$

with

$$v_{max,stat} = \min \left\{ \sqrt{\frac{\kappa_w \cdot \mu \cdot g}{|\rho|} - \frac{g}{\rho} \cdot \frac{\partial z}{\partial w}}, \frac{\kappa_f}{\kappa_v} \cdot v_{speedlimit} \right\} \quad (9)$$

Acceleration limits

The limited engine power P_{max} imposes an upper bound on the acceleration. Depending on the driver type the power limit is exploited by a fraction κ_p with $0 \leq \kappa_p \leq 1$, thus

$$F_s \cdot v \leq \kappa_p \cdot P_{max} \quad (10)$$

holds. After insertion of (4) and solving for the acceleration we get

$$\dot{v} \leq \dot{v}_{max,P} = \frac{\kappa_p \cdot P_{max}}{v \cdot m} + c \quad (11)$$

with

$$c = -\lambda \cdot v \cdot |v| - g \cdot (k_R(v) - \partial z / \partial s). \quad (12)$$

A valid interval for the vehicle's acceleration can be obtained from transformation of (3) and consideration of (4), (6), (11), and (12):

$$c - d \leq \dot{v} \leq c + e \quad (13)$$

with

$$d = g \frac{\kappa_s}{\kappa_w} \sqrt{\kappa_w^2 \cdot \mu^2 - \left(\frac{\rho \cdot v^2}{g} + \frac{\partial z}{\partial w} \right)^2} \quad (14)$$

and

$$e = \min \left\{ d, \frac{\kappa_p \cdot P_{max}}{v \cdot m} \right\}. \quad (15)$$

2.4 Numeric speed profile calculation formulae

On a sufficiently small section $s_0 \leq s \leq s_I$ of the road path where the longitudinal acceleration can be assumed constant the following equation holds:

$$s_I = s_0 + v_0 \cdot \Delta t + \frac{\dot{v}}{2} \Delta t^2 \quad (16)$$

Here, Δt is the time needed to drive along the road section and v_0 is the initial speed at $s = s_0$. Moreover, the speed v_I when reaching $s = s_I$ is

$$v_I = v_0 + \dot{v} \cdot \Delta t. \quad (17)$$

Depending on whether v_0 or v_I is given, after elimination of Δt from the set (16), (17) we obtain meaningful solutions for the other variable

$$v_I = \sqrt{v_0^2 + 2 \cdot \dot{v} \cdot (s_I - s_0)} \quad (18)$$

or

$$v_0 = \sqrt{v_I^2 - 2 \cdot \dot{v} \cdot (s_I - s_0)}. \quad (19)$$

This allows for a simple numeric integration algorithm (explicit Euler) for the calculation of reference speed profiles along the arc length s

$$v_{(k+1)} = \sqrt{v_{(k)}^2 + 2 \cdot \dot{v}_{(k)} \cdot (s_{(k+1)} - s_{(k)})} \quad (20)$$

or

$$v_{(k-1)} = \sqrt{v_{(k)}^2 - 2 \cdot \dot{v}_{(k)} \cdot (s_{(k)} - s_{(k-1)})} \quad (21)$$

for the reverse direction, respectively. For this purpose it is required that the road path information is given with sufficiently high resolution along s such that the assumption of constant longitudinal acceleration between the grid points is justified.

2.5 Numeric maximal speed profile calculation

The *maximal speed profile* denotes the speed profile $v_{\max}(s)$ along a considered road section exhibiting the maximum possible speed at all positions s while respecting the following constraints:

- The start speed at the road section beginning is $v_{\max}(s_{\text{start}}) = v_{\text{start}}$.
- The final speed at the road section end is $v_{\max}(s_{\text{end}}) = v_{\text{end}}$.
- At every position $s_{\text{start}} \leq s \leq s_{\text{end}}$ the inequalities (3), (9) with $\kappa_v = 1$, and (13) hold.

Hence, the maximal speed profile is a candidate for a reference speed along the considered road section to be used for speed control. It may, of course, be further processed according to one's needs.

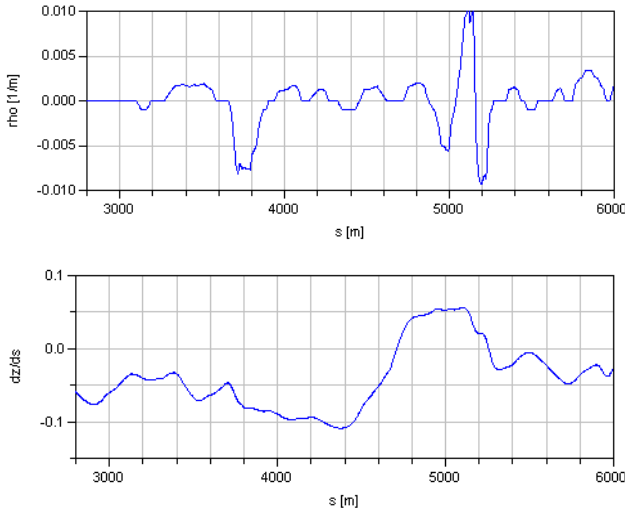


Figure 2: The road definition used for illustration in sections 2-4: $\rho(s)$ and $\partial z(s)/\partial s$ as displayed; $\mu = 1$, $\partial z(s)/\partial w = 0$, $v_{\text{speedlimit}} = 41.7\text{m/s}$.

In the sequel the procedure for calculating a tabled representation of the maximal speed profile is presented. In multiple steps the speed profile is reduced by considering new constraints at each time. See Figure 3 for illustration. The underlying road definition is given with Figure 2. The *normal driver* type was chosen; see Table 1 in section 4.2.

1. The static upper speed limit $v_{\max, \text{stat}}$ is calculated using (9) for all given sampling points of s .
2. Starting from the end of the road section s_{end} an interim profile $v_{\max, \text{back}}$ is calculated. Therefore, the recursive formula (21) is applied over all sampling points of s and $v_{\max, \text{back}}(s_{\text{end}}) = v_{\text{end}}$ is used as start value. With each integration step the value of \dot{v} is

set to $\dot{v} = c - d$ being the maximum deceleration (i.e. minimum acceleration) according to (13). Therefore, in (12) and (14) the current values for all varying quantities are inserted. During the recursive procedure $v_{\max, \text{back}}$ must be always limited to the static upper speed limit $v_{\max, \text{stat}}$. The resulting interim speed profile provides a necessary condition such that the vehicle starting with $v_{\max, \text{stat}}$ at any position can decelerate down to $v(s=s_{\text{end}}) \leq v_{\text{end}}$ while always respecting the inequalities (3), (9).

3. Not only when braking, also when accelerating the constraints must be fulfilled. Therefore, the previous step is repeated, however, in forward direction resulting in a new interim speed profile $v_{\max, \text{forw}}$. Formula (20) is used for recursive integration from the start value v_{start} at s_{start} . The acceleration is set to its current maximum value $\dot{v} = c + e$. Note, that $v_{\max, \text{forw}}$ mustn't exceed the previously calculated $v_{\max, \text{back}}$ in order to keep that information.
4. The finite difference equations (20) and (21) respectively which were used in the two previous steps are based on the assumption of constant acceleration between two sampling points. Depending on the effective gridding this assumption may be violated. In this case the gridding needs to be refined by inserting new sampling points where needed. This should be repeated until the resulting acceleration error is less than a predefined tolerance. Note that steps 3 and 4 also need to be repeated in that grid refinement loop.

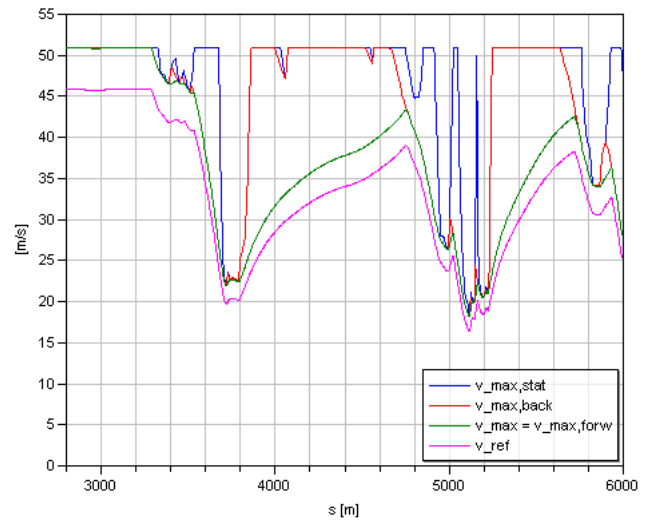


Figure 3: Calculation of the maximal speed profile and definition of a reference speed profile

The decisive maximal speed profile, finally, is

$$v_{\max}(s) = v_{\max, \text{forw}}(s). \quad (22)$$

If the vehicle exactly follows this speed profile then it drives at maximum speed while respecting all physical limits plus considering driver type dependent safety, comfort, and economy relevant margins.

2.6 Reference speed profile definition

The before calculated maximum speed profile can be adopted as a base for the definition of a *reference speed profile* which is suitable for speed control of the vehicle. As an example, linear scaling is applied

$$v_{\text{ref}}(s) = \kappa_v \cdot v_{\max}(s) \quad (23)$$

using a driver type dependent parameter κ_v with $0 \leq \kappa_v \leq 1$ (compare Figure 3 with $\kappa_v = 0.9$).

3 Speed control based on reference speed profiles

3.1 Using acceleration as control variable

The aim of speed feedback control is to make the error between reference speed and actual vehicle speed small. For adjustments of the vehicle speed, accelerations in the interval given by (13) are permitted. Accordingly, a reference longitudinal acceleration a_{ref} is formed by the controller. In a successive module which is not discussed here, the reference acceleration can be transformed into gas and brake pedal positions as accurately as possible e.g. by using nonlinear inverse static or inverse dynamic models [3]. Any speed errors resulting from model inaccuracies or induced by disturbances can be compensated for by the speed feedback control which is described as follows. It turned out that proportional feedback of the speed control error yields satisfactory results, even if the resulting control variable a_{ref} is limited according to \dot{v} in (13). Before being more precise with this issue, prediction of the speed error is introduced.

3.2 Prediction of the speed error

Significant control performance improvement can be achieved by compensation of plant delay. A parameterizable prediction time T_{pred} takes into account summarized lags which may be present in the control loop such as power train or brake dynamics. Hence,

both the reference speed and the vehicle speed are predicted by T_{pred} in advance. The approximation used here assumes that during the prediction time the acceleration remains constant.

The predicted reference speed is simply determined by evaluation of (23)

$$v_{\text{ref}, \text{pred}} = v_{\text{ref}}(s_{\text{pred}}) \quad (24)$$

at the predicted vehicle position

$$s_{\text{pred}} = s_{\text{veh}} + v_{\text{veh}} \cdot T_{\text{pred}} + \frac{1}{2} \dot{v}_{\text{veh}} \cdot T_{\text{pred}}^2. \quad (25)$$

On the other hand, the predicted vehicle speed is

$$v_{\text{pred}} = v_{\text{veh}} + \dot{v}_{\text{veh}} \cdot T_{\text{pred}} \quad (26)$$

The prediction time may also be considered a driver type dependent parameter. If no prediction virtue is wanted then T_{pred} can simply be set to zero.

3.3 Limited proportional feedback

As mentioned before the speed control uses feedback of the predicted speed error

$$a_{\text{ref}, \text{raw}} = \kappa_g \cdot (v_{\text{ref}, \text{pred}} - v_{\text{pred}}). \quad (27)$$

The driver type dependent parameter κ_g is the feedback gain. Finally, the controller must respect the acceleration limits (13):

$$a_{\text{ref}} = \begin{cases} c_{\text{pred}} - d_{\text{pred}} & \text{if } a_{\text{ref}, \text{raw}} < c_{\text{pred}} - d_{\text{pred}} \\ c_{\text{pred}} + e_{\text{pred}} & \text{if } a_{\text{ref}, \text{raw}} > c_{\text{pred}} + e_{\text{pred}} \\ a_{\text{ref}, \text{raw}} & \text{otherwise} \end{cases} \quad (28)$$

Reasonably, all variables take on their values at the predicted position $s = s_{\text{pred}}$. Note that if this kind of limited feedback (28) is used, then the limitation of the speed profile in forward direction is redundant and $v_{\max}(s) = v_{\max, \text{back}}(s)$ should be used rather than (22).

4 Simulation results

4.1 Modelica implementation

Figure 1 shows the total Modelica model we built during implementation and prototype testing of the driver model together with a very simple vehicle model. The speed control given by (27) is implemented in the block *driver*. The block *vehicle* uses the longitudinal acceleration as requested from *driver* as input. It consists of a first order lag element for representation of the power train / brake dynamics. The time constant is set unrealistically high to

$T_{lag} = 1.0$ s to demonstrate clearly the benefit of the speed error prediction concept. Two successive integrators compute speed v_{veh} and position s_{veh} , respectively.

The prime block *speedProfile* needs the tabled road definition as parameter. From that, the maximal speed profile is calculated in multiple steps as explained in section 2.5. A corresponding function is executed at model initialization time and stores the result in a parameter table. Therefore, also driver type and vehicle parameters are needed which are instantiated as records in the total model. During the simulation, the block *speedProfile* provides the predicted quantities needed for (27), (28). To facilitate their calculation all relevant variables are evaluated on the base of the predicted vehicle position (25) and speed (26). The tabled road data and the pre-calculated reference speed profile are correspondingly interpolated.

4.2 Prototype simulation

This section shows simulation results obtained with the prototypical implementation from Figure 1. The vehicle starts at $s_{veh} = 2900$ m and $v_{veh} = 0$. The *normal driver* record with the parameters given in Table 1 was used.

Table 1: *Normal driver* type parameters

| | | | |
|------------------|------------------|------------------------------|------------------|
| $\kappa_s = 0.4$ | $\kappa_w = 0.4$ | $\kappa_v = 0.9$ | $\kappa_f = 1.1$ |
| $\kappa_g = 10$ | $\kappa_p = 0.6$ | $T_{pred} = T_{lag} = 1.0$ s | |

In Figure 4 the blue line is the pre-calculated reference speed profile (cf. Figure 3). The red line is the simulated predicted reference speed. We find that the simulated vehicle speed (green line) matches very well the reference speed.

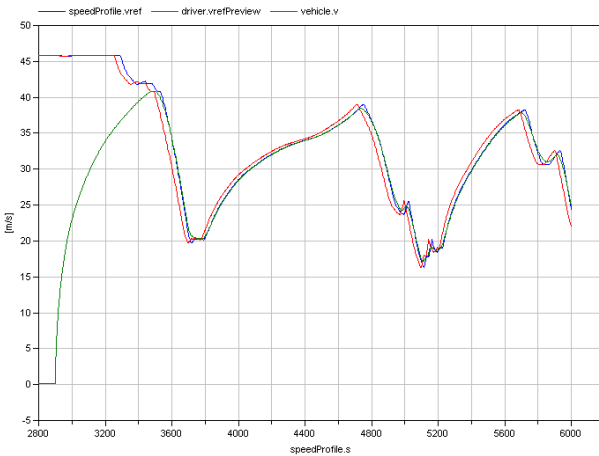


Figure 4: Speed profiles and actual vehicle speed in the simulation

Only at the beginning there is a big gap which is conditioned by the limited acceleration. Figure 5 shows the effective acceleration limits (blue, red) according to (13) and the actual vehicle acceleration (green line). The acceleration potential is fully exploited in the initial phase while there is a large speed error. Later, the driver model keeps some margin from the limits which is due to our choice

$$\kappa_v = 0.9 \text{ in (23).}$$

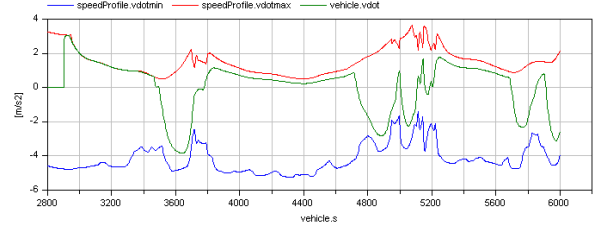


Figure 5: Acceleration limits and actual vehicle acceleration in the simulation

The *driver related degree of utilization of force transmission quota* from (3) is shown in Figure 6 with a blue line. As a consequence of our approach, it must never exceed one. The *physical degree of utilization of force transmission quota* is plotted as a red line. It is obtained by setting $\kappa_s = 1$ and $\kappa_w = 1$ in (3) and thus removing the driver type dependent implicit safety margin.

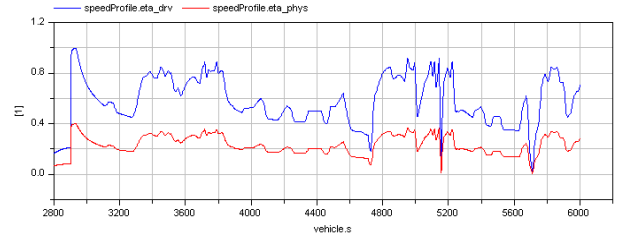


Figure 6: Utilization of force transmission quota in the simulation

5 Experimental results and applicational issues

The Modelica driver model was evaluated at Daimler and found suitable for the purpose of virtual drivetrain endurance tests. Thereupon, the driver model was deployed in a *software in the loop* environment (SiL) in conjunction with a detailed plant model. The functional code in the loop is the control code of an automatic Mercedes-Benz gearbox transmission. The used plant model describes the longitudinal dynamics of a vehicle and has its modeling focus on the 1-D rotational dynamics of the drivetrain.

Figure 7 shows a top-level screenshot of the model [6]. The calculation of the reference speed profile (23) as described in section 2 serves as reference speed and was integrated in the overall car model (Figure 7) which in turn was exported as a DLL for the SiL environment. For the code export we used the C-Code generated by Dymola 6.2 wrapped with an API for the co-simulation tool BACKBONE, a proprietary Daimler program.

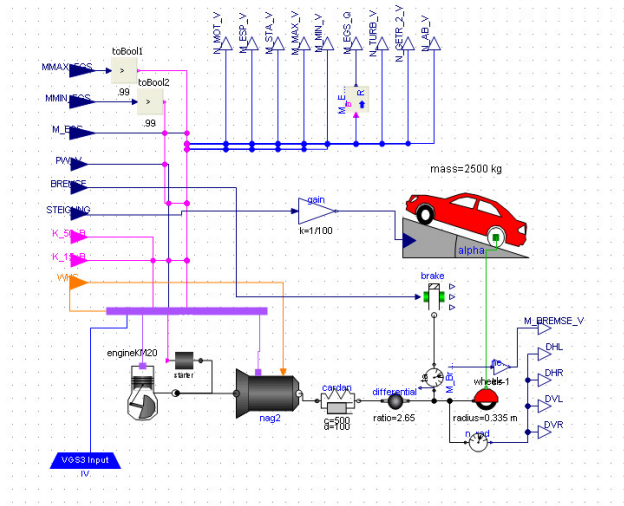


Figure 7: Modelica car model for SiL export

For SiL control of this model at Daimler an enhanced feedback control was used rather than (27). It also accomplishes the assignment to both throttle and brake pedal based on PI-control of the vehicle speed error. The driver specific parameters used for the calculation of the speed profile turned out to be useful for the calibration of the total SiL driver model.

This specific deployment of the SiL is used for virtual endurance testing of the drivetrain. The tracks we use are the same that our testing teams drive in reality. With the virtualisation we are able to

- examine the impact of code updates on the endurance of the hardware (gearbox and drivetrain components),
- detect bugs in the code, and
- calculate load collectives.

All this can be done in a fast and absolutely low cost manner. So far, at Daimler, simulation of load collectives for gearboxes primarily had taken place with special software which, however, didn't include the functional code. SiL simulations of this kind had been done by using a fixed speed profile derived from experimental measurements or a load collective simulation. The reference speed input to the SiL was therefore car specific and could not be used for other vehicle configurations. Moreover, the reference

speed had been time scheduled rather than position scheduled. Hence, a cumulative error in the calculation was unavoidable due to the deviation between desired and actual speed: After some simulation time on long tracks (some 100 km) the vehicle's position did not match the position the reference speed was assigned to. As a result of this error, peculiar situations occurred in the simulation such as full throttle while downhill driving etc.

With the new method of car specific speed profile calculation coming along with position dependent driver action we are now able to use the SiL directly for load collectives simulation without the need for extra software. Only the topology of a track is needed and track specific restrictions, such as speed limits, obligatory stops etc.).

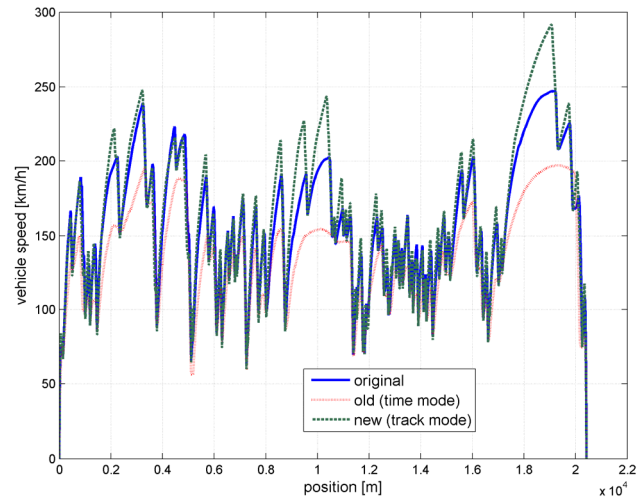


Figure 8: Comparison of simulation results (speed over vehicle position)

For illustration of the realized progress Figure 8 shows a comparison of speed profiles. The first curve (dep. "original") is produced by our special program for load collectives calculation, which used to be the input for SiL simulations. This speed profile is considered a benchmark for the new method. Two SiL simulations were made, one with the previously used method (dep. "old (time mode)") and one using the new approach (dep. "new (track mode)"), each with a similarly configured (engine power, mass) car model on the same drive track.

With the new method, the resulting vehicle speed fits the benchmark speed significantly better than with the old method despite only road track data but no direct information of the benchmark speed was processed. This also applies to the primarily relevant criterion for drivetrain endurance i.e. the load collective. In Figure 9 - Figure 11 one can see the compari-

son of the three variants w.r.t. the load shapes at the cardan shaft (torque over engine speed).

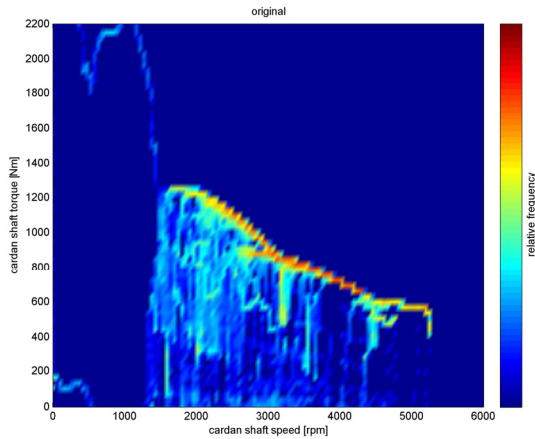


Figure 9: Simulated cardan shaft load collectives (torque over speed), original version.

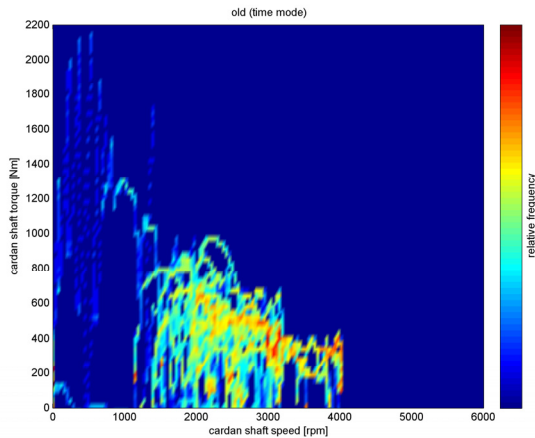


Figure 10: Simulated cardan shaft load collectives (torque over speed), old version using time mode.

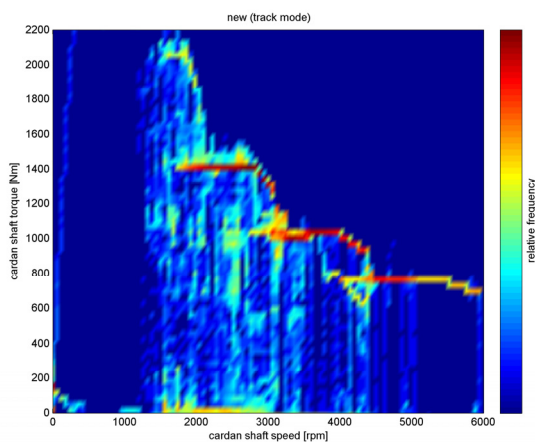


Figure 11: Simulated cardan shaft load collectives (torque over speed), new version using track mode.

It is obvious that the distribution yielded with the new method matches better the benchmark than the old approach. Analogous results are obtained for all drive tracks that Daimler uses in the gearbox development. The flexible car dependent parameterization obviously yields better robustness of the load collectives results.

With the new method for the calculation of time independent, vehicle specific speed profiles we made an important step towards the evaluation of load collectives in conjunction with SiL simulation. Our input to the simulation from now on consists in time independent track data and is identical for all vehicle models, regardless of car weight or installed engine power.

6 Conclusions

A new method for quick automatic calculation of reference speed profiles which are applicable for automatic gearbox testing was developed at DLR and implemented using Modelica. The resulting speed profiles are specific for the assumed vehicle data. Moreover, they can easily be adapted by tuning of various parameters which are interpretable to represent different driver behavior.

At Daimler, the algorithm is now used for both flexible and reproducible generation of load collectives for virtual drivetrain endurance testing. The new approach replaces the less efficient procedure where static time-dependent speed profiles were taken as inputs which had been produced from special load collective generation software or driving experiments separately for each car type.

7 Acknowledgement

The presented results were compiled in the context of the ITEA2 project Modelisar [4].

References

- [1] Aguilera, V., Glaser, S., Arnim, A.: An advanced driver speed assistance in curves: risk function, cooperation modes, system architecture and experimental validation. Proc. IEEE Intelligent Vehicles Symposium. Las Vegas, Nevada, 2005,
- [2] Jimenez, F., Aparicio, F., Paez, J.: Evaluation of in-vehicle dynamic speed assistance in Spain: algorithm and driver behaviour. IET

Intelligent Transport Systems, Vol. 2, No. 2, 2008.

- [3] Thümmel M. et. al. Nonlinear Inverse Models for Control. Proc. 4th Int. Modelica Conf., Hamburg, 2005.
- [4] “*MODELISAR Project Profile*” 2008, http://www.itea2.org/public/project_leaflets/MODELISAR_profile_oct-08.pdf
- [5] Schlabe, D., Knostmann, T., Bünte, T.: Scade Suite Modelica Interface. Proc. 8th Int. Modelica Conf., Dresden, Germany, 2011.
- [6] Chrisofakis, E., Junghanns, A., Kehrer, C., Rink, A.: Simulation-based development of automotive control software with Modelica. Proc. 8th Int. Modelica Conf., Dresden, Germany, 2011.