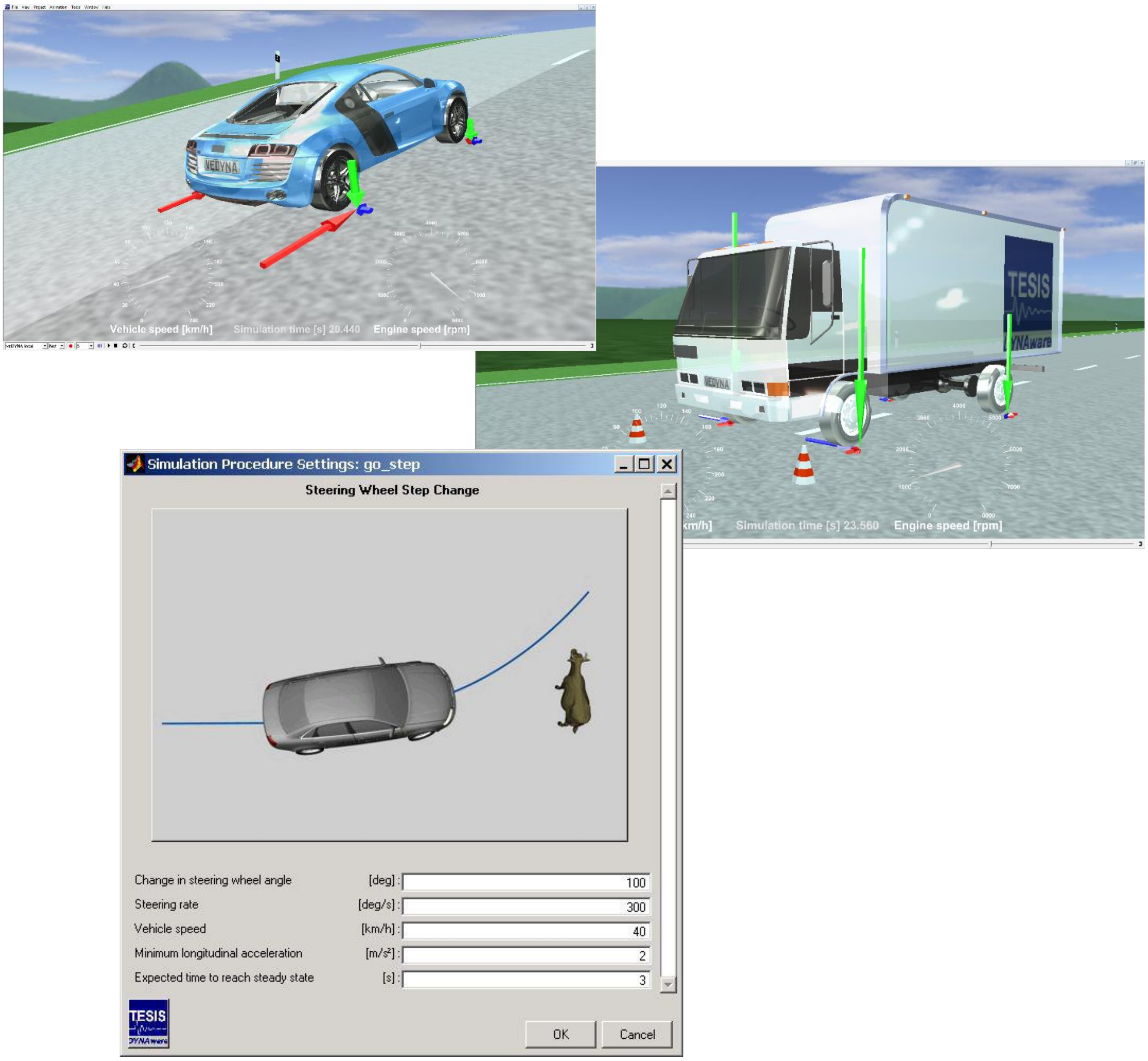
Pushing Innovation

veDYNA Entry

Example Book for Preconfigured Maneuvers



Example Book

Modification: August 22, 2014

For Software Version: General ve-DYNA® Documentation

TESIS DYNAware
Technische Simulation Dynamischer Systeme GmbH

Baierbrunner Straße 15

D-81379 Munich

Germany

Telephone:	+49 89 74 73 77-0
Telefax:	+49 89 74 73 77-99
E-mail:	tesis.dynaware@tesis.de
Web:	www.tesis-dynaware.com

Contents

1 Introduction

veDYNA Entry incorporates the compiled version of the veDYNA vehicle model, comprising all relevant parts of the vehicle:

- Vehicle body including aerodynamics
- Tabled axle kinematics for front and rear axle
- Tyre model TM-Easy
- Generic Drivetrain model for front wheel drive, rear wheel drive and all wheel drive
- Manual and automatic Transmission
- Brake on every wheel

This model approach is proven for a broad range of vehicle dynamics simulation tasks.

veDYNA Entry comes with various ready-to-go procedures for the simulation of standard tests, such as steer step, double lane change, and many more.

These procedures are coded in the MATLAB programming language and are based on pre-configured maneuver and road definitions.

The entire MATLAB functionality may be used to specify preprocessing, execution and postprocessing. These so-called User Procedures are therefore a powerful means to control the execution of the simulation. Specifically designed Graphical User Interfaces allow comfortable variations of test parameters. As the User Procedures are delivered as open Matlab code, user specific code may be added in all supplied User Procedures. Please refer to the appropriate manuals for MATLAB programming instructions when you develop your own procedures, especially for graphical functions.

This document contains detailed descriptions of the provided User procedures.

By extending the example procedures and adopting the main ideas shown in this manual, you will be able to develop individual applications to meet your particular simulation requirements.

2 General Handling of Simulation Procedures

Generally, all actions that can be performed via the veDYNA GUI (e.g. applying simulation parameters, running the simulation) are also available via specific functions of the MATLAB interface, which can be used in the Simulation Procedures. In addition, it is possible to set parameters and do all kinds of calculations for pre-processing and post-processing.

A simulation procedure comprises the following steps:

1. Definition of a user interface for entry of simulation parameters
2. Launch of the dialogue
3. Retrieving the entered data from the user interface
4. Preprocessing: Preliminary calculations, setting of simulation parameters (vehicle, road, maneuver)
5. Specification and update of driving maneuver, road and trace variables
6. Execution of the simulation
7. Postprocessing: Evaluation or of simulation results
8. Plotting of simulation results

To execute an existing procedure, select a file in the **User Procedure** area in the veDYNA GUI and click the button next to the pull-down menu.

For most of the supplied example procedures a user data mask pops up, where some simulation parameters can be entered. If data entry is complete, click the OK button in the user data mask. If no user interface is available, the procedure is executed immediately. The default parameters set within the procedure are used.

The procedures are available as a MATLAB m-file and can be edited using the button. The appearance of the data entry mask and the parameters to be entered is data driven and can therefore also be defined within the procedure.

The easiest way to generate a new simulation procedure is to modify one of the provided examples. Alternatively, individual simulation procedures can be created from the currently selected settings for maneuver, road and trace by means of the menu function **Simulation | Generate User Procedure with Current Simulation Settings...**.

The veDYNA functions `vm_apply` and `vlm_apply` are used to select and update the model data of the maneuvers, driver, road, and vehicle data, respectively. The effect is equivalent to the button in the veDYNA GUI. The functions can be used in Simulation Procedures or from the MATLAB Command Window.

If no vehicle data is specified in the procedure, the currently active vehicle configuration is used (from the last parameter apply). If the procedure applies only parts of the vehicle data, veDYNA will read the remaining data sets of the previous data apply.

For a comprehensive description of this function please refer to the veDYNA User Manual or type `help vm_apply` in the MATLAB command window during your veDYNA session.

The function `vm_load` is used to read data files from the database. `vm_load` returns the content of the data file in a local data structure. `vm_load` and `vm_apply` provide a mechanism to manipulate model data within a simulation procedure:

1. Read the data file (`vm_load`)
2. Modify the local data structure
3. Put the modified data into operation (`vm_apply`)

To define the tracing, use the function `up_trace(start_time, stop_time, 'trace_file')`.

The simulation is started by means of the function `vm_runsim(0.0, stop_time)`.

The function `res = vm_results` is used to access the current simulation output (trace variables). The function returns a structure containing the result data. The traced variables then are called with the structure prefix, e.g. `res.tsim`. Apart from the prefix, the variable names are the same as those in the trace file. The MATLAB interface is completely listed in the veDYNA User Manual.

3 Example Simulation Procedures for Standard Tests

In this chapter selected examples of simulation procedures for standard applications are presented. The following procedures are collected in the Simulation Project **tutorial**:

- go_AccelerationTest: Acceleration and Braking
- go_AxleKinematics: Analysis of Axle Kinematics
- go_Braking_in_a_Curve: Braking in a Curve
- go_Braking_on_MuSplit: Braking on a μ-Split surface
- go_Static_Equilibrium: Reaching the Static Equilibrium
- go_SteadyState_Circular_Drive: Self-Steering Behaviour
- go_Steerstep: Steering Wheel Angle Step
- go_Slalom: Slalom Test
- go_ISO_DoubleLanechange: ISO Double Lane Change
- go_MonteCarlo: Driving the Monte Carlo Race Track
- go_Uphill_Driveaway: Starting from standstill on ascending road with μ-Split

Please note that in all examples the vehicle remains at a standstill for the first second. As the spring stresses do not necessarily correspond to the stresses in the design position of the vehicle, an allowed transient effect will initially start in this phase.

3.1 go_Static_Equilibrium: Maneuver to Reach Static Equilibrium

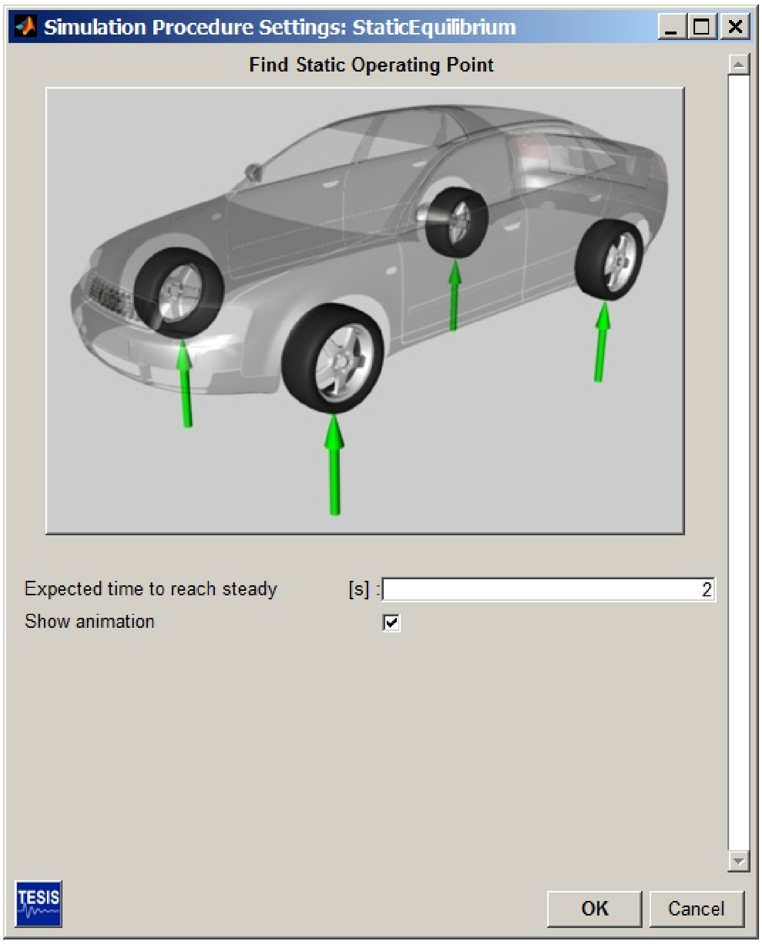


FIGURE 3.1: **Data Entry Mask for Simulation Procedure go_StaticEquilibrium**

The procedure **go_StaticEquilibrium** calculates the transient behaviour of the vehicle reaching static equilibrium from the design position. For this purpose the vehicle is simply kept standing without driver inputs (i.e. with the steering wheel in zero position and without forward movements).

In this example the interesting results are front and rear axle loads, spring lengths, spring forces, roll, pitch and yaw angles of the vehicle and the co-ordinates of the chassis reference point.

This is a test that should be performed on every newly configured vehicle.

> Every simulation should start with the vehicle in the static equilibrium position to avoid transient phenomena. Therefore the first maneuver phase should always be a waiting phase of at least one second.

The following table shows the maneuver definitions used in the **go_StaticEquilibrium** procedure.

TABLE 3.1: Maneuver Definition to Reach Static Equilibrium

Phase	**Time**	**Longitudinal Dynamics**	**Lateral Dynamics**	**Constraints**
Maneuver definition		`Static.m`	`Static.m`	`Static.m`
Find Static Position	2 s	Accelerator Pedal Position (0)	Absolute Steering Input (0)	Neutral gear

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

   TABLE 3.2: Maneuver Parameters of Finding the Static Equilibrium

Specifier	**Description**
Expected time to reach steady state	Time to reach the static equilibrium position from the design position

2. Load and update of the maneuver definition, road and trace files defining the Static Equilibrium test according to the given parameters.

   For this maneuver, a standard road without any features is used.

3. Execution of the simulation

4. Result evaluation: Calculation of axle loads, and display of spring and damper forces, spring deflections, tyre and axle loads the veDYNA Plot GUI, see Figure 3.2.

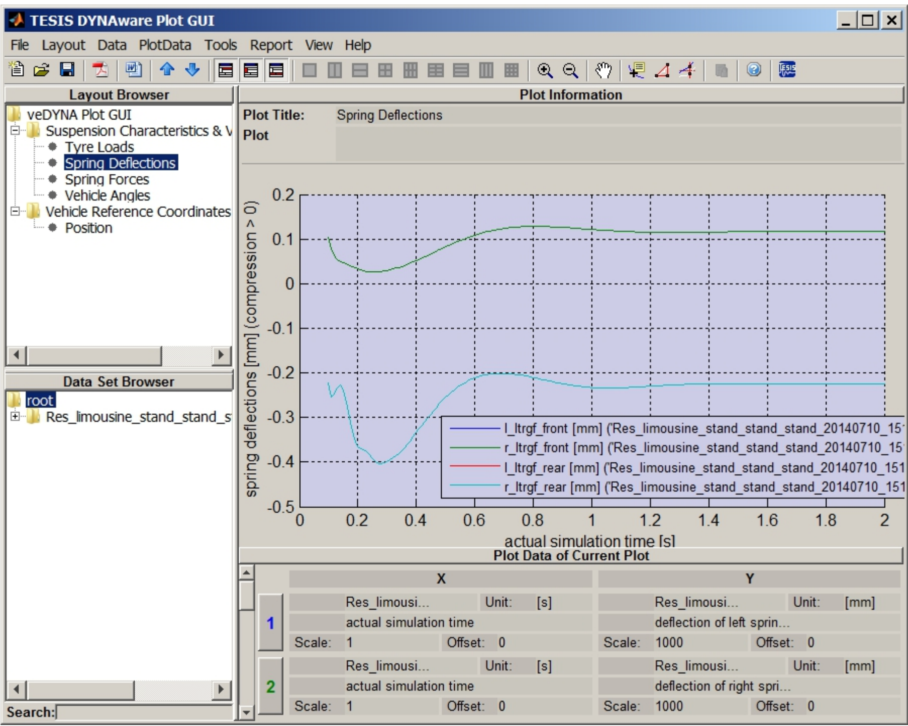


FIGURE 3.2: Example Result Plot: Spring Deflection for Transient Oscillation to Static Equilibrium

3.2 go_AxleKinematics: Analysis of Axle Kinematics

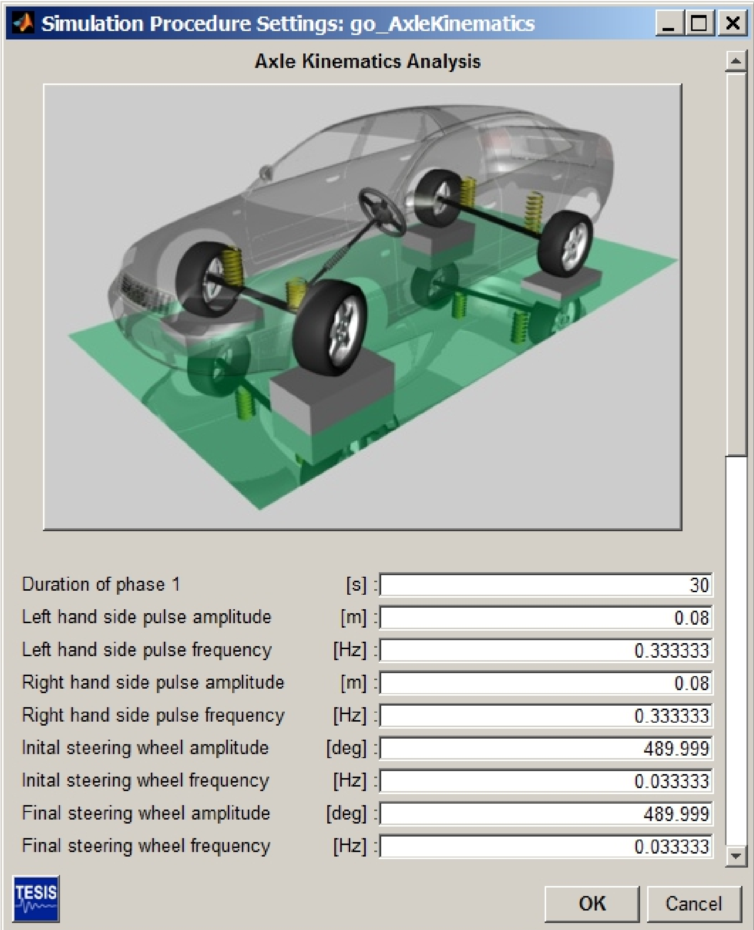


FIGURE 3.3: Data Entry Mask of Simulation Control Procedure go_AxleKinematics

The procedure **go_AxleKinematics** simulates a hydropulse test rig for suspension kinematics by fixing the vehicle body while lifting and steering the wheels. Initially, high frequency sinusoidal wheel lift is combined with low frequency steering movement. Thereafter the wheels are steered at high frequency and at the same time lifted at low frequency. There are no maneuver constraints. With this procedure it is possible to inspect the complete range of axle positions.

Interesting results are the axle kinematics tables for wheel center translation, and caster, camber and toe angles as functions of wheel lift and steer input.

TABLE 3.3: Maneuver Phases of Axle Kinematics

Phase	Time	Longitudinal Dynamics	Lateral Dynamics	Con-straints
Maneuver definition		`Axle_Kinematics.m`	`Axle_Kinematics.m`	`none.m`
High frequency wheel lift, Low frequency steering	30 s	Fixed body, axle kinematics analysis	Sinusoidal wheel input	Neutral gear
Low frequency wheel lift, High frequency steering	30 s	Fixed body, axle kinematics analysis	Sinusoidal wheel input	Neutral gear

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be en-

tered:

TABLE 3.4: Maneuver Parameters of Axle Kinematics

Specifier	**Description**
Duration of Phase 1	Phase length of the first axle excitation mode
Left hand side pulse amplitude	wheel lift amplitude left
left hand side pulse frequency	wheel lift frequency left
Right hand side pulse amplitude	wheel lift amplitude right
Right hand side pulse frequency	wheel lift frequency right
Initial steering wheel amplitude	Steering wheel amplitude at the beginning of the phase
Initial steering wheel frequency	Steering wheel frequency at the beginning of the phase
Final steering wheel amplitude	Steering wheel amplitude at the end of the phase, the amplitude is linearly interpolated in between
Final steering wheel frequency	Steering wheel frequency at the end of the phase, the amplitude is linearly interpolated in between
Duration of Phase 1	Phase length of the first axle excitation mode
Left hand side pulse amplitude	wheel lift amplitude left
Left hand side pulse frequency	wheel lift frequency left
Right hand side pulse amplitude	wheel lift amplitude right
Right hand side pulse frequency	wheel lift frequency right
Initial steering wheel amplitude	Steering wheel amplitude at the beginning of the phase
Initial steering wheel frequency	Steering wheel frequency at the beginning of the phase
Final steering wheel amplitude	Steering wheel amplitude at the end of the phase, the amplitude is linearly interpolated in between
Final steering wheel frequency	Steering wheel frequency at the end of the phase, the amplitude is linearly interpolated in between

2. Load and update of the maneuver definition, road and trace files defining the Axle Kinematics test according to the given parameters.

3. Execution of the simulation

4. Result evaluation:

   The veDYNA variables describing the wheel orientation are transformed into the more common variables toe, camber and caster angles by use of the routine `toe_caster_camber_from_ed`. Wheel position and wheel orientation are arranged in result matrices and written to an ASCII file. This data is further processed in an evaluation routine (`ax_elakin_prepro`) and written to the files 'front_template.m' and 'rear_template.m' in the `~\work .current` directory to be used in veDYNA table kinematics.

5. Plot of wheel orientation angles according to DIN 70000 using the function `vm_plot_din70000_angles` (Matlab figures).

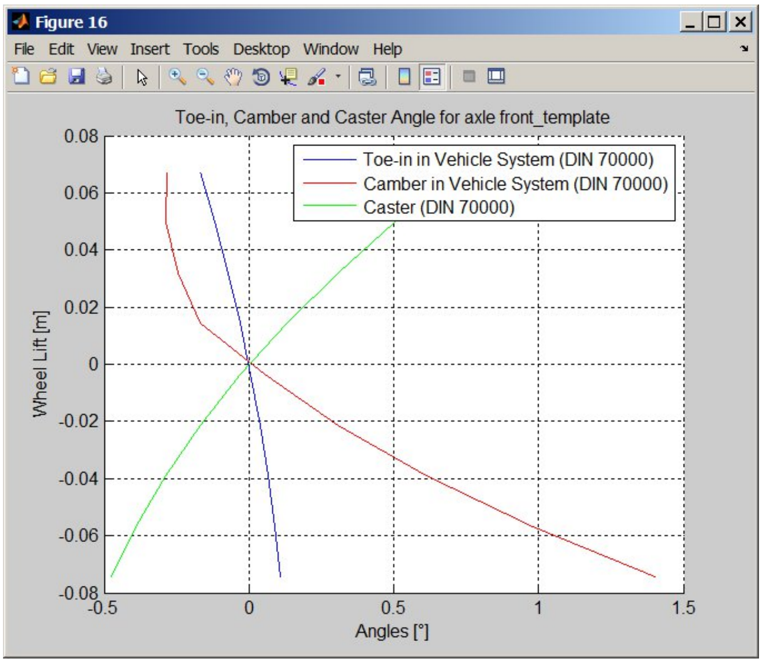


FIGURE 3.4: Example Result Plots: ISO 70000 diagram of caster, camber, and toe angles from Axle Analysis

3.3 go_AccelerationTest: Acceleration and Brake Test

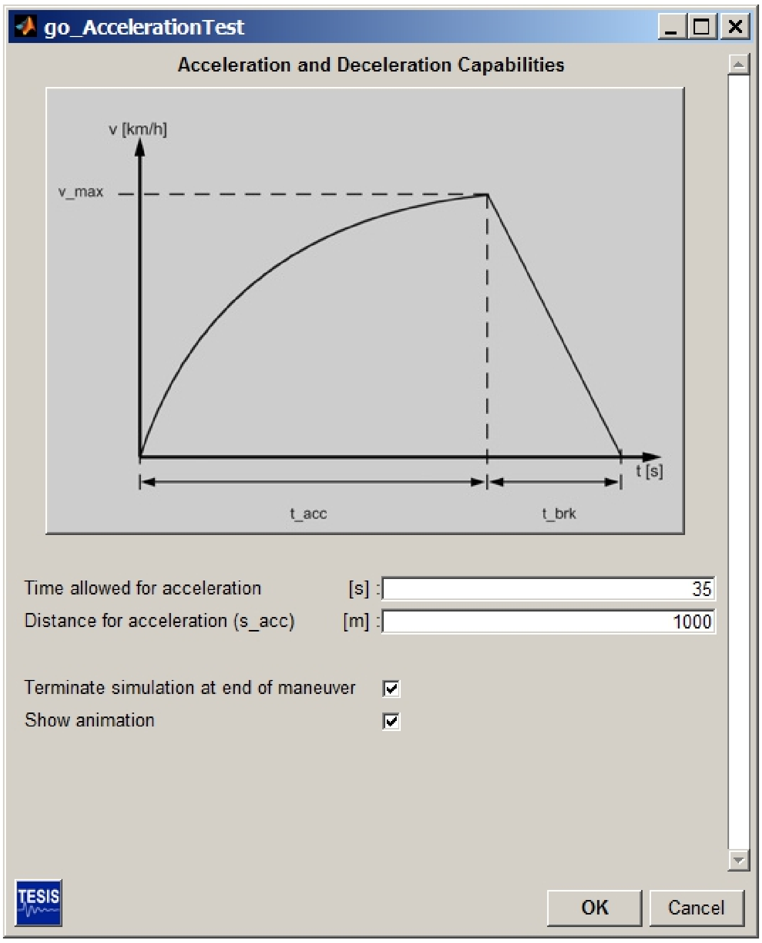


FIGURE 3.5: Data Entry Mask for Simulation Control Procedure go_AccelerationTest

The procedure **go_AccelerationTest** determines acceleration and deceleration capabilities of a vehicle by applying full throttle and full brake over a given time or distance.

Interesting results are wheel loads, spring lengths and forces, pitch, yaw and roll angle, vehicle acceleration, vehicle velocity, as well as rotational speeds and torques in the driveline.

> The specified time and distance for acceleration should be set high enough such that a velocity of at least 100km/h is reached. The braking scenario starts as soon as either the specified time or acceleration distance is reached and ends when zero speed is obtained. The specified brake time should be set high enough to fulfill this. Warnings are otherwise displayed in the Command Window.

This is the third test (after go_StaticEquilibrium and go_AxleKinematics) that should be performed on a newly configured vehicle.

TABLE 3.5: Table of Maneuver Phases go_AccelerationBrakeTest

Phase	Duration	Longitudinal Dynamics	Lateral Dynamics	Constraints
Maneuver definition		AcceleratorBrake Test.m	AcceleratorBrake Test.m	AcceleratorBrakeTest .m
Rev up the engine	1 s	Acc. Pedal Input (90)	Abs. Steering Input (0)	Neutral gear
Acceleration at full throttle	s_acc t_acc	Acc. Pedal Input (HOLD)	Fixed Steering Wheel	Gears 1-7
Operate Brake	0.1s	Acc. Pedal Input (0)	Fixed Steering Wheel	Brake Pedal Input (1)
Deceleration with full brake	t_brk	Acc. Pedal Input (HOLD)	Fixed Steering Wheel	Brake Pedal Input (1)

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.6: Maneuver Parameters of Acceleration - Brake Test

Specifier	Description
Time allowed for acceleration	Time for vehicle acceleration at full throttle
Distance for acceleration	Distance over which vehicle is accelerated at full throttle; the acceleration is terminated as soon as either the acceleration time has elapsed or the distance has been traveled
Time allowed for braking	Time for vehicle deceleration with brake pedal fully pressed down

2. Load and update of the maneuver definition, road and trace files defining the Axle Kinematics test according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   In the postprocessing additional characteristic values are generated, e.g. the time and distance required to reach 100km/h. Additional information, such as maximum speed, acceleration time and distances, is displayed in the Matlab Command Window.

   Results and additional characteristics are displayed in the veDYNA Plot GUI, see Figure 3.6

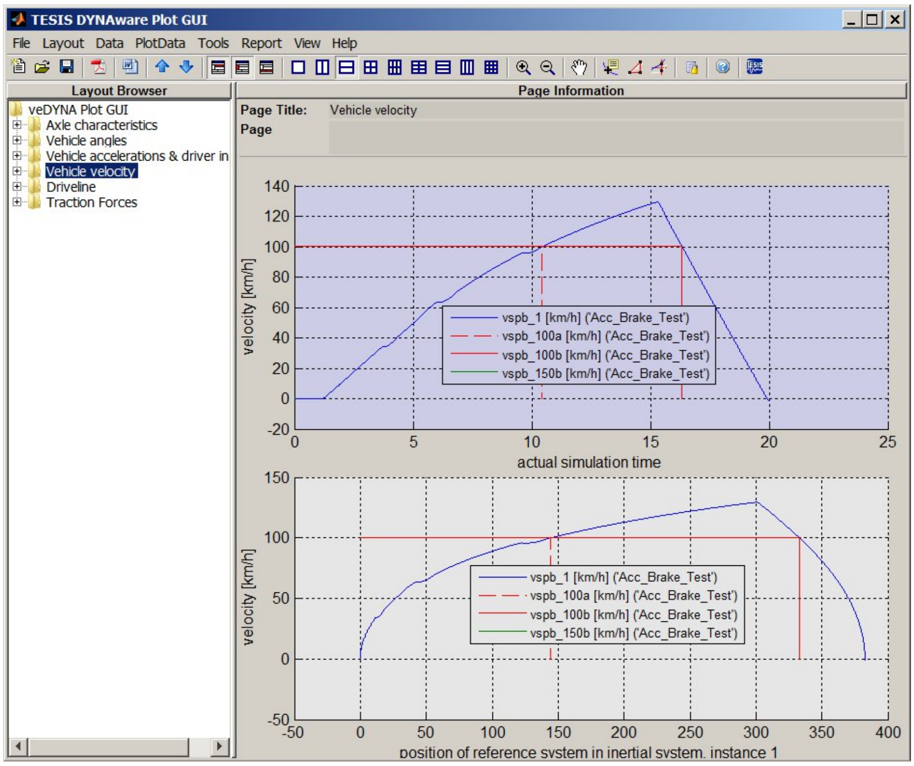


FIGURE 3.6: Example Result Plot: Vehicle Velocity from Acceleration and Brake Test

3.4 go_SteadyState_Circular_Drive: Self-Steering Behaviour

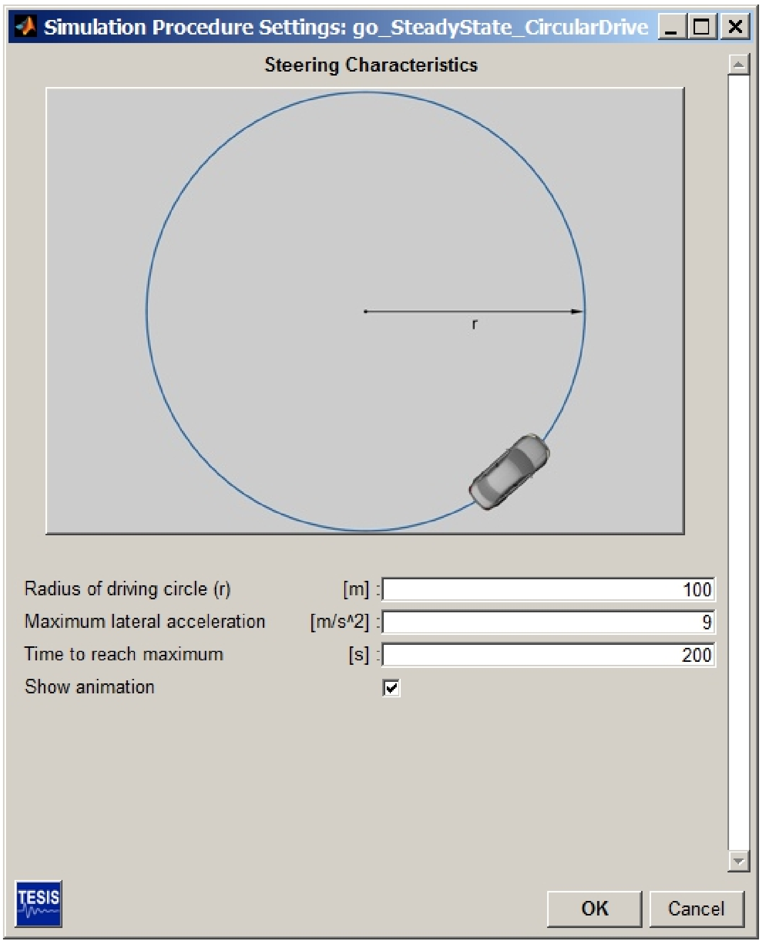


FIGURE 3.7: Data Entry Mask for User Procedure go_SteadyState_Circular_Drive

In the procedure go_SteadyState_CircularDriving the vehicle is constantly accelerated on a circular course of radius r, until the defined lateral acceleration a_y is achieved. This is the standard test to determine self-steering behaviour.

Interesting results to estimate the self-steering behaviour are the steering gradient, yaw velocity amplification, and characteristic speed.

Yaw velocity amplification is defined as the yaw angle over the average steering angle of the front axle. The steering characteristics show the steering wheel angle as a function of lateral speed. The neutral steering in the plot is shown as a line with constant gradient. Velocity amplification above the neutral steering line describes an oversteering tendency and below the line an understeering tendency. The characteristic speed is the maximum speed of an understeering vehicle and shows the speed at which the steering is most sensitive.

i	In order to have approximately steady state conditions, the time allowed for acceleration should be reasonably long. If the maneuver time is too small, only transient states will occur and steady state is not reached.

TABLE 3.7: Description of Maneuver go_SteadState_CircularDrive

Phase	Time	Longitudinal Dynamics	Lateral Dynamics	Constraints
Maneuver definition		`Circle_Drive.m`	`Circle_Drive.m`	`Start1s.m`
Find Static Position	1 s	Cruise Control (0)	Abs. Steering Input (0)	Neutral gear
Acceleration on circular drive with constant radius	`t_sim`	Nominal lateral acceleration control	Controlled steady-state circular run	Gears 1-5

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.8: Maneuver Parameters of Steady State Circular Drive

Specifier	Description
Radius of driving circle	Radius of the nominal circular driving trajectory
Maximum lateral acceleration	The vehicle is accelerated linearly from zero to the maximum velocity given by circle radius and maximum lateral acceleration
Time to reach maximum acceleration	Time allowed for acceleration: should be set high enough to have steady state conditions

2. Load and update of the maneuver definition, road and trace files defining the Steady State Circular Driving test according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   A mean value of the effective steering ratio is calculated for lateral accelerations up to a user-specific limit for linearisation. Steering gradients for the linear model and the longitudinal velocity at which the sideslip angle crosses zero are calculated and displayed in the Command Window. Vehicle states and characteristic values are displayed in the Plot GUI, see Figure 3.12.

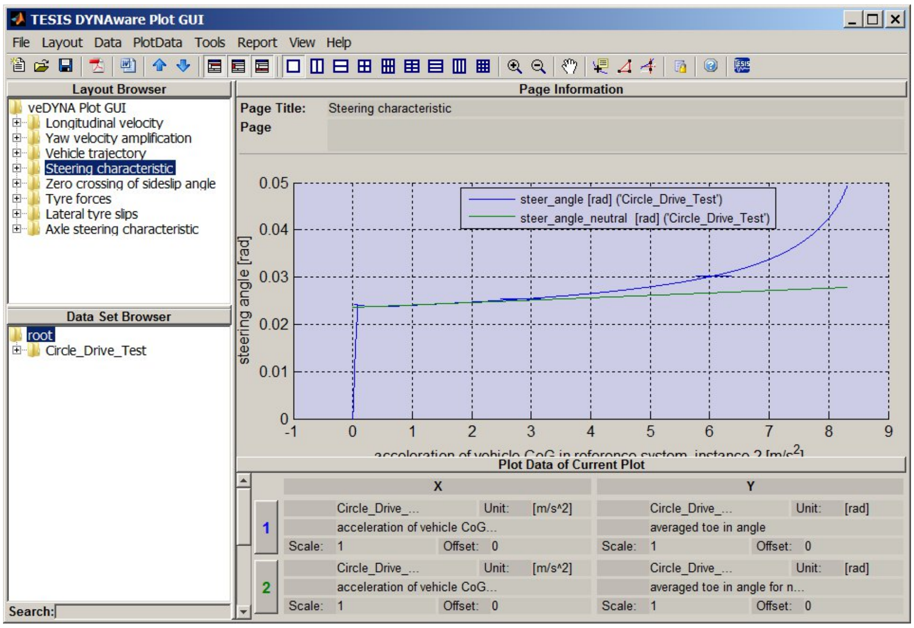


FIGURE 3.8: Example Result Plot: Steering Angle versus Lateral Acceleration for Steady State Circular Drive

3.5 go_SteerStep: Steering Wheel Step

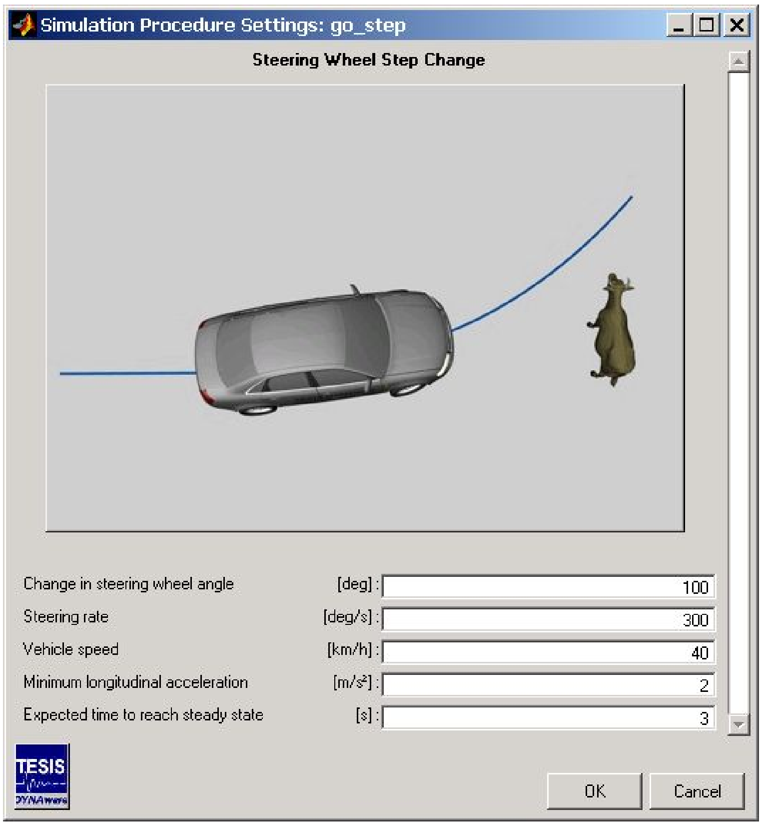


FIGURE 3.9: Data Entry Mask for Simulation Control Procedure go_SteerStep

The steering wheel step is a standard test used for the evaluation of vehicle handling characteristics. The vehicle is accelerated to a given speed, then the steering wheel angle is turned in quickly according to user specification.

Interesting results are the yaw response and the wheel loads.

TABLE 3.9: Maneuver Definition for Steering Step

Phase	**Time**	**Longitudinal Dynamics** `SteerStep.m`	**Lateral Dynamics** `SteerStep.m`	**Constraints** `Start1s.m`
Find Static Position	1 s	Cruise Control (0)	Absolute Steering Input (0)	Neutral gear
Acceleration	t = v/a_x	Cruise Control (v_{target})	Fixed Steering Wheel	Gears 1-5
Stabilisation	t = t_{std}	Cruise Control (v_{target})		
Steering wheel step	t = dt		Absolute steering input (δ_h)	
Stabilisation	t = t_{std}		Fixed Steering Wheel	

> The steering wheel step should produce a lateral acceleration of approximately 4 m/s^2 for the selected speed.

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.10: Maneuver Parameters of the Steering Wheel Step

Specifier	Description
Change in steering wheel angle	Amount of steering wheel turn
Steering Rate	Steering wheel turning speed
Vehicle Speed	Nominal velocity at which the steering wheel step is performed
Minimum longitudinal acceleration	Maximum acceleration during the acceleration phase from standstill to the nominal velocity; used for the maneuver timing

2. Load and update of the maneuver definition, road and trace files defining the Steering Wheel Step according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   For post-processing of the simulation results a local function `res_extract` has been added to the procedure. As only the range from the beginning of the steering angle step to the end of the maneuver is of interest, the initial phases (static equilibrium, acceleration and stabilisation) are skipped.

   Assuming that a steady state is reached at the end of the simulation, the relative (scaled) yaw rate and lateral acceleration are calculated as a percentage of the final values. A characteristic value is the time, at which 90 % of the steady yaw rate is reached. The time of the first maximum yaw rate . The values of the maximum yaw rate in relation to the steady yaw rate and the TB-value are calculated and displayed in the MATLAB command window.

   Finally vehicle states such as the calculated relative yaw rate, lateral acceleration, and tyre loads are plotted in the veDYNA Plot GUI, see Figure 3.10.

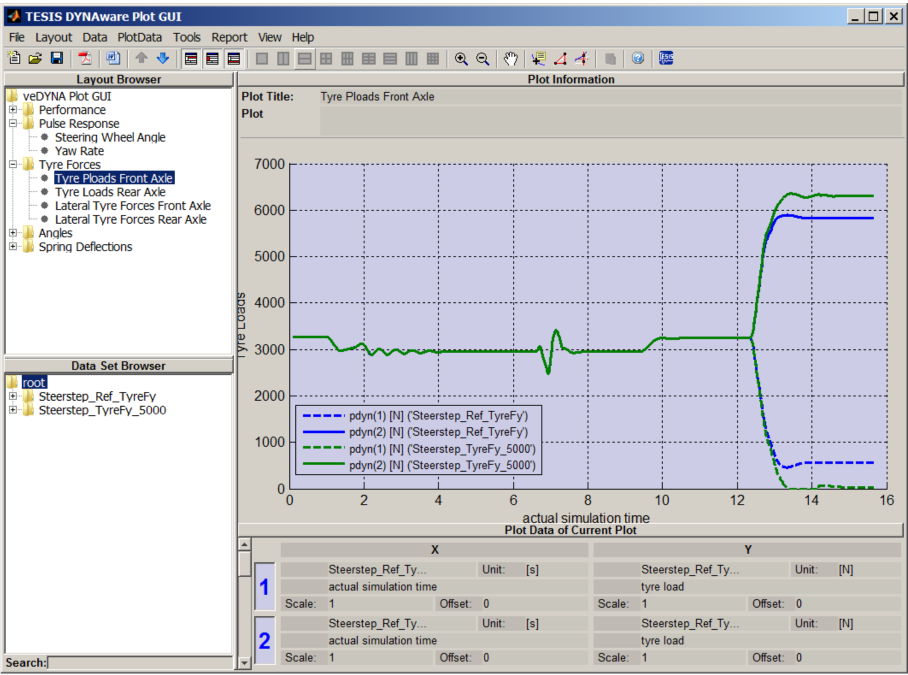


FIGURE 3.10: Example Result Plot: Tyre Loads at the Front Axle for the Steering Wheel Step (two Vehicle Variants)

3.6 go_Slalom: Slalom Test

The procedure **go_Slalom** makes the selected vehicle drive at given speed through a series of cones positioned at a prescribed distance `ds`.

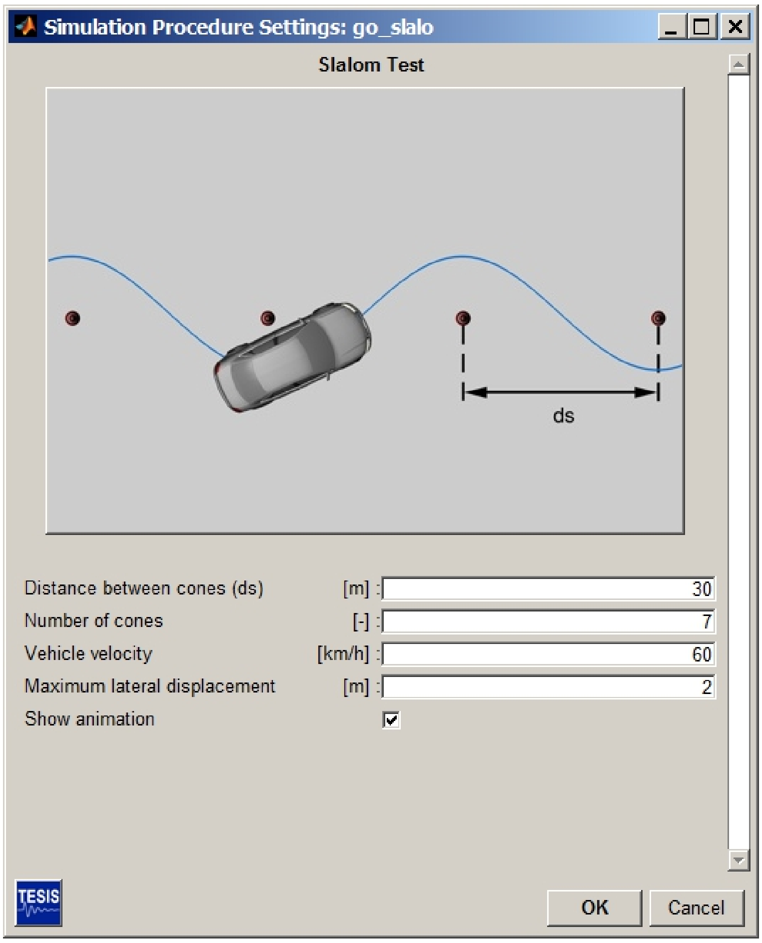


FIGURE 3.11: Data Entry Mask for procedure go_Slalom

The vehicle is accelerated to the required velocity, then the Basic Driver follows a sinusoidal path through the cones at constant speed.

TABLE 3.11: Maneuver Definition of Slalom

Phase	Time	Longitudinal Dynamics	Lateral Dynamics	Constraints
Maneuver definition		`Slalom.m`	`Slalom.m`	`Start1s.m`
Find Static Position	1 s	Basic Driver (speed)	Fixed Steering Wheel (0)	Neutral gear
Acceleration	`s_acc+s_stat`		Basic Driver (QHOLD)	Gear 1-5
Drive Slalom Course	`n*ds`		Basic Driver (QSINE)	
Straight Line Driving	`s_exit`		Basic Driver (QHOLD)	

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.12: Maneuver Parameters of Slalom Drive

Specifier	Description
Distance between cones (ds)	Spacing of cones for slalom drive, indicates the frequency of the sinusoidal curve through the cones
Number of cones	Indicates the length of the slalom drive
Vehicle velocity	Nominal velocity for the slalom drive through the cones
Nominal longitudinal acceleration	Maximum acceleration during the acceleration phase from standstill to the nominal velocity; used for the maneuver timing
Maximum lateral displacement	Maximum lateral amplitude to be driven, indicates the amplitude of the sinusoidal curve through the cones

2. Load and update of the maneuver definition, road and trace files defining the Slalom test according to the given parameters.

3. Execution of the simulation

4. Result evaluation:

   Calculation of cone positions and display of vehicle trajectory, vehicle states such as speed, yaw angle, yaw rate in the veDYNA Plot GUI, see Figure 3.12.

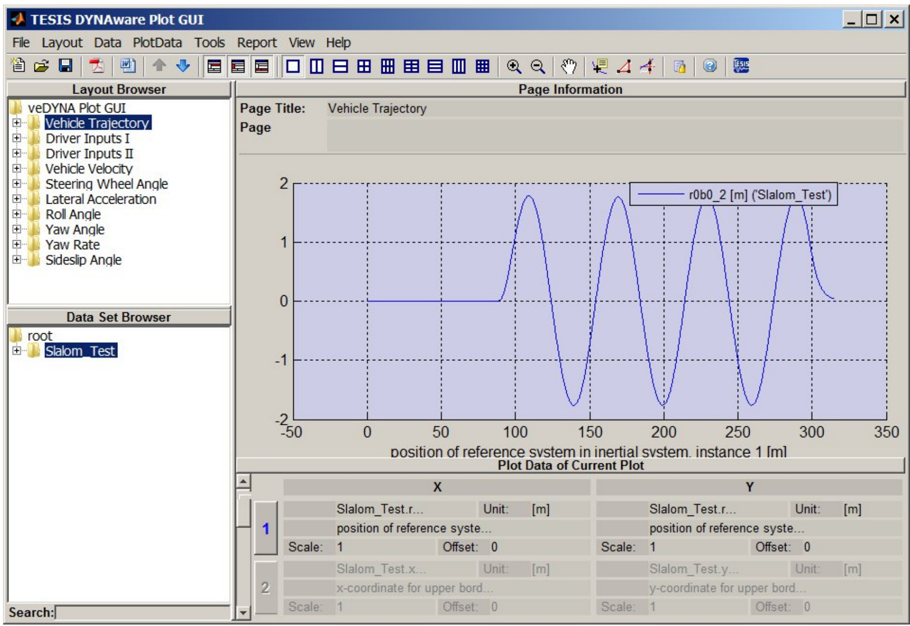


FIGURE 3.12: Example Result Plot: Vehicle Trajectory for the Slalom Test

3.7 go_Braking_in_a_Curve: Braking in a Curve

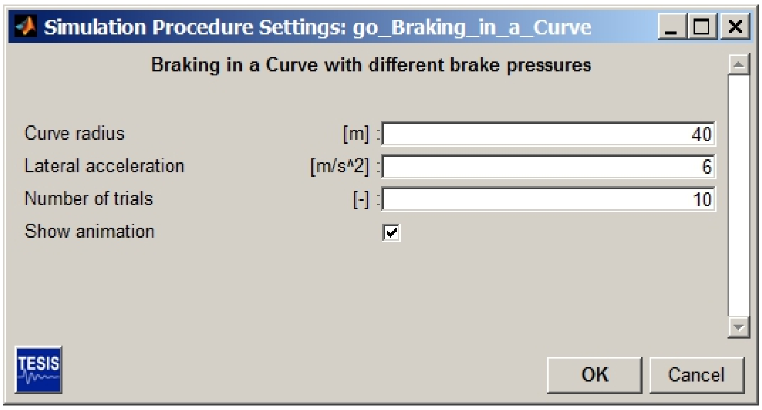


FIGURE 3.13: Settings for Simulation Procedure go_Braking_in_a_Curve

The procedure **go_Braking_in_a_Curve** simulates braking during a steady circular path with different constant braking pressures. The vehicle is accelerated on a circular course of constant radius until the desired lateral acceleration is achieved. Then a specified brake pressure is applied for 1 s to decelerate the vehicle. The maneuver phases can be described as shown in Table 3.13.

TABLE 3.13: Description of Maneuver Braking in a Curve

Phase Maneuver definition	**Time**	**Longitudinal Dynamics** `Brake_in_Curve.m`	**Lateral Dynamics** `Brake_in_Curve.m`	**Constraints** `Brake_in_Curve.m`
Find Static Position	1 s	Accelerator pedal input (0)	Abs. Steering Input (0)	Neutral gear
Acceleration	`t_acc`	Nominal lateral acceleration control (`ay`)	Controlled steady-state circular run (`1/r`)	Gear 1-7
Stabilisation	`t_std`	Nominal lateral acceleration control (Hold)		
Braking	1 s	Accelerator pedal input (0)	Fixed Steering Wheel	Neutral gear, Apply Brake

Interesting results are normalized yaw rate, lateral acceleration and slip angle at the end of braking (w.r.t. their respective steady-state values at the end of the stabilization phase). These values are plotted against longitudinal vehicle deceleration.

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.14: Maneuver Parameters of Braking in a Curve

Specifier	Description
Curve Radius	Radius of the Driven Curve
Lateral Acceleration	Lateral Acceleration to be reached when braking starts
Longitudinal Acceleration	Longitudinal acceleration for the acceleration to final velocity
Number of trials	Number of simulations: The braking is performed for different brake pedal positions, logarithmically spaced between 0 and 1

2. Load and update of the maneuver definition, road and trace files defining the Brake Maneuver according to the given parameters.
3. Execution of the simulation One simulation run is required for the calculation of braking with different brake pressures. For each run loading of the maneuver files and the execution of the simulation need to be repeated. The brake pressure is increased at each simulation run until full brake pressure is attained.
4. Result evaluation:

   Yaw rate, lateral acceleration and slip angle at the end of braking are normalized w.r.t. their respective steady-state values at the end of the stabilization phase. These values are plotted against longitudinal vehicle deceleration (a diagram showing the results of all simulation run is generated as a Matlab figure), see Figure 3.14.

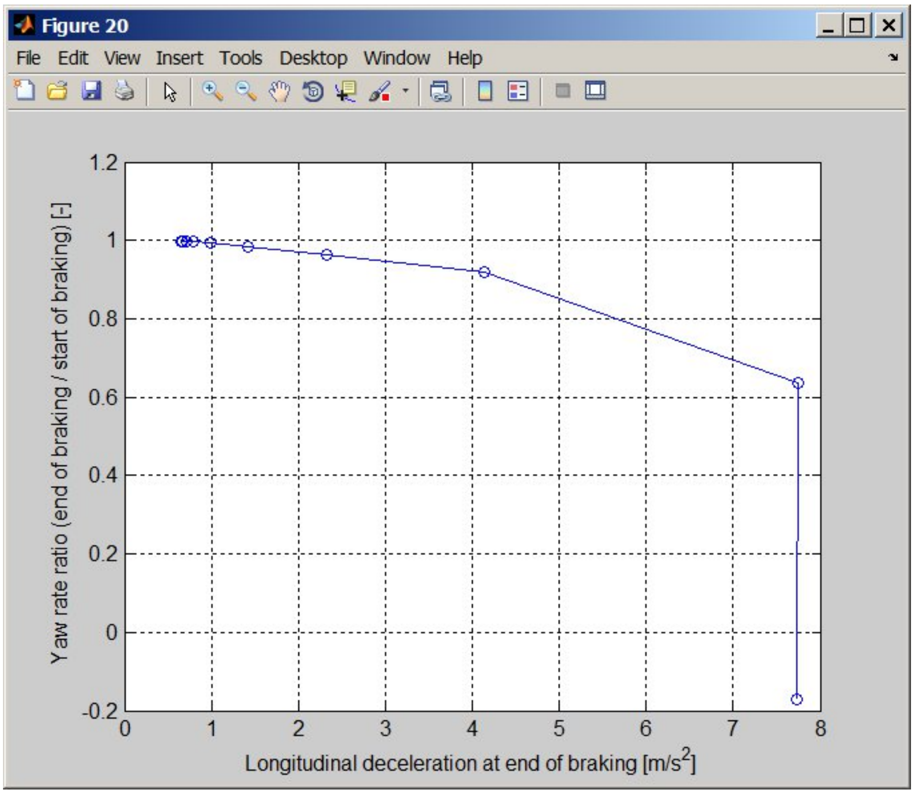


FIGURE 3.14: Example Result Plot: Yaw Rate Ratio as a Function of Brake Deceleration

3.8 go_Braking_on_MuSplit: Braking on Mu-Split

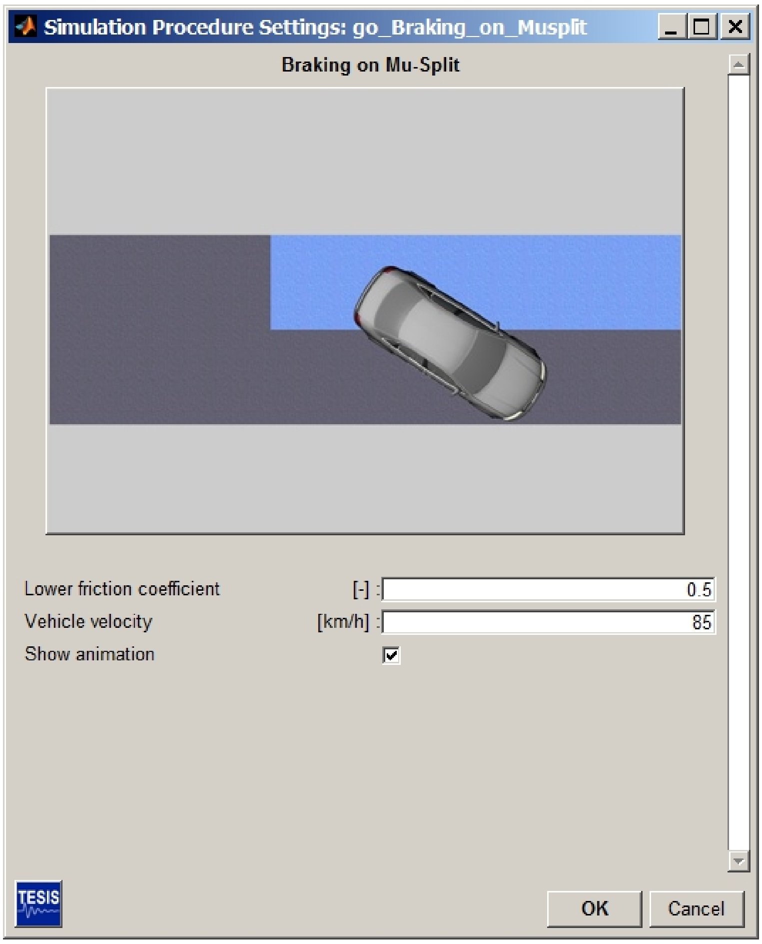


FIGURE 3.15: Settings for Simulation Procedure go_Braking_on_MuSplit

The procedure **go_Braking_on_Mu_Split** simulates full braking on a road with different friction coefficients on the right and left hand side of the road.

The vehicle is accelerated up to the desired speed and is then decelerated with fully pressed brake pedal until standstill. This maneuver is a good test for comparing the driving behaviour of vehicles with and without ABS. The maneuver can be described by the maneuver phases in Table 3.15.

TABLE 3.15: Table of Maneuver Phases go_Braking_on_Mu-Split

Phase	Time	Longitudinal Dynamics	Lateral Dynamics	Constraints
Maneuver definition		`Brake_on_MuSplit.m`	`Brake_on_MuSplit.m`	`Brake_on_MuSplit.m`
Find Static Position	1 s	Acc. Pedal Input (0)	Abs. Steering Input (0)	Neutral gear
Accelerate	`t_acc`	Cruise Control (`speed`)	Controlled Straight Line Driving	Gear 1-7
Stabilisation	`t_std`	Cruise Control (`speed`)	Fixed Steering Wheel	
Full brake application	`t_brake`	Acc. Pedal Input (0)		Full brake

Interesting results are yaw angle and yaw rate, acceleration and deceleration, distance to standstill.

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.16: Maneuver Parameters of Braking on Mu-Split

Specifier	Description
Lower friction coefficient	Friction coefficient of the left half of the proving ground, the friction of the right half is set to 1
Vehicle velocity	Nominal velocity at which the brake is applied
Nominal longitudinal acceleration (traction)	Nominal acceleration during the acceleration phase from standstill to the nominal velocity; used for the maneuver timing

2. Load and update of the maneuver definition, road and trace files defining the Brake test according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   The vehicle trajectory, yaw angle, yaw rate as well as vehicle speed, vehicle acceleration and deceleration are plotted in the Plot GUI, see Figure 3.16.

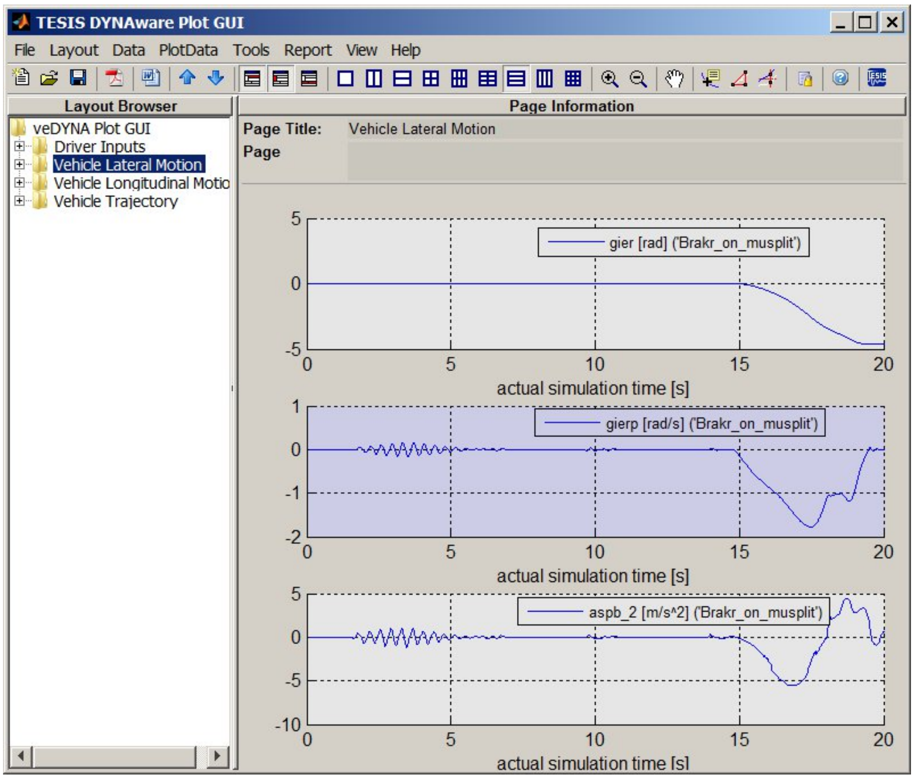


FIGURE 3.16: Example Result Plot: Lateral Vehicle Motion during Braking on a mu Split Surface

3.9 go_ISO_DoubleLaneChange: ISO Double Lane Change

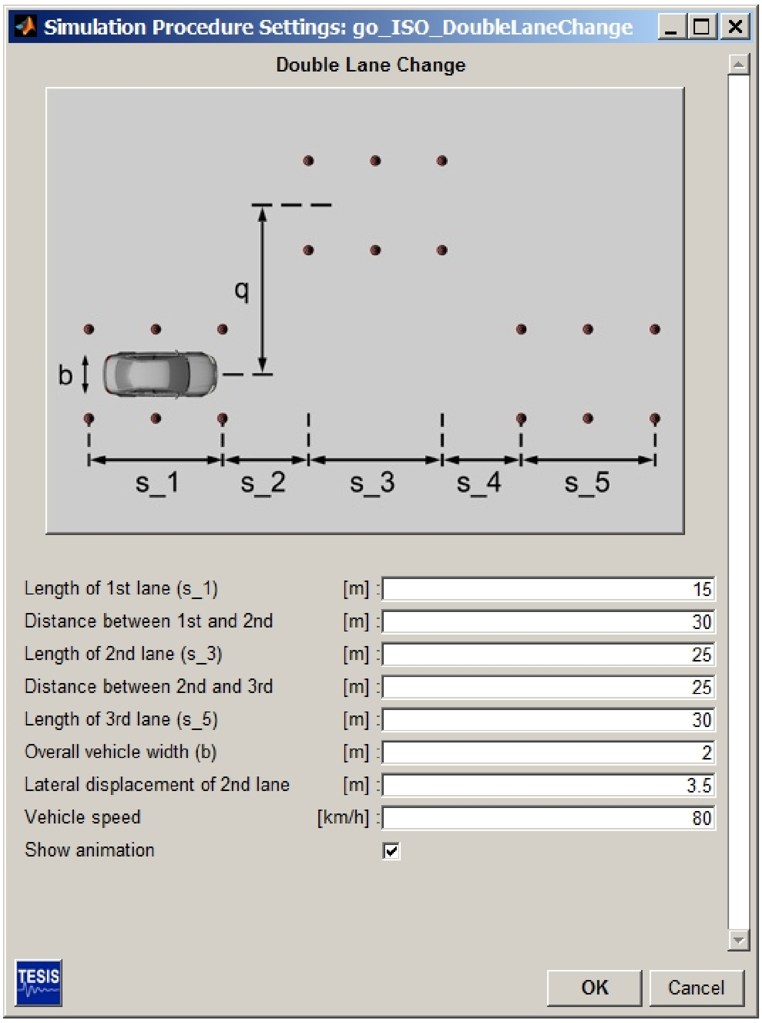


FIGURE 3.17: **Data Entry Mask of Simulation Control Procedure go_ISO_DoubleLaneChange**

This procedure implements the ISO double lane change using the Basic Driver. The target course is specified by means of lateral displacements defined in the Basic Driver submaneuvers. For the animation cones are positioned and displayed automatically.

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.17: Maneuver Parameters of ISO Double Lanechange

Specifier	Description
Length of 1st lane (s_1)	Length of first lane
Distance between 1st and 2nd lane (s_2)	Spacing between lane 1 and lane 2
Length of 2nd lane (s_3)	Length of second lane
Distance between 2nd and 3rd lane (s_4)	Spacing between lane 2 and lane 3
Length of 3rd lane (s_5)	Length of third lane
Overall vehicle width	Vehicle width, used to set the lane width
Vehicle speed	Vehicle speed for driving through the lanes

2. Load and update of the maneuver definition, road and trace files defining the ISO Double Lane Change according to the given parameters.

3. Execution of the simulation

4. Result evaluation:

   Successful maneuver completion is checked, and vehicle states, such as vehicle trajectory, steering wheel angle, lateral acceleration,roll angle, etc. are plotted in the veDYNA Plot GUI, see Figure 3.18.

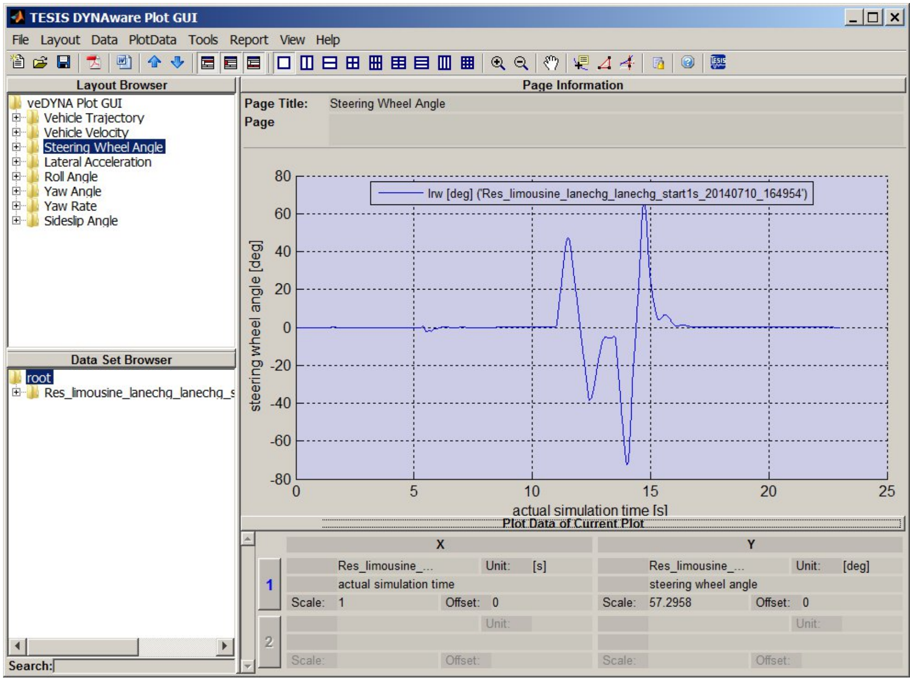


FIGURE 3.18: Example Result Plot: Steering Wheel Angle from an ISO Double Lanechange)

3.10 go_Uphill_Driveaway: Starting on a Hill with mu-Split

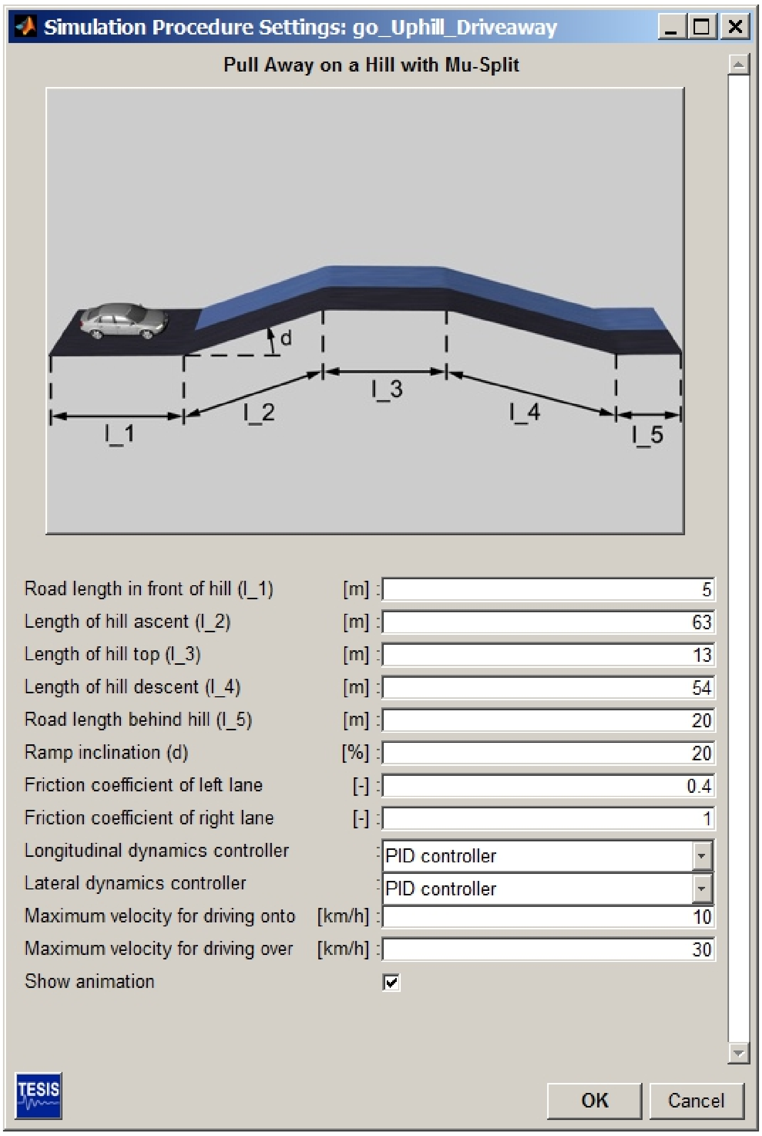


FIGURE 3.19: Data Entry Mask for Simulation Control Procedure go_Uphill_Driveaway

The procedure **go_Uphill_Driveaway** simulates pulling away at a hill ramp from standstill. For this purpose the vehicle is first driven to the ramp and stopped there, before the actual test begins. Different road friction coefficients can be prescribed for the wheels at both sides of the two lane road. This maneuver is useful for analyzing traction differences between various driveline configurations (i.e. rear wheel drive and four wheel drive) and also differential locking, see Table 3.18.

TABLE 3.18: Maneuver Description go_Uphill_Driveaway

Phase	Time	Longitudinal Dynamics	Lateral Dynamics	Constraints
Maneuver definition		`Uphill_Driveaway.m`	`Uphill_Driveaway.m`	`Uphill_Driveaway.m`
Static position	1 s	Cruise control (0)	Abs. Steering Input (0)	Neutral gear
Drive to ramp	t_acc1	Cruise Control (vmax1)	Fixed Steering Wheel	Gear 1-2
Stop on Ramp	t_acc1	Cruise Control (0.0)	Fixed Steering Wheel	Gear 1-2
Wait for static position	t_wait	Cruise Control (0.0)		Neutral gear
Start on ramp	t_acc2	Basic driver (automatic speed)	Basic driver (QHOLD)	Gear 1-2

> If the speed to drive over the ramp is set too high, the front axle may break while driving downhill.

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.19: Maneuver Parameters of Uphill Driveway

Specifier	Description
Road length in front of hill	Distance from initial vehicle position to beginning of hill slope
Length of hill ascent	Length of uphill slope section
Length of hill top	Length of flat hill top
Length of hill decent	Length of downhill slope section
Road length behind hill	length of flat road part behind hill
Ramp inclination	Hill slope in %
Friction coefficient of left lane	
Friction coefficient of left lane	
Longitudinal dynamics controller	Type of driver controller to control given speed
Lateral dynamics controller	Type of driver controller to control course control
Maximum velocity to drive onto ramp	Maximum velocity to drive onto the ramp: there the vehicle stops and initiates the uphill driveaway
Maximum velocity to drive onto ramp	Maximum velocity to drive over the ramp
Expected vehicle acceleration	Used for the maneuver timing
Length of smooth segment	Used for smoothing the edges of the hill geometry

2. Load and update of the maneuver definition, road and trace files defining the Uphill Driveaway according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   Vehicle states, such as tyre loads, tyre forces (traction and lateral), tyre slips, the vehicle trajectory etc. are plotted in the veDYNA Plot GUI, see Figure 3.20.

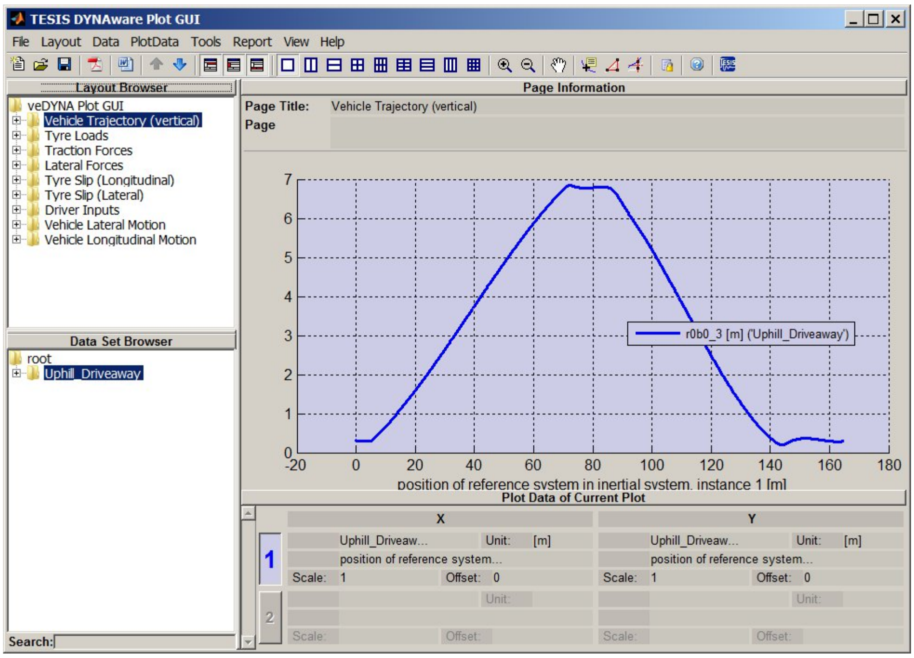


FIGURE 3.20: Example Result Plot: Vertical Vehicle Trajectory for Uphill Driveaway)

3.11 go_MonteCarlo: Driving along the MonteCarlo Racetrack

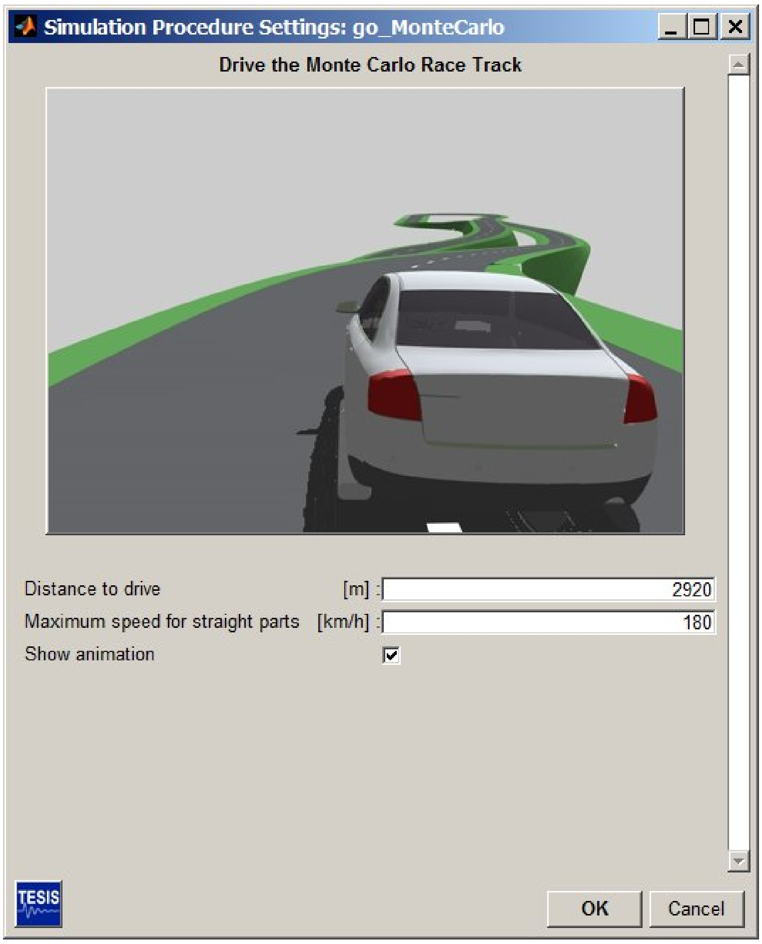


FIGURE 3.21: Data Entry Mask for Simulation Control Procedure go_MonteCarlo

The procedure **go_MonteCarlo** simulates driving along the Monte Carlo race track. The driver model guides the vehicle along the road centerline. For the speed setting the road geometry is considered (i.e. the vehicle brakes in curves). It is possible to drive only parts of the complete track, the simulation terminates after the given driving distance, see Table 3.20.

> The road is defined as a closed circuit, therefore it is necessary to provide an estimate for the time to drive. Otherwise the vehicle travels on the track for the given simulation time.
> It is possible to generate other road geometries from measurements using the GPS Data Converter.

TABLE 3.20: Description of Maneuver Drive MonteCarlo

Phase Maneuver definition	**Time**	**Longitudinal Dynamics** `Driver.m`	**Lateral Dynamics** `Driver.m`	**Constraints** `Start1s.m`
Find Static Position	1 s	Accelerator pedal input (0)	Abs. Steering Input (0)	Neutral gear
Driver Control	`time`	Basic Driver (Automatic Speed)	Basic Driver	Gear 1-7

The procedure works as follows:

1. Display of data entry mask for defining the simulation parameters. The following parameters can be entered:

TABLE 3.21: Maneuver Parameters of MonteCarlo Drive

Specifier	**Description**
Time to drive	The time to drive on the track - the simulation terminated thereafter.
Maximum speed for straight parts	Driver controller setting for maximum allowable speed. The speed in curves is computed automatically from maximum lateral acceleration (this is a driver controller parameter)

2. Load and update of the maneuver definition, road and trace files defining the Monte Carlo drive according to the given parameters.
3. Execution of the simulation
4. Result evaluation:

   Vehicle states, such as vehicle speed and acceleration, the vehicle trajectory compared to the target path, etc. are plotted in the veDYNA Plot GUI, see Figure 3.22.

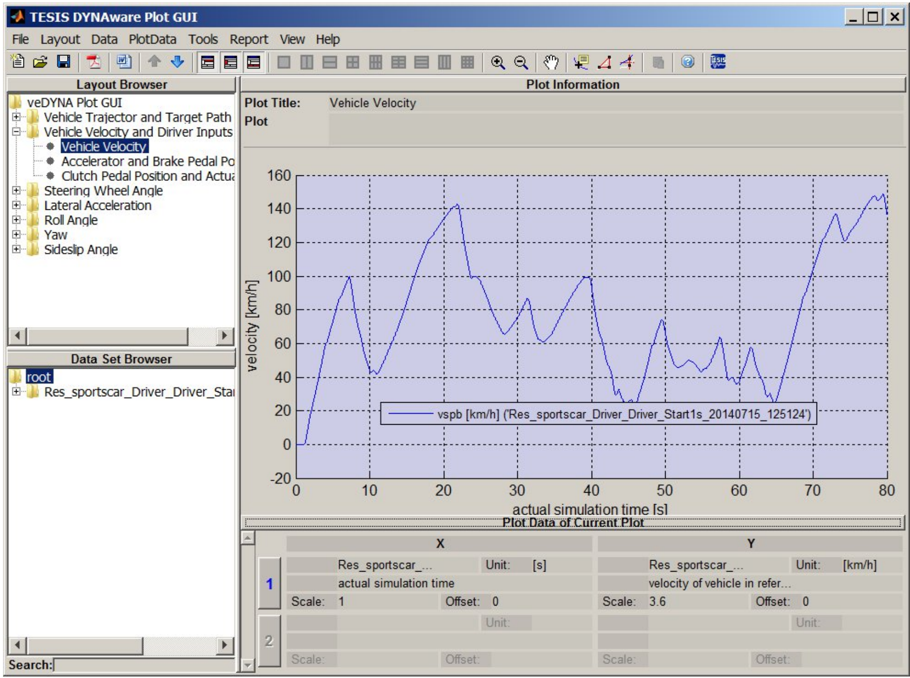


FIGURE 3.22: Example Result Plot: Speed Profile for Monte Carlo Drive)

Imprint

TESIS DYNAware
Technische Simulation Dynamischer Systeme GmbH

Documentation

Baierbrunner Straße 15

D-81379 Munich

Germany

Telephone: +49 89 74 73 77-0

Telefax: +49 89 74 73 77-99

E-mail: tesis.dynaware@tesis.de

Web: www.tesis-dynaware.com