Mechanical Simulation

755 Phoenix Drive, Ann Arbor MI, 48108, USA
Phone: 734 668-2930 • Fax: 734 668-2877 • Email: info@carsim.com



# Tire Models



Several tire models are available in VS Solver programs for the calculation of tire forces and moments. All these models are designed to produce the tire vertical force, the shear forces at the ground, and moments due to tire carcass deflection.

This document describes the screens in the VS Browser for specifying tire properties. It also provides mathematical details of the modeling assumptions and calculation methods used in VS Math Models.

The tire models are based on interactions defined at the ground; however, the multibody model equations in VS Math Models transform the forces and moments and apply them at the wheel center. Import variables are defined for both sets of forces (those applied at the center of tire contact, and those expressed at the wheel center), so advanced users who wish to replace or extend the VS tire models can work with either contact-center forces or wheel-center forces.

Advanced users can extend the definitions of some of the variables in the tire model at run time by using VS commands.

# Tire Property Screens

This section describes the screens used to define the parameters and links that characterize a tire. Three screens are described in detail:

1. **Tire** (the main tire screen),
2. **Tire: Vertical Load vs. Deflection**,
3. **Tires: 2 Axles with Support for Duals** (and three other TruckSim screens with tire groups)

The other library screens are used for specifying tire properties with tables that have standard editing and viewing controls as described in the *VehicleSim Browser Reference Manual***.** Table 1 lists these screens and identifies the keyword root names for the table. The independent variables for these tables are vertical load ($F_Z$) and either lateral slip angle $\alpha$ (alpha), longitudinal slip ratio $\kappa$ (kappa), or inclination angle $\gamma$ (gamma).

*Table 1. Summary of tire table libraries.*

| Library Screen | Root Keyword | Description |
|---|---|---|
| Tire: Aligning Moment | MZ_TIRE | Aligning moment vs. alpha and $F_Z$ |
| Tire: Aligning Moment vs. Inclination | MZG_TIRE | Aligning moment vs. gamma and $F_Z$ |
| Tire: Camber Thrust Coefficient | KGAMMA | Camber thrust coefficient vs. Fz |
| Tire: Lateral Force | FY_TIRE | Lateral force vs. alpha and $F_Z$ |
| Tire: Lateral Force vs. Inclination | FYG_TIRE | Lateral force vs. gamma and $F_Z$ |
| Tire: Longitudinal Force | FX_TIRE | Longitudinal force vs. kappa and $F_Z$ |
| Tire: Overturning Moment | MX_TIRE | Overturning moment vs. alpha and $F_Z$ |
| Tire: Overturning Moment vs. Inclination | MXG_TIRE | Overturning moment vs. kappa and $F_Z$ |
| Generic Table | L_RELAX_X, L_RELAX_Y | Relaxation length as nonlinear function of $F_Z$ and slip. |
| Generic Table | L_CONTACT_X, L_CONTACT_Y | Contact patch dimension as nonlinear function of tire compression. |

The **Tire: Camber Thrust Coefficient** screen defines the camber thrust variable ($dF_Y/d\gamma$) as a conventional VS Configurable Function of vertical load. As with most other Configurable Functions, it can be set to a constant or calculated from the independent variable (load $F_Z$) using a linear coefficient, a nonlinear table, or an algebraic equation provided at runtime. The calculation can include a gain and offset for the camber thrust and a scale factor for load.

Nonlinear functions for relaxation length and contact patch dimensions are not needed by most users of the software, and are usually represented with constant values. When they are defined as functions of other variables, the **Generic Table** library is used, as described in a later section (page 19).

The other Configurable Functions listed in the table always calculate a force or moment using a nonlinear table based on columns and rows of data, where the columns correspond to a load and the rows correspond to alpha, kappa, or gamma.

Prior to version 9 (2015), the table-based tire model assumed symmetric behavior. Tables for calculating tire forces and moments were required to include a column for zero load and a row for zero slip or inclination. To avoid potential errors, the VS Math Models automatically added the necessary column and row of zeros. This behavior occurs when the table type for Configurable Functions is set to a carpet (e.g., `FY_TIRE_CARPET`) with the type `2D_FROM_ZERO`, and is supported in the GUI screen. For example, Figure 1 shows the screen for specifying data for the `FY_TIRE` Configurable Function. As with other Configurable Function screens, it has a drop-down control for specifying how tabular data are interpolated and extrapolated (1). When set to the option **Legacy Tire (2D, absolute slip, from zero)** (2), the keyword `2D_FROM_ZERO` is written so the VS Solver will properly add the column and row of zeros.

On the other hand, if any of the other calculation methods are selected (Figure 2), the interpretation is the same as for any other Configurable Function, with one exception noted below.

**Note** When asymmetric tire data are provided (e.g., Figure 2), the first column of the table should contain zeros, indicating that when the tire leaves the ground (Fz is zero), the Fx, Fy, Mz, or Mx forces and moments must also be zero. If the first Fz in the table is not zero, the VS Solver automatically adds a column of zeros. (In this case, the Echo file will show the modified table.)

When the symmetric option is used, the tabular data should show the relationship between absolute slip angle and absolute lateral force, which forces the origin of the plot axes to the lower-left (Figure 1). For asymmetric data, the signs of the variables should match the ISO sign conventions. In the case of lateral force, increasing slip angle causes increasingly negative side force (Figure 2).

As with all VS Configurable Functions and tables, details for each are provided in the echo file generated by the VS Solver for each run (Figure 3). View the echo file using the **View** button in the lower-right corner of the **Run Control** screen. Notice that in addition to the tire parameters, the comments at the start of the Tires section mention all of the tire-related Configurable Functions that are listed in Table 1.

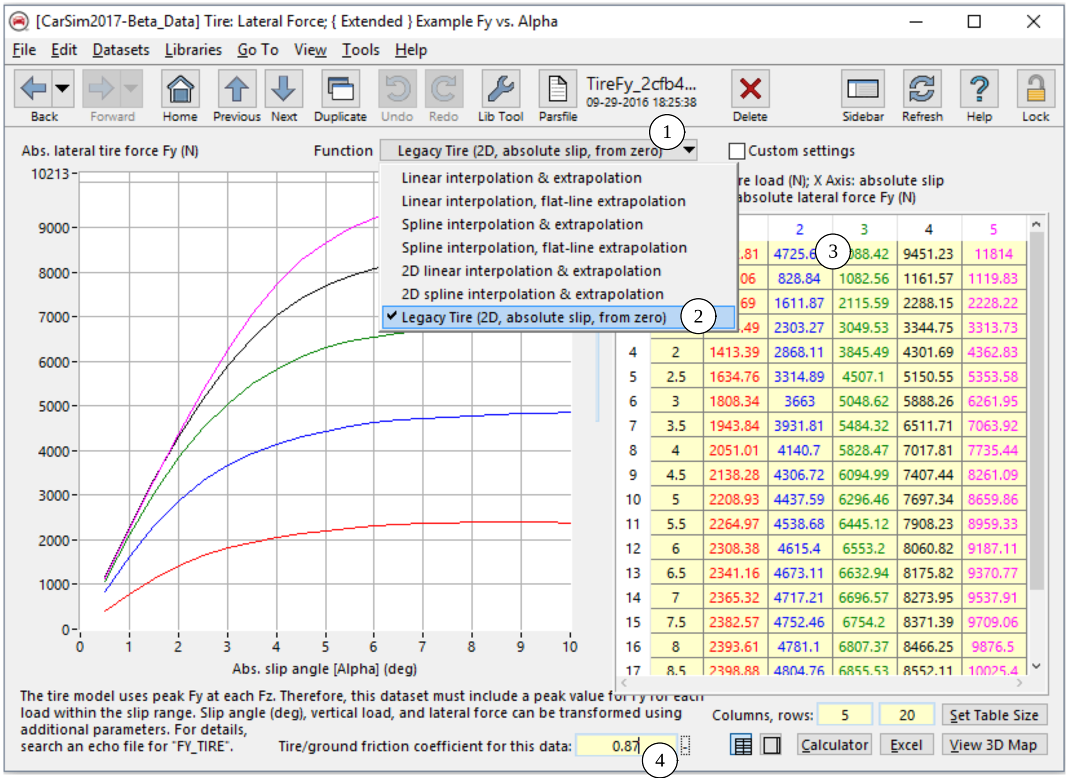


*Figure 1. Tire lateral force screen, showing symmetric data.*

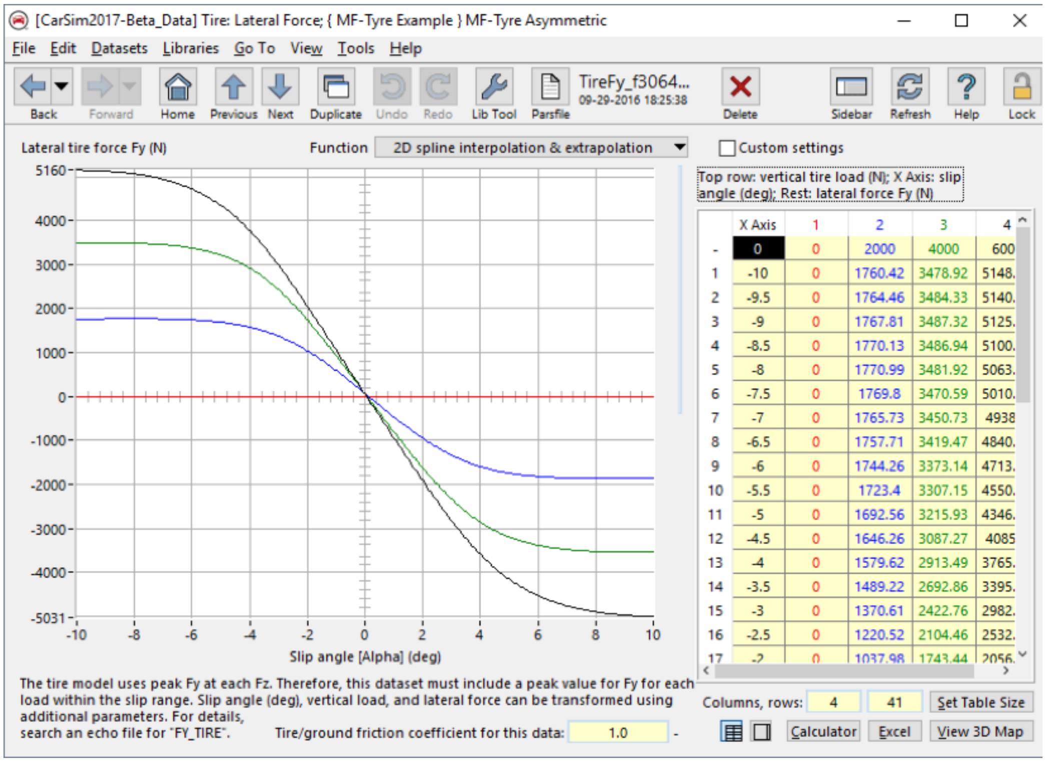
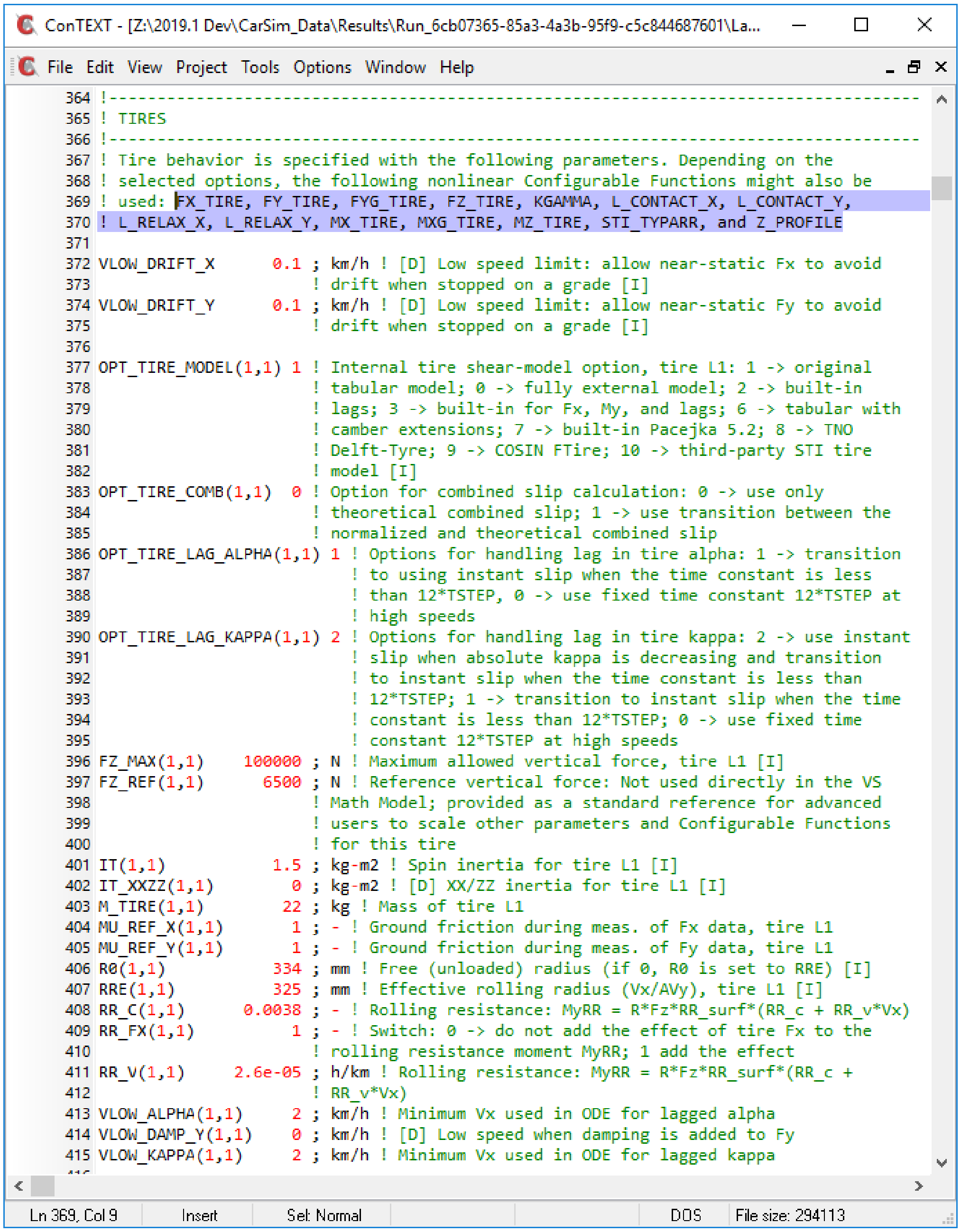


*Figure 2. Tire lateral force screen, showing asymmetric data.*



ConTEXT - [Z:\2019.1 Dev\CarSim_Data\Results\Run_6cb07365-85a3-4a3b-95f9-c5c844687601\La...

File Edit View Project Tools Options Window Help

```
!------------------------------------------------------------------------------
! TIRES
!------------------------------------------------------------------------------
! Tire behavior is specified with the following parameters. Depending on the
! selected options, the following nonlinear Configurable Functions might also be
! used: FX_TIRE, FY_TIRE, FYG_TIRE, FZ_TIRE, KGAMMA, L_CONTACT_X, L_CONTACT_Y,
! L_RELAX_X, L_RELAX_Y, MX_TIRE, MXG_TIRE, MZ_TIRE, STI_TYPARR, and Z_PROFILE

VLOW_DRIFT_X       0.1 ; km/h ! [D] Low speed limit: allow near-static Fx to avoid
                           ! drift when stopped on a grade [I]
VLOW_DRIFT_Y       0.1 ; km/h ! [D] Low speed limit: allow near-static Fy to avoid
                           ! drift when stopped on a grade [I]

OPT_TIRE_MODEL(1,1) 1 ! Internal tire shear-model option, tire L1: 1 -> original
                      ! tabular model; 0 -> fully external model; 2 -> built-in
                      ! lags; 3 -> built-in for Fx, My, and lags; 6 -> tabular with
                      ! camber extensions; 7 -> built-in Pacejka 5.2; 8 -> TNO
                      ! Delft-Tyre; 9 -> COSIN FTire; 10 -> third-party STI tire
                      ! model [I]
OPT_TIRE_COMB(1,1)  0 ! Option for combined slip calculation: 0 -> use only
                      ! theoretical combined slip; 1 -> use transition between the
                      ! normalized and theoretical combined slip
OPT_TIRE_LAG_ALPHA(1,1) 1 ! Options for handling lag in tire alpha: 1 -> transition
                          ! to using instant slip when the time constant is less
                          ! than 12*TSTEP, 0 -> use fixed time constant 12*TSTEP at
                          ! high speeds
OPT_TIRE_LAG_KAPPA(1,1) 2 ! Options for handling lag in tire kappa: 2 -> use instant
                          ! slip when absolute kappa is decreasing and transition
                          ! to instant slip when the time constant is less than
                          ! 12*TSTEP; 1 -> transition to instant slip when the time
                          ! constant is less than 12*TSTEP; 0 -> use fixed time
                          ! constant 12*TSTEP at high speeds
FZ_MAX(1,1)     100000 ; N ! Maximum allowed vertical force, tire L1 [I]
FZ_REF(1,1)       6500 ; N ! Reference vertical force: Not used directly in the VS
                           ! Math Model; provided as a standard reference for advanced
                           ! users to scale other parameters and Configurable Functions
                           ! for this tire
IT(1,1)            1.5 ; kg-m2 ! Spin inertia for tire L1 [I]
IT_XXZZ(1,1)         0 ; kg-m2 ! [D] XX/ZZ inertia for tire L1 [I]
M_TIRE(1,1)         22 ; kg ! Mass of tire L1
MU_REF_X(1,1)        1 ; - ! Ground friction during meas. of Fx data, tire L1
MU_REF_Y(1,1)        1 ; - ! Ground friction during meas. of Fy data, tire L1
R0(1,1)            334 ; mm ! Free (unloaded) radius (if 0, R0 is set to RRE) [I]
RRE(1,1)           325 ; mm ! Effective rolling radius (Vx/AVy), tire L1 [I]
RR_C(1,1)       0.0038 ; - ! Rolling resistance: MyRR = R*Fz*RR_surf*(RR_c + RR_v*Vx)
RR_FX(1,1)           1 ; - ! Switch: 0 -> do not add the effect of tire Fx to the
                           ! rolling resistance moment MyRR; 1 add the effect
RR_V(1,1)      2.6e-05 ; h/km ! Rolling resistance: MyRR = R*Fz*RR_surf*(RR_c +
                           ! RR_v*Vx)
VLOW_ALPHA(1,1)      2 ; km/h ! Minimum Vx used in ODE for lagged alpha
VLOW_DAMP_Y(1,1)     0 ; km/h ! [D] Low speed when damping is added to Fy
VLOW_KAPPA(1,1)      2 ; km/h ! Minimum Vx used in ODE for lagged kappa
```

Ln 369, Col 9 | Insert | Sel: Normal | DOS | File size: 294113

*Figure 3. Tire information listed in Echo file.*

All of the parameters and Configurable Functions that are set on a tire screen are indexed to the axle number as indicated by the current value of the system parameter `IAXLE` and to the side as indicated by the current value of the system parameter `ISIDE` (1 = left, 2 = right). For models with trailers, they are also indexed to the unit as indicated by the current value of the system parameter `IUNIT`.

If the vehicle model supports dual tires, then the tire models are also indexed for the tire on the wheel with the system parameter `ITIRE` (1 = inner, 2 = outer). For example, Figure 4 shows the start of the Tires section in a CarSim Echo file where dual tires are used. The axle parameters with keyword `L_DUAL` indicate whether the axle has single or dual tires (see lines 393 and 394, Figure

4). Also, notice that the tire parameters (e.g., `OPT_TIRE_MODEL`) have an extra index indicating whether the parameter applies for the inner or outer tire on the wheel.

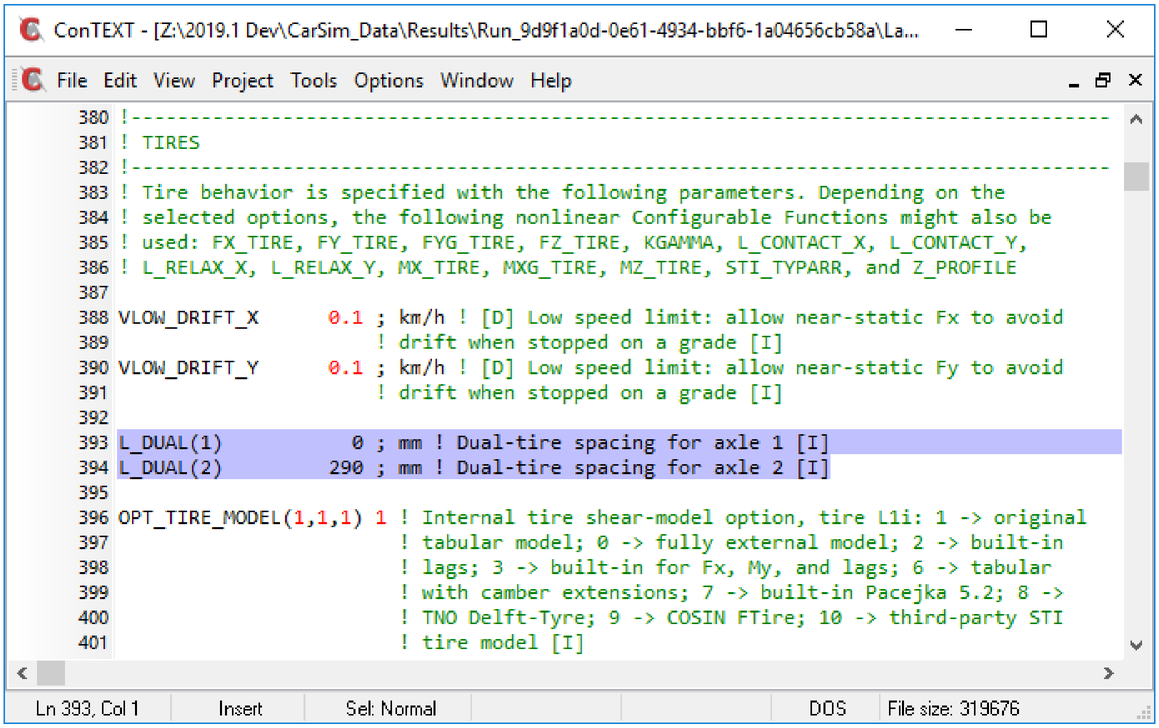



*Figure 4. Tire information in the Echo file if the vehicle model has dual tires.*

## The Main Tire Screen

Figure 5 shows the main tire screen. The appearance of the screen will adjust based on the selected model options.

The basic geometry is shown on the screen, involving a tire/ground reference frame with Z axis perpendicular to the ground at a point called the tire contact center, an X axis perpendicular to both the ground Z and the wheel spin axis, and a Y axis perpendicular to the X and Z axes.

This screen contains six kinds of information about a tire/wheel assembly:

1. vertical force (upper-left part of the screen),
2. rolling resistance (middle-left part of the screen),
3. shear forces and moments (lower-left part of the screen),
4. animator information (upper-right part of the screen),
5. dynamic (transient) properties (middle-right part of the screen), and
6. contact patch dimensions (lower-right part of the screen).

For most of the tire model options, these categories are independent, with the exception that the effective rolling radius is used both for the vertical force calculations and for sizing the shapes shown by the animator.

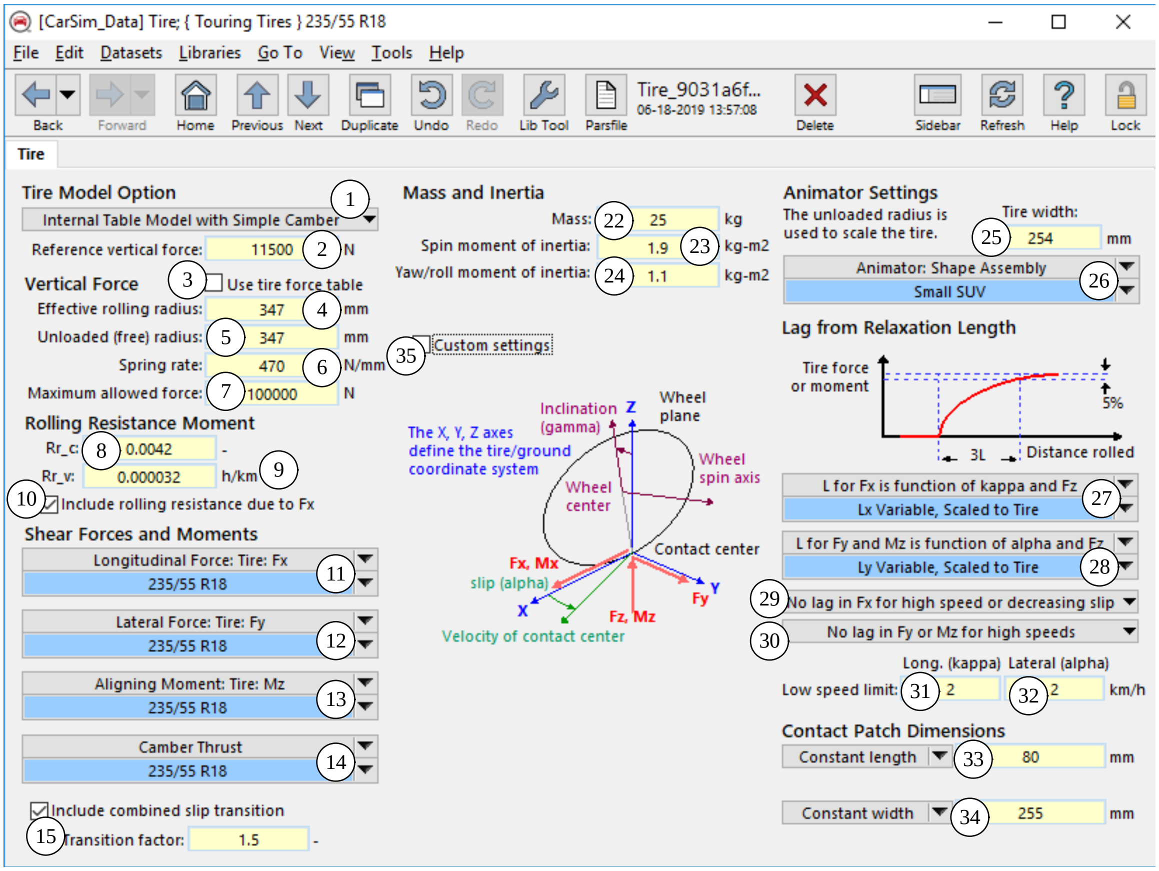



*Figure 5. Tire screen with the first tire model selected (Table Model with Simple Camber).*

## *Tire Model Option*

This screen supports several tire models for predicting rolling resistance and the shear forces and moments.

(1) Model option drop-down list (Figure 6) (keyword = `OPT_TIRE_MODEL`). The selection displays or hides various controls on the screen as need for each optional tire model. For example, when the **Internal Table Model with Simple Camber** is chosen, the lower part of the screen has the appearance shown in Figure 5. When the other items are chosen, the unused controls are hidden and other links or fields are shown.

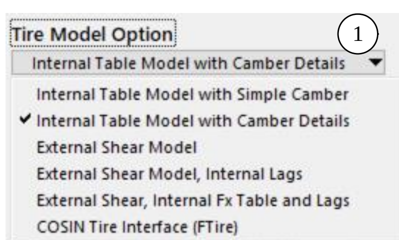



*Figure 6. Model option drop down list*

There are three kinds of tire models available from this list:

1. Two **Internal Table** models built into VS Math Models use tables of the shear forces and moments, as would be measured in a laboratory or on-road tester as functions of vertical load, longitudinal slip, lateral slip, inclination angle, and tire/surface friction. Calculations of forces and moments not covered explicitly in the tables are made using "combined slip theory" and "similarity" [1, 2], as described in section *Tire Force and Moment Calculations* (page 25). The first option (**with Simple Camber**) uses four linked datasets (11 - 14, Figure 5); the second (**with Camber Details**) uses seven linked datasets (Figure 8, page 11).

2. The VS Math Models support external tire models that communicate (import and export) force and kinematical variables with the VS Math Models. Imported forces may be defined at either the tire contact center or the wheel center, to accommodate virtually any tire model (see the subsection **External Shear Models**, page 12).

3. FTire is an advanced external model from COSIN that requires a separate license.

4. VS Math Models also support user defined tire models with TYDEX STI (Standard Tire Interface) and VS STI (VS version of STI), along with external MF-Tyre and MF-Swift models from SIEMENS. These options have been moved to **Tire (External)** screen described in **Connection to third-party tire models** (page 47).

### *Vertical Force and Tire Radius*

The models calculate vertical tire force using a spring that reacts to changes in the tire radius (the distance between the wheel center and the contact center), relative to the undeflected radius. The stiffness of this spring is defined by a linear parameter, a table of vertical load vs. deflection, or a polynomial description suggested by Pacejka.

2 Reference vertical force (keyword = `FZ_REF`). This parameter is not used directly by the VS Math Model, but is provided as a standard property that can be used as a built-in reference for scaling other tire properties that are related to force.

3 Checkbox to select the method for defining tire deflection with vertical load. Checking the box reveals a blue link (Figure 7) to a data set described in the section **Tire: Vertical Load vs. Compression** (page 15). Un-checking the box specifies the use of several parameters that are specified in the data fields shown in Figure 5 (4 – 7).

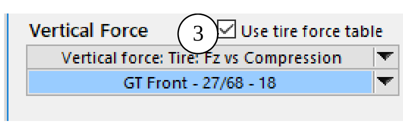



*Figure 7. Link to Vertical Load vs. Deflection dataset*

4 Effective rolling radius (keyword = `RRE`). This is the distance the tire rolls in one revolution at a typical load (usually the rated load), divided by 2π. It is used in the VS Math Models to relate rim speed of the tire to the forward speed at the wheel center.

The animator also uses this radius to size the shapes for the tire and wheel.

This field is visible when checkbox 3 is unchecked.

(5) Tire unloaded (free) radius (keyword = `R0`). This is the radius of the tire at zero load. It is used with the vertical spring rate to determine the vertical force at the tire contact. If this field is left blank, the browser automatically inserts the value specified for the effective rolling radius (4) into it.

This field is visible only when checkbox (3) is unchecked.

(6) Tire vertical spring rate (keyword = `FZ_TIRE_COEFFICIENT`). This coefficient is used to develop a table for tire deflection vs. load, used in the model to calculate change in vertical load due to tire compression. The behavior is linear until the load reaches zero. Once the tire lifts off the ground, the force remains at zero until contact is made again.

This field is visible only when checkbox (3) is unchecked.

(7) Maximum allowed vertical force (keyword = `FZ_MAX`). A maximum value is used to prevent numerical instabilities in some real-time applications that involve severe maneuvers. If the vertical tire force calculated with the spring should exceed this limit, the limit is used as the vertical force.

This field is visible only when checkbox (3) is unchecked.

### *Rolling Resistance Moment*

Selecting one of the **Internal Table** model options or the **External Shear, Internal Fx and Lags** option displays the following controls to specify tire rolling resistance properties (Figure 5). Details of the calculation are provided in the subsection *Rolling Resistance and Power Balance* (page 35).

(8) Tire rolling resistance coefficient (keyword = `RR_C`). This coefficient is used in the equation for rolling resistance on the screen as the **Rr_c** term.

(9) Tire rolling resistance speed coefficient (keyword = `RR_V`). This coefficient is used in the equation for rolling resistance on the screen as the **Rr_v** term.

> **Note** In the equation for rolling resistance, there is a term **Rr_surf** that does not appear on this screen. It is the surface coefficient and is set on the **Road: 3D Surface** screen. **Rr_c** and **Rr_v** are tire properties, while **Rr_surf** is a property of the road surface.

(10) Checkbox to include an additional rolling resistance effect due to $F_X$ and the difference between the effective rolling radius and the current compressed radius of the tire. Checking this box provides a more accurate rolling resistance. It can be unchecked to provide backward compatibility for versions prior to CarSim and TruckSim version 8 (2010).

### *Shear Forces and Moments: Internal Table Models*

Selecting one of the **Internal Table** model options displays links to tabular data (Figure 5). Details of the calculation are presented in the section *Tire Force and Moment Calculations* (page 25).

Most of the linked dataset involve a special version of the VS Configurable Function that is specific to tire data and was described earlier (see Figure 1, page 4). These functions all calculate a force or moment as a function of instant load and slip (lateral alpha or longitudinal kappa) or inclination angle (gamma).

As with other Configurable Functions, all three variables in the table have associated scale factor parameters that can be used to rescale normalized data (e.g., by specifying values in a miscellaneous data field). The scaling parameters all have default values of 1, so they do not have to be set unless additional scaling is needed.

> **Note** Be aware that the tire model compares calculated force values to the peak potential forces when combining lateral and longitudinal forces. The method used to find peak values involves some extra processing that is done by the VS Solver during initialization, after all information is read from the input Parsfiles. If changes are made to a table or scaling property (e.g., `FY_TIRE_GAIN`), they must be made before the initialization. For example, do not change a gain dynamically using VS Command equations; the result would be a mismatch between the new table and the table used during initialization, resulting in invalid tire calculations.

(11) Link to a **Tire: Longitudinal Force** dataset. Along with tabular data that gives values of $F_X$ for columns of constant $F_Z$ and rows of constant longitudinal slip, a parameter is included on the linked screen for the reference coefficient of friction (keyword = `MU_REF_X`) that applies for the data in the table. It is used together with the coefficient of friction specified for the road to scale the tire shear forces. If zero or a negative value is entered for `MU_REF_X`, a value of 1.0 is used.

(12) Link to a **Tire: Lateral Force** dataset. Along with tabular data that gives values of $F_Y$ for columns of constant $F_Z$ and rows of constant lateral slip angle, a parameter is included on the linked screen for the reference coefficient of friction (keyword = `MU_REF_Y`) that applies for the data in the table. It is used together with the coefficient of friction specified for the road to scale the tire shear forces. If zero or a negative value is entered for `MU_REF_Y`, a value of 1.0 is used. If zero or a negative value is entered for `MU_REF_Y`, a value of 1.0 is used.

(13) Link to a **Tire: Aligning Moment** dataset. This screen has a table with values of $M_Z$ for columns of constant $F_Z$ and rows of constant lateral slip angle.

(14) Link to a **Tire: Camber Thrust Coefficient** dataset. This screen has data for a Configurable Function that links camber thrust ($dF_Y/d\gamma$) to vertical load $F_Z$.

(15) Checkbox and data field for a combined slip transition. (keyword for checkbox = `OPT_TIRE_COMB`; keyword for transition factor = `R_TIRE_COMB`). If the box is not checked, the transition is disabled in the model and the corresponding field is hidden.

When used, this factor controls a smooth transition from the direction of the normalized slip to the total theoretical slip, as described in the subsection *Combined Slip Theory* (page 29). With the method of Bakker, Pacejka, and Lidner [1], a normalized combined slip is used to determine the direction of the total horizontal force at small slip, while a theoretical combined slip is used at large slip.

The factor is denoted as $q_1$ in equation 19 (page 32). If this factor is zero, the direction of the normalized slip is always applied, while if this factor is ∞, the direction of the total theoretical slip is always applied. This factor should be tuned in order to match the actual tire characteristics. If this factor is too small, the model might be less accurate for large slip; if the factor is too large, the model might be less accurate for small slip (this can be seen by a dent in the Fy - Fx friction ellipse).

If the box not checked, the total theoretical slip is always used to determine the direction. (When unchecked, calculations match the methods used in CarSim and TruckSim versions prior to 2011.)

### *Shear Forces and Moments: Internal Model with Camber Details*

Selecting the model option **Internal with Camber Details** (1) displays all fields and links shown in Figure 5 except the Camber Thrust link (14). Three additional links are shown for tabular data that define the effects of tire inclination (gamma) on lateral force, aligning moment, and overturning moment (Figure 8). Also, a link is shown for tabular data that defines overturning moment $M_X$ as a function of load and lateral slip.

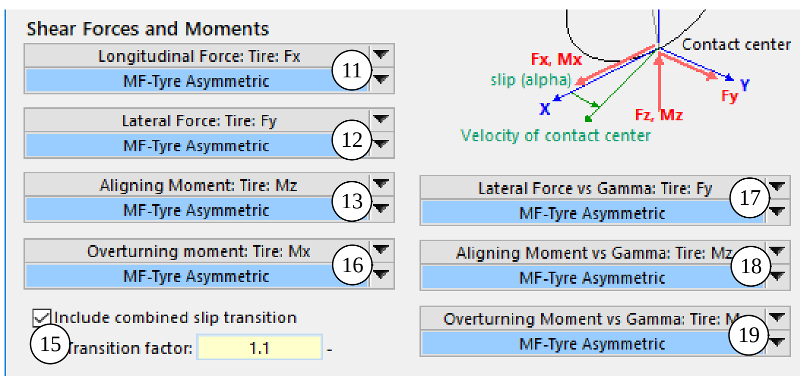



*Figure 8. Appearance of tire screen when Camber Details are selected.*

(16) Link to a **Tire: Overturning Moment** dataset. This screen has a table with values of $M_X$ for columns of constant $F_Z$ and rows of constant lateral slip angle.

(17) Link to a **Tire: Lateral Force vs. Inclination** dataset. Along with tabular data that gives values of $F_Y$ for columns of constant $F_Z$ and rows of constant inclination angle, a parameter is included on the linked screen to control the rate at which the camber effects on lateral force diminish with slip (keyword = `CRO_FY`).

(18) Link to a **Tire: Aligning Moment vs. Inclination** dataset. Along with tabular data that gives values of $M_Z$ for columns of constant $F_Z$ and rows of constant inclination angle, a parameter is included on the linked screen to control the rate at which the camber effects on aligning moment diminish with slip (keyword = `CRO_MZ`).

(19) Link to a **Tire: Overturning Moment vs. Inclination** dataset. Along with tabular data that gives values of $M_X$ for columns of constant $F_Z$ and rows of constant inclination angle, a parameter is included on the linked screen to control the rate at which the camber effects on overturning moment diminish with slip (keyword = `CRO_MX`).

### Shear Forces and Moments: External Shear Models

The tire model option drop-down list 1 (Figure 6, page 7) includes three options that involve an external (user-defined) shear model, linked through one of the many methods available to extend VS math models. In all three cases, the data links for lateral force, aligning moment, and camber thrust are hidden because they are not used.

Depending on the option selected, controls related to dynamics and longitudinal forces may be visible or hidden.

21 Cornering stiffness of tire used by driver path-follower model (keyword = `KY_TIRE`). The built-in driver model may use tire cornering stiffness to determine the steering wheel angle needed to follow a target path. When the built-in tire models are used, cornering stiffness is obtained automatically. When the internal models are not used, the cornering stiffness must be provided if the driver model will be used with `OPT_DM` = 1 or `OPT_DM` = 2. (Cornering stiffness is not needed by the driver model when `OPT_DM` = 3.)

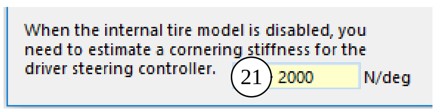



*Figure 9. Cornering stiffness for the driver model when using an external tire model.*

### Mass and Inertia

The mass and inertia of the tire/rim object that is mounted on the suspension wheel are associated with the tire dataset, and are added to the spinning part of the vehicle unsprung mass.

22 Mass of the tire and rim (keyword = `M_TIRE`).

23 Spin moment of inertia of the tire and rim (keyword = `IT`).

24 Roll/yaw moment of inertia of the tire and rim (keyword = `IT_XXZZ`).

### Animator Settings

The visual appearance of the tire and wheel in the animator is typically specified with an animator dataset supplying shape information for the tire and wheel assembly. Audio information can also be provided to generate squeal noises during high slip conditions.

25 Tire width (keyword = `SET_THICKNESS`). This dimension is used only to generate animations of the wheels; it has no influence on the simulated vehicle response.

26 Optional link to animator data to show the tire/wheel assembly in the animation.

The wheel reference frame that provides X-Y-Z coordinates and rotation information to the animator is provided by the vehicle screen (vehicle assembly or trailer) that links to this screen; this information provides the shape for the rotating tire/wheel assembly. By convention, shapes for the left side of the vehicle are shown as defined, while those for the right side are flipped (so the wheel hubs always face outside). The linked shapes are scaled according to the effective tire rolling radius 4 and width 25.

### *Dynamic Properties*

Tires do not immediately generate forces when they develop lateral slip angles and/or longitudinal slip ratios: they must roll some distance to generate carcass and tread deflections necessary to sustain forces. Under step inputs in slip, lateral and longitudinal forces build up approximately as first-order lags in distance. Distances known as relaxation lengths characterize the delays, as described in the subsection *Tire Transient Behavior (Lags and Relaxation Length)* (page 36).

The middle right part of the screen has settings that affect the transient response of the tire (Figure 5 and Figure 10).

27 Tire longitudinal relaxation length is about one-third the distance that the tire must roll before the longitudinal force due to slip ratio builds up to its full value. It is a spatial version of a time constant. There is little or no directly measured data for this property, but the consensus of tire experts is that it is shorter than the more easily measured lateral relaxation length 28, that it increases with vertical load, and that it decreases when longitudinal slip ratio magnitude increases. It is represented in the VS Math Model with a Configurable Function whose root keyword is `L_RELAX_X`.

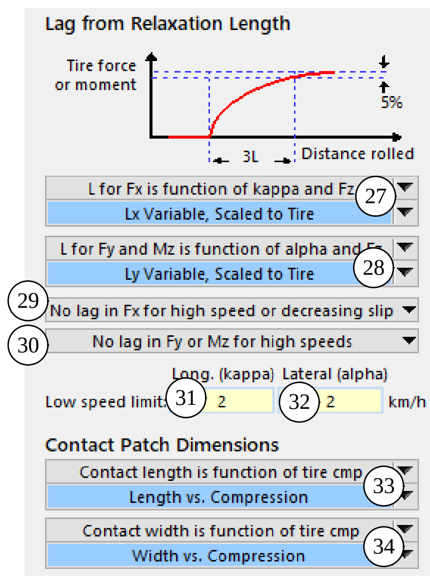



*Figure 10. Links to datasets with Configurable Functions for variable lengths.*

The drop-down control provides three means for defining the relaxation length (Figure 11):

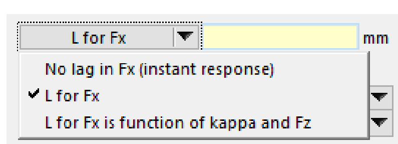



*Figure 11. Options for defining relaxation length.*

1. No dynamic lag (the tire response is instant, specified with the setting `L_RELAX_X_CONSTANT` = 0).
2. A constant relaxation length (keyword = `L_RELAX_X_CONSTANT`).

3. Relaxation length is a nonlinear function of vertical load and lagged kappa (longitudinal slip ratio with dynamic lag).

If relaxation length is constant, a yellow field appears (Figure 11). If the function is chosen, a data link appears to the **Generic Table** library (Figure 10). An example Configurable Function dataset is described in a later subsection (page 19).

If the option is selected for no lag in $F_X$, then the controls that specify behavior at high and low speed limits (29 and 31) are hidden.

28 Tire lateral relaxation length is about one-third the distance that the tire must roll before the lateral force due to slip angle builds up to its full value. It is a spatial version of a time constant. Tire lateral relaxation length typically has a value of about twice the rolling radius of the tire at normal loads and with small slip angles. Measures over a range of conditions typically show that lateral relaxation length increases with vertical load, and that it decreases when lateral slip magnitude increases [6, 7]. It is represented in the VS Math Model with a Configurable Function whose root keyword is `L_RELAX_Y`.

The drop-down control for lateral dynamic lag has the same three options available for longitudinal lag (Figure 11), as described above, except that lagged lateral slip angle is considered (lagged alpha) rather than lagged longitudinal slip (lagged kappa).

If the option is selected for no lag in $F_Y$ and $M_Z$, then the controls that specify behavior at high and low speed limits (30 and 32) are hidden.

29 Drop-down list for longitudinal lag option (keyword = `OPT_TIRE_LAG_KAPPA`). The differential equation for lag in longitudinal force generation is based on the assumption of a rolling tire, with the dynamic lag being within the frequency range of the overall vehicle math mode. When these conditions do not apply, adjustments are made to the differential equations based on the parameter `OPT_TIRE_LAG_KAPPA`.

A time constant is calculated internally using the relaxation length divided by the forward speed of the wheel center. If the time constant becomes so short that the math model would become numerically unstable, an adjustment is made.

There are also situations where the differential equation can lead to unrealistic behavior, in which energy is put into a spinning wheel undergoing significant angular acceleration. This can occur with spinning under power on ice, or with rapid braking cycles controlled by ABS.

The drop-down control offers three options for handling these situations (Figure 12).

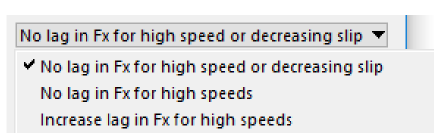



*Figure 12. Options for handling lag in Fx for high speed and decreasing slip.*

1. The first option on the list is the default (`OPT_TIRE_LAG_KAPPA` = 2). In this case, the Math Model degree of freedom (DOF) is removed for the lagged kappa at high speeds, and instant longitudinal slip (kappa) is used. At speeds below this transition,

lag is reduced when the longitudinal slip ratio is decreasing. This is done to reduce a side effect of the longitudinal relaxation length concept in which energy can be stored and put back into the spinning wheel when it is undergoing significant angular acceleration.

2. The second option on the list (`OPT_TIRE_LAG_KAPPA` = 1) removes the lag DOF at high speeds and always uses the full lag at lower speeds.
3. With the third option (`OPT_TIRE_LAG_KAPPA` = 0), the DOF is retained at all times, and the lag is increased to maintain a time constant that maintains numerical stability, such that the time constant is not allowed to be drop below a limit based on the simulation time step.

(30) Drop-down list for lateral lag option (keyword = `OPT_TIRE_LAG_ALPHA`). The differential equation for lag in lateral force generation is based on the assumption of a rolling tire, with the dynamic lag being within the frequency range of the overall vehicle math mode. If the relaxation length is short (due to light loading and/or high slip angle), and/or if the speed is high, then the time constant (defined as the relaxation length divided by the speed) becomes too small and can cause numerical instability. The tire model has two options to maintain numerical stability:

1. The first option on the list is the default (`OPT_TIRE_LAG_ALPHA` = 0); in this case the lag DOF is removed at high speeds and the instant slip angle is used.
2. With the second option (`OPT_TIRE_LAG_ALPHA` = 1), the DOF is retained at all times, and the lag is increased to maintain a time constant that maintains numerical stability, such that the time constant is not allowed to be drop below a limit based on the simulation time step.

(31) Minimum speed used in longitudinal relaxation differential equation (keyword = `VLOW_KAPPA`). The definition of longitudinal slip ratio (kappa) involves division by the forward speed of the wheel center. At very low speeds, small changes in angular velocity cause large changes in slip ratio, which in turn can result in unrealistically harsh changes in $F_X$. To reduce the harshness, this limit speed is used in lag differential equation.

This setting is not shown if the drop-down control (27) is set to disable lag in $F_X$.

(32) Minimum speed used in lateral relaxation differential equation (keyword = `VLOW_ALPHA`). The definition of lateral slip angle (alpha) involves division by the forward speed of the wheel center. At very low speeds, small changes in lateral velocity cause large changes in slip angle, which in turn can result in unrealistically harsh changes in $F_Y$ and $M_Z$. To reduce the harshness, this limit speed is used in the lag differential equation.

This setting is not shown if the drop-down control (28) is set to disable lag in $F_Y$ and $M_Z$.

### *Contact Patch Dimensions*

A tire contacts the ground over a small area called the contact patch. All of the built-in tire models calculate forces and moments based on an assumption that the ground is flat, but not necessarily level, in the contact patch. When the ground is not flat, the models use several points to calculate a virtual tire contact center with elevation and slope to represent the contact patch area. The models support four options:

1. A single point (the contact center).
2. Two points at the forward and rear limits of the contact patch, spaced by a specified length (the contact center is at the center of these two points). In this case, the contact patch width is zero.
3. Two points at the left and right limits of the contact patch, spaced by a specified width (the contact center is at the center of these two points). In this case, the contact patch length is zero.
4. Four points at the corners of a rectangular contact patch, spaced by a specified length width (the contact center is at the center of these four points).

33 Tire contact length (root keyword = `L_CONTACT_X`). As with the relaxation length drop-down controls, the length and width of the contact patch can be set with drop-down controls that have three options (Figure 13):

1. Contact length = 0.
2. A constant length.
3. Length is a nonlinear function of tire compression.

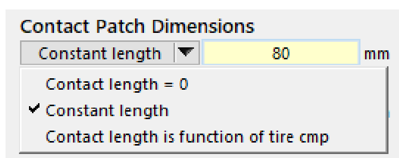



*Figure 13. Options for defining a contact patch dimension.*

34 Tire contact width (root keyword = `L_CONTACT_Y`). The drop-down control for contact width has the same three options available for contact length (Figure 13), as described above.

### *User-Defined Data Field and Link*

Advanced users can provide custom settings such as parameters for externally defined models, VS commands, etc.

35 Checkbox to show a miscellaneous field 25 and link 26 (Figure 14).

36 Miscellaneous field. This is a location for advanced users to specify parameters for the tire model or animation. It can be used to extend the model with VS commands. When the Internal Table models are selected, this field can be used to scale the data by setting parameters associated with the Configurable Functions. (See the *VS Math Models Reference Manual* for more information about Configurable Functions).

This field is visible only when checkbox 24 is checked.

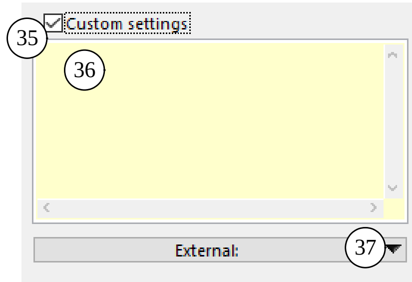



*Figure 14. Miscellaneous field and link for custom settings.*

The location of this field depends on the model option (1). Regardless of the location on the screen, the contents are always written to the parsfile after all other data on the screen (except for the miscellaneous link (26)). Therefore, it can be used to override settings from other links if needed for advanced applications.

(36) Miscellaneous link. This link can be used to add data for any tire model, including user-defined extensions, externally defined tire models, and external tire models from MF-Tyre/MF-Swift and FTIRE.

This link is visible only when checkbox (24) is checked.

If this link is used, the linked dataset is sent to the math model after all other data from this screen. Therefore, it can be used to override settings if needed for advanced applications.

## Tire: Vertical Load vs. Deflection

The relationship between tire vertical load and deflection usually can be defined with a linear stiffness parameter. On the other hand, some tires exhibit significantly non-linear changes in force with deflection. Examples include tires undergoing extreme variations in load and tires with very low vertical stiffness. When the checkbox (3) (Figure 7) labeled **Use tire force table** is checked on the **Tire** screen, a link to the **Tire: Vertical Load vs. Deflection** screen (Figure 15) is displayed.

### *Information for the Vehicle Model*

The vehicle model will use a few parameters and the tabular data from this screen, described below.

(1) Effective rolling radius (keyword = `RRE`). This is the distance the tire rolls in one revolution at a typical load (usually the rated load), divided by 2π. It is used in the simulation model to relate rim speed of the tire to the forward speed at the wheel center.

This radius is also used by the animator to size the shapes used to create the tire and wheel.

(2) Tire unloaded radius (keyword = `R0`). This is the radius of the tire at zero load. It is used with the vertical spring rate to determine the vertical force at the tire contact. If this field is left blank, the browser automatically inserts the value specified for the effective rolling radius into it.

(3) Maximum allowed vertical force (keyword = `FZ_MAX`). A maximum value is used to prevent numerical instabilities in some real-time applications that involve severe maneuvers. If the vertical tire force calculated with the spring should exceed this limit, the limit is used as the vertical force.

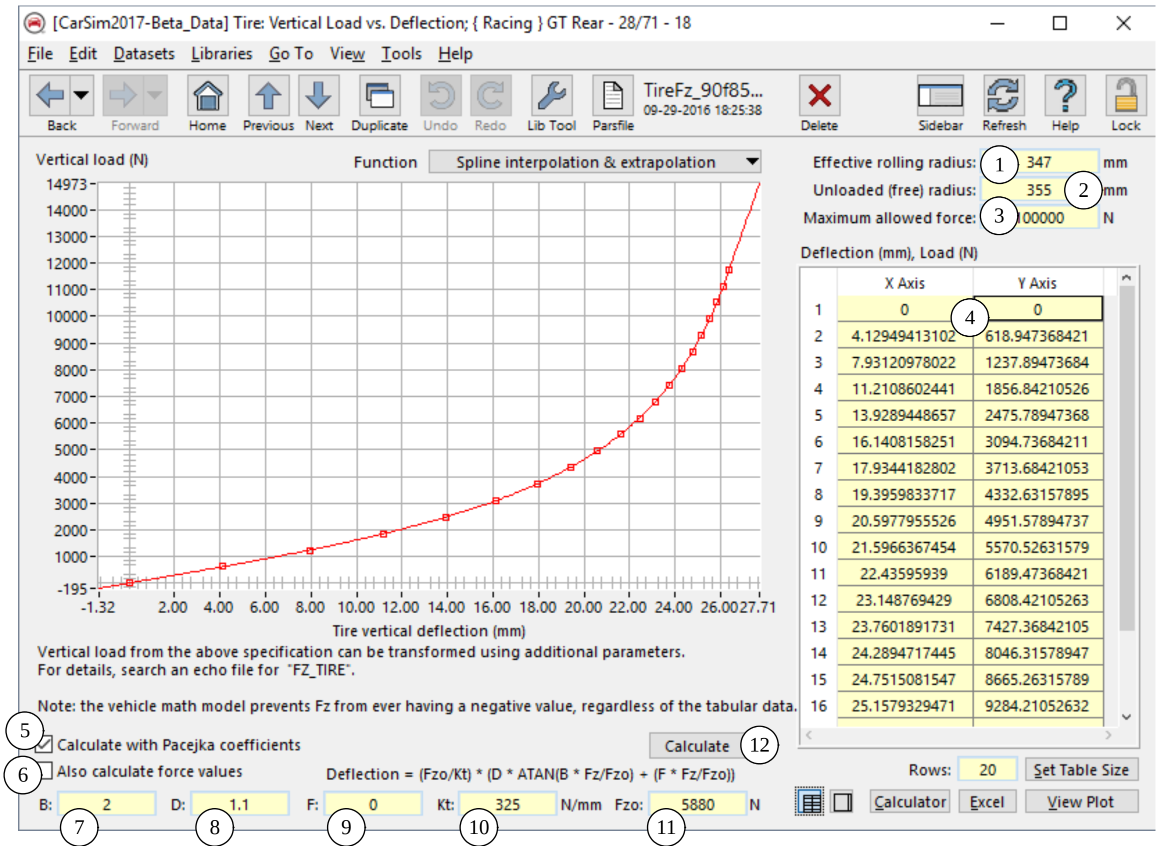



*Figure 15. The Tire: Vertical Load vs. Deflection screen.*

(4) Table of vertical deflection (mm) and vertical force (N) (root keyword = `FZ_TIRE`). This table includes the options for viewing and editing tables that are described in the *VehicleSim Browser Reference Manual*.

### *Information for the Calculator*

This screen has an option to calculate tabular data. Parameters described below are used for the calculation but are not sent to the math model; they are used only to generate the tabular data (4).

If checkbox (5) is checked, the values displayed can be calculated from an equation by clicking the **Calculate** button (12).

Although the screen might show extrapolated force going negative, VS Math Models limit the minimum tire vertical force to zero to account for the case of the tire leaving the ground.

(5) Checkbox to select a Pacejka Magic Formula method to define table values. The equation used for the calculation is displayed on the screen.

If the checkbox is not selected, the deflection of the tire under load is simply specified by filling in the table (4).

In either case, the data sent to the math model is the table of values (4). The Magic Formula calculation is only used to generate the table.

When the box is checked, controls 6 through 12 are displayed. When it is not checked, they are hidden.

6 Checkbox to specify the method for defining the vertical force increment. When the box is checked, deflections are calculated at equal load increments from zero to the maximum allowed force specified 3. The number of rows in the table determines the number of points. When the box is not checked, the calculator uses loads typed into the "Y Axis" column of the spreadsheet control.

7 Parameter "B" in the Magic Formula equation for load versus deflection (database keyword = `PAC_B`).

8 Parameter "D" in the Magic Formula equation for load versus deflection (database keyword = `PAC_D`).

9 Parameter "F" in the Magic Formula equation for load versus deflection (database keyword = `PAC_F`).

10 The nominal vertical stiffness coefficient for the tire, used in the Magic Formula equation for load versus deflection (database keyword = `PAC_KT`).

11 The nominal vertical load for the tire, used in the Magic Formula equation for load versus deflection (database keyword = `PAC_FZO`).

12 **Calculate** button. After selecting the calculation option, setting the table size, entering calculation parameters, and possibly specifying load points, click this button to calculate the deflection at each load and update the table.

## Setting Variable Relaxation Length or Contact Patch Size

Four of the tires properties that can be represented with Configurable Functions are often given constant values. These are the relaxation lengths (lateral and longitudinal) and contact patch dimensions (length and width). As described earlier, they can also be defined with Configurable Functions, in which case the yellow fields are replaced with blue links to datasets from the **Generic Table** library (Figure 10, page 13).

In order to use the **Generic Table** library, you must specify keywords in order for the VS Solver to identify exactly which Configurable Function is being set. The main resource for finding these keywords is the echo file generated with each simulation run. For example, the Echo file shown in Figure 3 on page 5 identified the Configurable Functions used in the tire model, including one named `L_RELAX_Y`. Search the Echo file for this name (`L_RELAX_Y`) to find more information (Figure 16).

The comments tell what the function does, what keywords are used, what calculation methods are supported, and how the indexing is defined. For this example, there are three indices corresponding the system parameters `IAXLE`, `ISIDE`, and `ITIRE`. Gains and offsets can be applied to the calculated length using `L_RELAX_Y_GAIN` and `L_RELAX_Y_OFFSET`. The independent variables (Fz and Lagged slip) can also be scaled using `FZ_SCALE_L_RELAX_Y` and `LAGGED_ALPHA_SCALE_L_RELAX_Y`.

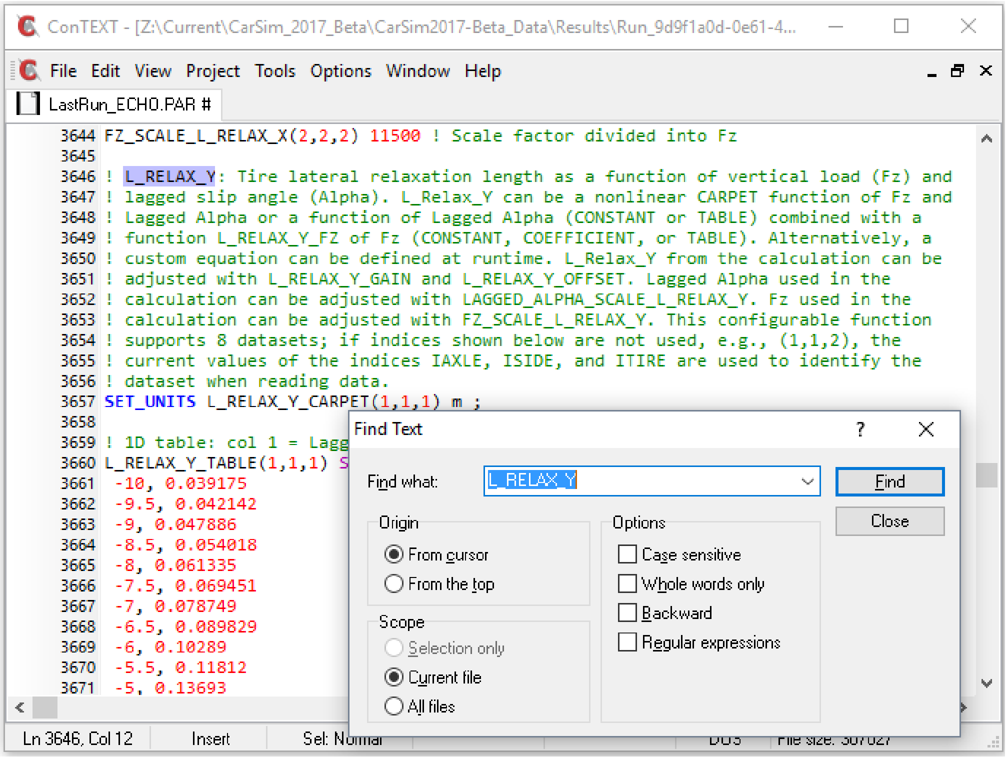



*Figure 16. Search the Echo file for the Configurable Function L_RELAX_Y.*

Figure 17 shows how a dataset for lateral relaxation length is set up in the **Generic Table** library to define normalized tables that are scaled to tire properties.

> **Note** This dataset was set up to be scaled automatically to work with any tire dataset in which the `FZ_REF` value was set (2 in Figure 5, page 7). To do so, it makes use of what might be considered "advanced features" of Configurable Functions.

As with other table screens, the **Generic Table** screen has drop-down controls to specify calculation methods (4, 6, and 7) and spreadsheets with tabular data. In this case, the relaxation length is defined by multiplying two 1D table functions together. The root keywords for the two functions are entered in the data fields (1 and 3) on the screen, the box to enable the second variable is checked 2, and the option is selected to multiply the two function results together 6. When the multiplication option is used, the second function is redefined internally as a dimensionless ratio.

Two of the scale factors for the Configurable Function are set in the miscellaneous yellow field 5. The output gain `L_RELAX_Y_GAIN` is set to twice the effective rolling radius, `RRE`. The vertical load, used as the independent variable for the bottom table, is divided by the scale factor `FZ_SCALE_L_RELAX_Y`, which is set to the tire reference vertical force `FZ_REF`. Therefore, a value of 1.0 on the horizontal axis for the bottom table corresponds to a vertical force of `FZ_REF`.

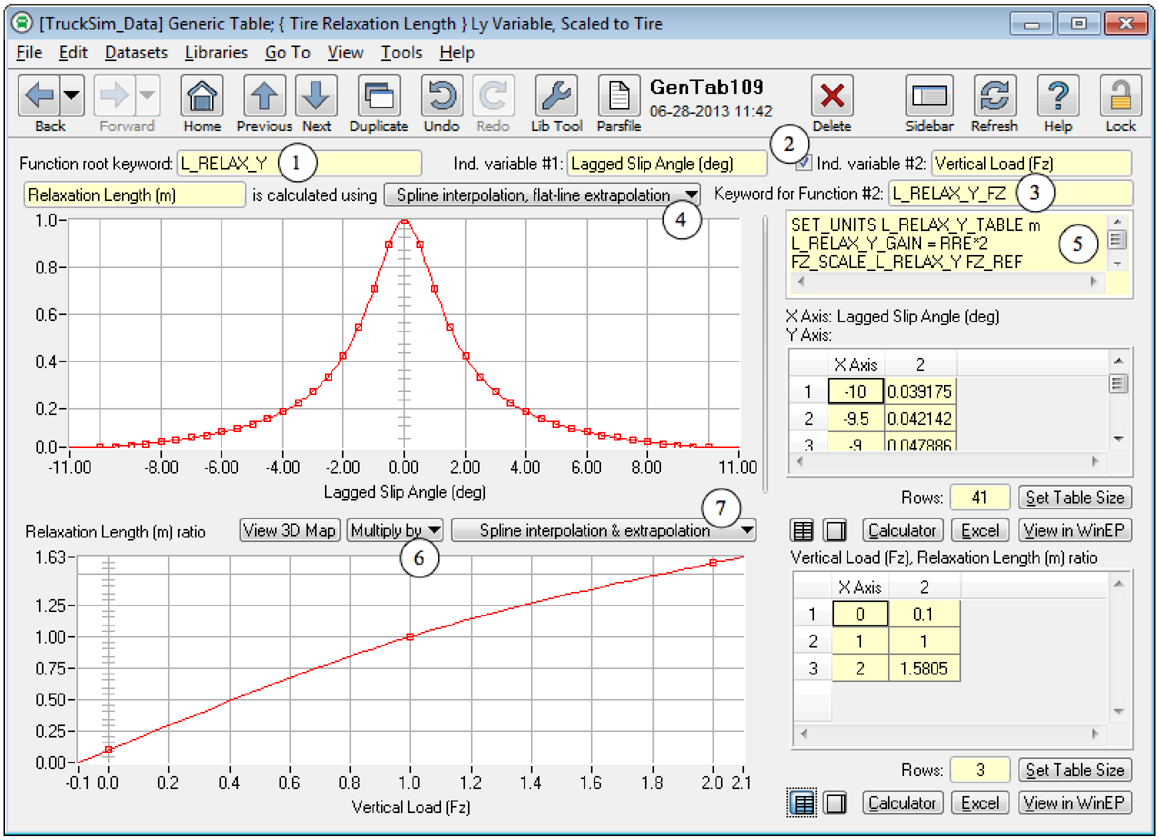



*Figure 17. Using the Generic Table screen to specify variable relaxation length.*

In order to simply the scaling concepts, the top table has a maximum value of 1.0 at zero slip; the units for the table are set to meter (the internal SI units of length) with the `SET_UNITS` command 5.

More information about using the **Generic Table** screen is provided in the documentation for that screen.

## Groups of Dual Tires

Each axle in TruckSim has two wheels that are either set up for single or dual tires. The tires are not necessarily the same, so there is sometimes a need to specify four distinct tire datasets for an axle. For vehicle units with more than a single axle, there is not enough room for those four links per axle. Therefore, tire group screens are used for all TruckSim vehicle units with more than one axle to select whether the wheels are single or dual, and to specify which tire data sets are used.

TruckSim has four library screens for selecting the tires that go on a vehicle unit with two or more axles. The libraries are:

- **Tires: Vehicle Unit with 2 Axles**
- **Tires: Vehicle Unit with 3 Axles**
- **Tires: Vehicle Unit with 4 Axles**
- **Tires for Any Vehicle Unit**

### *CarSim Dual Tire Option*

CarSim models typically have wheels with single tires, which are specified on the vehicle assembly or trailer screens. However, starting with version 2017, CarSim also supports dual tires. In this case, the links to tire datasets are also made on a tire group screen **Tires: 2 Axles with Support for Duals** (Figure 18) that can be used for the four-wheeled lead unit or trailer with one or two axles. The CarSim option to use dual tires is activated by linking to a dataset from this library. Doing so causes the CarSim VS Browser (`carsim.exe`) to install the dual tire option before the math model (VS Solver) reads the input Parsfile, such that options involving dual tires are properly recognized.

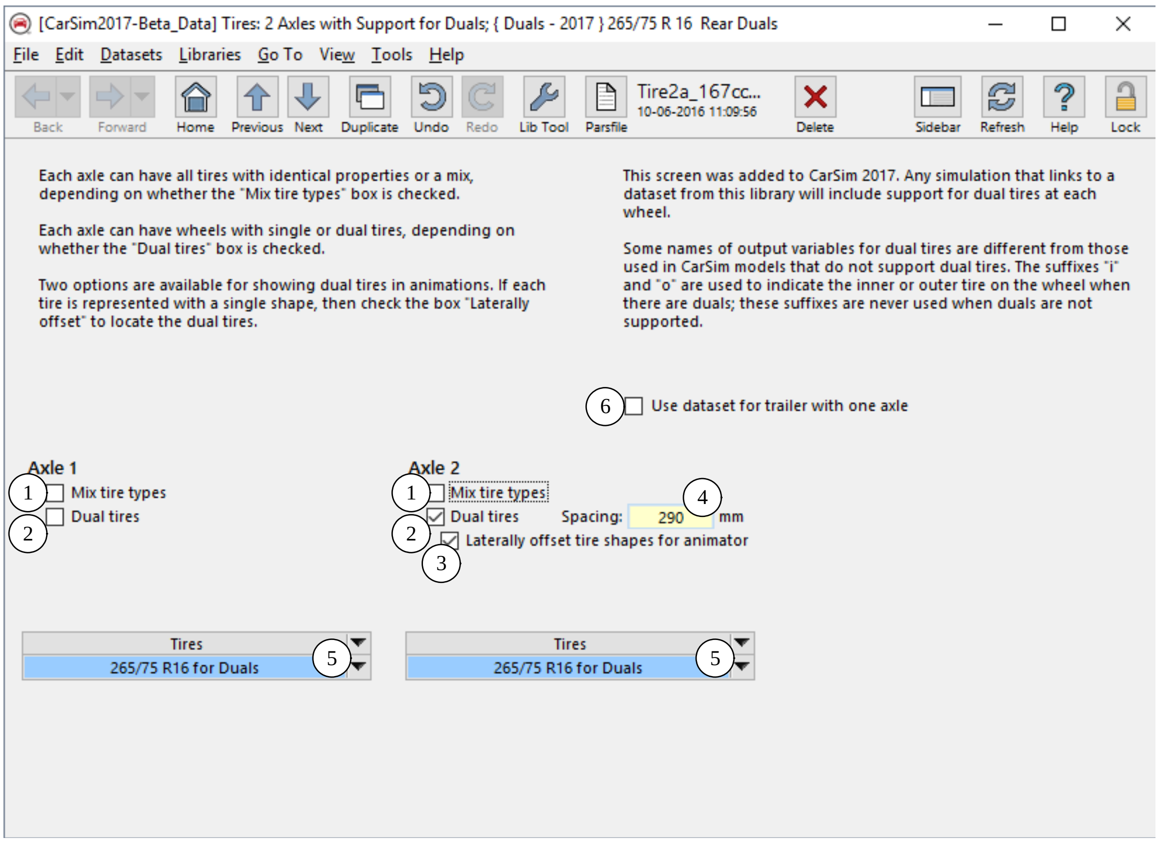



*Figure 18. Selection of tires for a vehicle unit.*

### *User Controls*

1 **Mix Tire Types** checkbox. When checked, all tires on the axle (either two or four, depending on whether the **Dual tires** box 2 is checked) can be specified with separate links (Figure 19). When not checked, a single link is shown (Figure 18) and the specified data set is copied for all of the tires on the axle.

2 **Dual tires** checkbox. When checked, a yellow field 4 is shown for specifying the dual spacing. When not checked, the yellow field is hidden and a value of 0 is written in the data file, indicating that this axle has wheels with single tires.

③ Checkbox for offsetting tire shapes laterally for the animator. For routine use, this box should be checked. When this is checked, the shape information associated with a tire data set is used twice. One copy is shifted by half the dual spacing inward and the other is shifted by the same amount outward. Also, the shapes are rotated 180° for dual tire assemblies so the wheels face inward, giving a concave appearance typical of dual assemblies. If this box is not checked, then the animation information associated with the tires is used without modification, as might be needed for some advanced animation presentations.

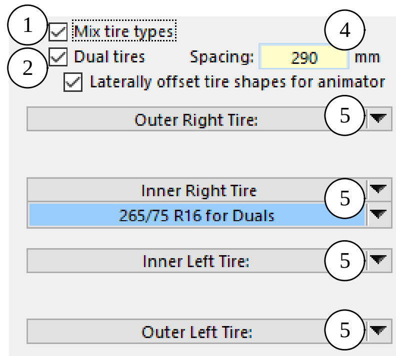



*Figure 19. Option to specify four distinct tire datasets for an axle.*

④ Spacing for dual tires (keyword = `L_DUAL`) When the **Dual tires** box is checked ②, this value specifies the spacing between tires, as measured center-to-center.

⑤ Link(s) to a **Tire** data set. Depending on whether the axle has single or dual tires, and on whether the tire types are mixed, there can be one, two, or four tire links shown for an axle.

⑥ Checkbox for specifying that the dataset is for a single-axle trailer. This control exists only in the CarSim screen shown in the figure. When checked, settings for Axle 2 are hidden.

# The Tire Tester Utility

Tire properties are defined with a combination of tabular data and parameters. The input data typically describes is single operating condition, such as free rolling or zero slip. Often it is useful to examine the predicted performance of a tire under conditions involving various combined slip conditions, different loads, inclination angles, or surface coefficients of friction. The **Tire Tester** utility calls the tire models to obtain predicted performance under user-specified conditions. It simulates the behavior of a steady-state tire test machine.

| | |
|---|---|
| **Note** | The Tire Tester presently supports the standard and extended lookup table models, MF-Tyre/MF-Swift and any other user defined tire model. |

To use the tire tester, select the **Tire** library under the **Test Specifications** section on the **Run Control** screen and choose a tire dataset (Figure 20).

Select the **Tire Tester** library under the **Procedure** link and choose a test dataset (Figure 21).

The Tire Tester can be configured to define four kinds of tests (Figure 22).

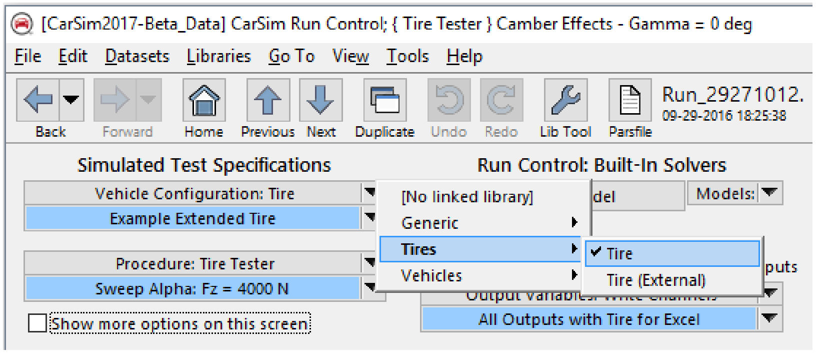



*Figure 20. Select a tire dataset for the tire tester.*

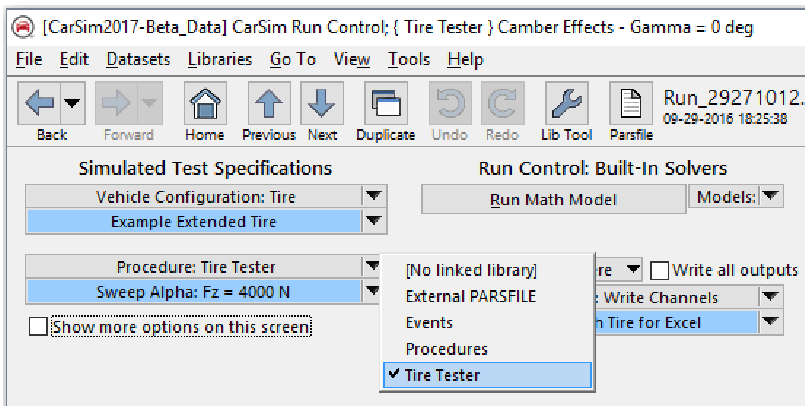



*Figure 21. Select a tire test procedure.*

(1) Drop down list to select the test mode for the tire tester. The four available modes are:

**Sweep Vertical Load**. Slip angle, slip ratio, and inclination angle are held at constant user-specified values while vertical load is incremented between user-specified limits.

**Sweep Slip Angle**. Vertical load, slip ratio, and inclination angle are held at constant user-specified values while slip angle is incremented between user-specified limits.

**Sweep Slip Ratio**. Slip angle, vertical load, and inclination angle are held at constant user-specified values while slip ratio is incremented between user-specified limits.

**Sweep Inclination Angle.** Slip angle, slip ratio, and vertical load are held at constant user-specified values while inclination angle is incremented between user-specified limits.

(2) Depending on the mode selected, data fields are displayed for the upper limit, lower limit, and step size of the variable to be swept, and for the values of the remaining variables to be held fixed.

Data fields are also displayed for the longitudinal and lateral coefficients of friction. To use the data exactly as it is entered in the property tables without modifying it for surface friction, enter -1.

A field for the test speed is also present, but in the steady state test it is not used.

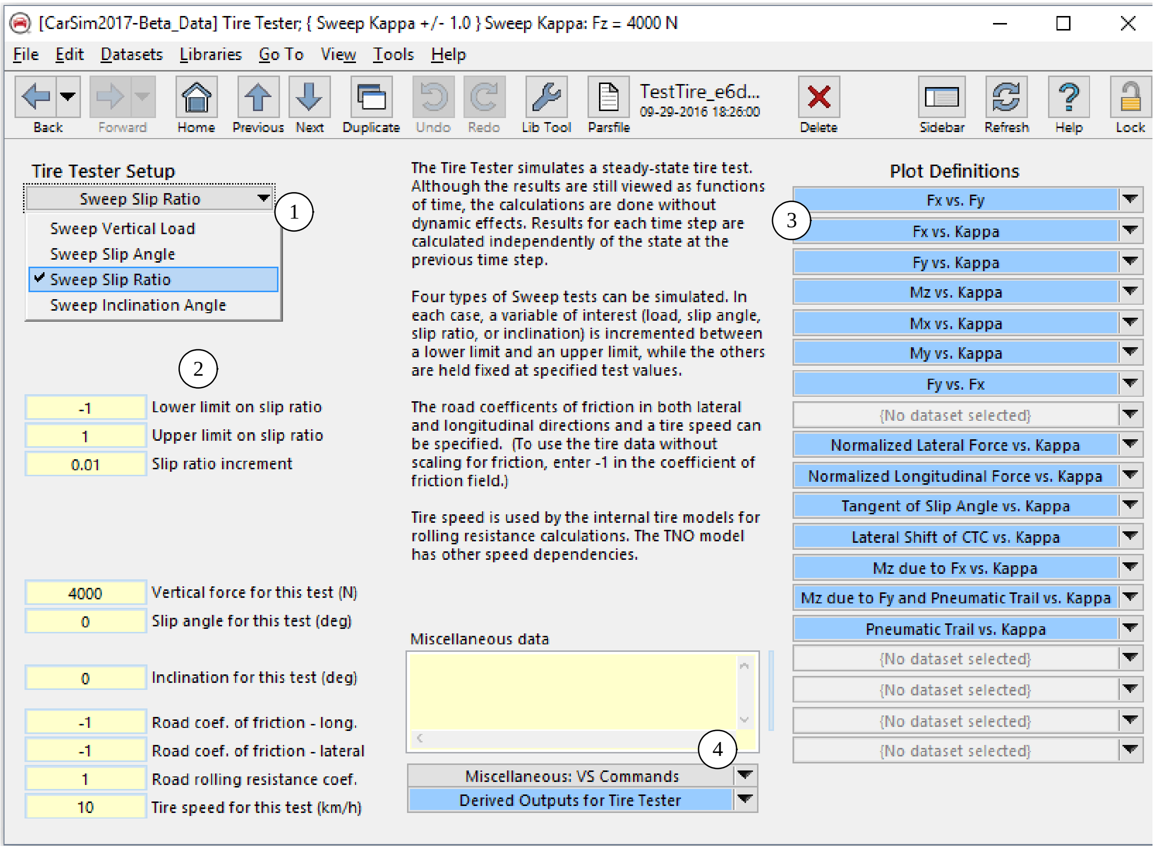



*Figure 22. The tire tester screen.*

3 Plot links are available to select plotting of output variables. Six forces and moments are calculated. The tire tester operates as if it is installed at the left front of a vehicle without duals, so the force and moment output variables are named Fx_L1, Fy_L1, Fz_L1, Mx_L1, My_L1, and Mz_L1. The input variables are also written to output.

4 Miscellaneous data field and library link. This field and link have no specific purpose. They can be used to add VS commands to define new output variables based on the calculated outputs.

# Tire Force and Moment Calculations

The interaction of each tire with the ground is defined by a longitudinal force ($F_X$), a lateral force ($F_Y$), a vertical force ($F_Z$), an overturning moment ($M_X$), ), a rolling resistance moment ($M_{yRR}$), and an aligning moment ($M_Z$). $F_X$, $F_Y$, $F_Z$, $M_X$, $M_{yRR}$, and $M_Z$ are applied to the wheels and reacted by the ground at the contact center point shown in Figure 23.

The figure shows three axis systems. They are defined based on two points (the wheel center and the contact center) and two directions (the ground normal at the contact center $\mathbf{g}_Z$ and the spin axis of the wheel $\mathbf{w}_Y$).

1. The tire/ground axis system has X and Y axes that lie in the plane of the road as defined at the contact center. The Z axis direction $\mathbf{g}_Z$ is normal to the ground surface at the contact center.

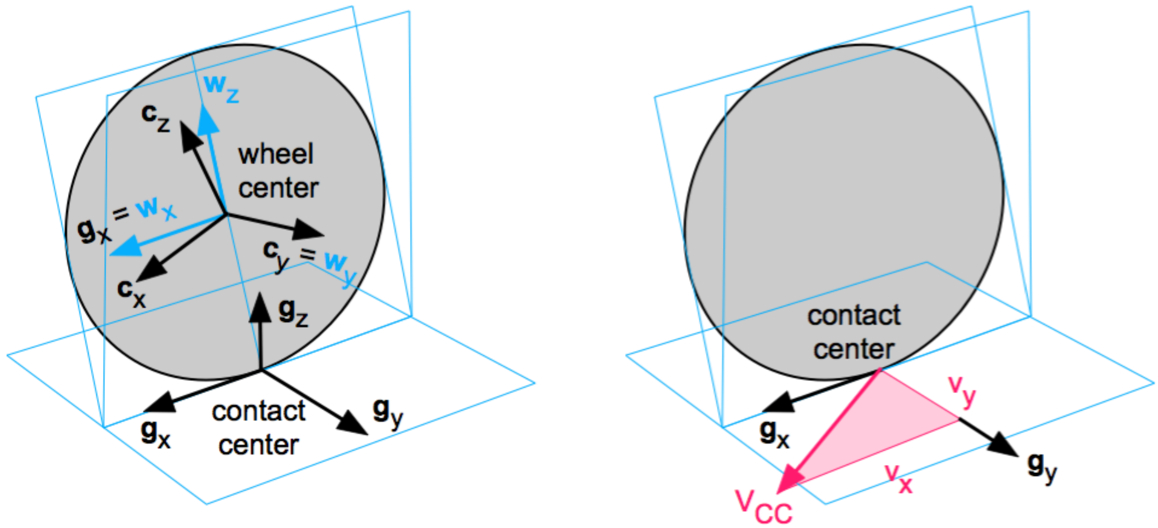



*Figure 23. Tire points and axes.*

2. The wheel axis system has a Y axis in the direction of wheel spin $\mathbf{w}_Y$, and a Z axis in the direction that connects the contact center to the wheel center. The X axis for the wheel is parallel to that of the ground at the contact center: $\mathbf{w}_X = \mathbf{g}_X$.

3. The steered (but non-rolling) wheel carrier axis system has the same Y axis as the wheel ($\mathbf{w}_Y = \mathbf{c}_Y$), but can pitch relative to the wheel axis system based on its connection to the vehicle sprung mass through the suspension and steering system kinematics.

**Notes** The axis directions of the ground contact $\mathbf{g}_X$, $\mathbf{g}_Y$, and $\mathbf{g}_Z$ are expressed in the vehicle dynamics standards SAE J670 and ISO 8855 as $\mathbf{X}_G$, $\mathbf{Y}_G$, and $\mathbf{Z}_G$. The axis directions of the wheel $\mathbf{w}_X$, $\mathbf{w}_Y$, and $\mathbf{w}_Z$ are expressed in the standards as $\mathbf{X}_W$, $\mathbf{Y}_W$, and $\mathbf{Z}_W$.

The contact center is also known as the center of tire contact in J670. The letters "CTC" have always been used in VehicleSim products in the names of some output variables associated with the center of tire contact.

Most tire models calculate the tire forces and moments based on a few kinematical variables: lateral slip angle $\alpha$, longitudinal slip ratio $\kappa$, inclination angle $\gamma$, and tire vertical deflection defined by the spatial relationship between the wheel center and contact center.

The tire models built into VS Math Models use a detailed static model that calculates tire forces and moments form the kinematical variables under steady-state conditions. The static model is extended to include transient behavior as described later (page 36).

## Tire/Ground Axes

The tire/ground X and Y axis directions, $\mathbf{g}_X$ and $\mathbf{g}_Y$, lie in the plane of the ground as defined at the contact center. $\mathbf{g}_x$ is defined as being perpendicular to the wheel spin axis $\mathbf{w}_Y$ which is also the carriage Y axis $\mathbf{c}_Y$ (from the multibody vehicle model). Thus,

$$\mathbf{g}_x = \frac{\mathbf{c}_y \times \mathbf{g}_z}{\left|\mathbf{c}_y \times \mathbf{g}_z\right|} \qquad \mathbf{g}_y = \mathbf{g}_z \times \mathbf{g}_x \tag{1}$$

## Lateral Slip

The slip angle ($\alpha$) for each tire is defined in terms of the X and Y velocity of $W_C$, $V_X$ and $V_Y$, expressed in the ground plane:

$$v_x = \mathbf{g}_x \bullet \mathbf{v}^{CC} \qquad v_y = \mathbf{g}_y \bullet \mathbf{v}^{CC} \tag{2}$$

where $\mathbf{v}^{CC}$ is the velocity vector for the contact center (Figure 23).

The slip angle is the arc tangent of the ratio:

$$\alpha = \tan^{-1} (v_y / |v_x|) \tag{3}$$

> **Note** Vehicle dynamics standards such as SAE J670 and ISO 8855 define lateral slip angle for the tire axis system, which has a tire X axis that always points in the direction of motion. In VS Math Models, the tire X axis ($\mathbf{g}_x$ in Figure 23) always points forward, even if the wheel center is moving backward. For this reason, the absolute value of $v_x$ is used in the denominator in equation 3; on the other hand, SAE J670 and ISO 8855 define longitudinal slip ratio with $v_x$ in the denominator (not the absolute value of $v_x$.)

## Longitudinal Slip

Longitudinal slip ratio ($\kappa$) is defined in VS Math Models as

$$\kappa = \frac{\omega R - v_x}{|v_x|} \tag{4}$$

where $\omega$ is the wheel-spin velocity of the wheel, R is the effective rolling radius, and $v_x$ is the forward speed of the wheel center.

> **Note** Vehicle dynamics standards SAE J670 and ISO 8855 define the longitudinal slip ratio with $v_x$ in the denominator (not the absolute value of $v_x$). As noted earlier, the direction $\mathbf{g}_x$ always points forward in VS Math Models; the absolute value is used to define lateral slip such that the resulting forces and moments are applied in the correct directions on the vehicle.

## Pure Longitudinal and Lateral Slip

$F_X$, $F_Y$ and $M_Z$ are calculated with VS Configurable Functions of two independent variables. $F_X$ is modeled as a function of $F_Z$ and longitudinal slip ($\kappa$):

$$F_X = FX(F_Z, \kappa) \qquad \{ \alpha = 0\} \tag{5}$$

$F_Y$ and $M_Z$ are modeled as table functions of $F_Z$ and lateral slip angle ($\alpha$):

$$F_Y = FY(F_Z, \alpha) \qquad M_Z = MZ(F_Z, \alpha) \qquad \{ \kappa = 0\} \tag{6}$$

## Sign Conventions

For the above definitions of longitudinal and lateral slip,

positive $\kappa$ generates positive $F_X$,

positive $\alpha$ generates negative $F_Y$, and

positive (small) $\alpha$ generates positive $M_Z$.

In VS Math Models, the X axis always points towards the front of the vehicle and the Y axis always points to the vehicle left. Therefore, the signs of $F_X$ and $M_Z$ are reversed when the wheel speed is negative. The sign of $F_y$ is not reversed due to the use of $|v_X|$ in the definition of $\tan(\alpha)$.

## Friction Similarity

Tire data are measured to fill out tables to define the functions shown in Equations 5 and 6. When measured over a full range of slip conditions, the resulting shear forces and aligning moment depend on the friction limits between the tire and test surface. On the other hand, the linear behavior for small amounts of slip is essentially unaffected by the friction limits, because the forces are caused solely by deformation of the tire structure.

In order to simulate the tire behavior on different surfaces, a method called *similarity* is used [2]. It predicts the change in limit shear force, while maintaining the linear behavior for small amounts of slip. Given a friction coefficient tire measurements, $\mu_O$, and the friction coefficient for the simulated condition, $\mu$, the table functions defined in Equations 5 and 6 are modified as follows:

$$F_X = \frac{\mu}{\mu_o} FX(F_Z, \frac{\mu_o}{\mu}\kappa) \qquad \{\text{For } \alpha = 0\} \tag{7}$$

$F_Y$ and $M_Z$ are modeled as table functions of $F_Z$ and lateral slip angle ($\alpha$):

$$F_Y = \frac{\mu}{\mu_o} FY(F_Z, \frac{\mu_o}{\mu}\alpha) \qquad M_Z = \frac{\mu}{\mu_o} MZ(F_Z, \frac{\mu_o}{\mu}\alpha) \qquad \{\text{For } \kappa = 0\} \tag{8}$$

In VS Math Models, the similarity method is integrated within the combined slip theory described below.

## Asymmetric Characteristics (Anisotropic properties)

The original *Combined Slip Theory* assumed symmetric tire behavior such that tabular data must goes through the origin (i.e. zero slip causes zero force). However, real tires are inherently asymmetric due to rolling resistance, ply steer and conicity, and typical tabular data for Fx and Fy from the tire measurements have side offset $\delta\kappa$ and $\delta\alpha$, respectively, as shown in Figure 24.

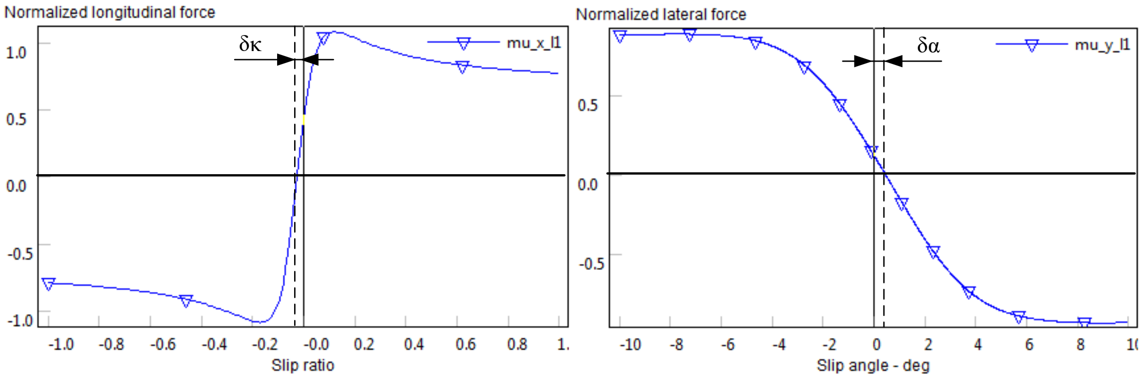



*Figure 24. Shifted longitudinal and lateral forces due to rolling resistance and conicity/ply steer, respectively (exaggerated).*

In order to make the theory work with asymmetry, Bakker et al. suggested that the tabular data is adjusted by the side offset (adjust the characteristic curves pass through the origin - they are called as the *base curves*), and the slip ratio and slip angle are compensated with $\delta\kappa$ and $\delta\alpha$, respectively [1]. This approach is used in the VS Tire model, and is included in the following equations where applicable.

## Camber Effect (Simple)

In the internal model with simple camber, the small effect of inclination is handled by modifying the slip angle using camber thrust:

$$\alpha_{eff} = \tan^{-1}(\tau) + \gamma \bullet R_\gamma \qquad R_\gamma = \left|\frac{K_\gamma}{K_\alpha}\right| \tag{9}$$

where $\tau$ is the tangent of the slip angle $\alpha$, $K_\gamma$ is a linear camber-thrust coefficient and $K_\alpha$ is the linear cornering stiffness. Both coefficients are evaluated at the instant vertical load $F_Z$.

## Combined Slip Theory

The longitudinal and lateral slips are combined to get total theoretical (combined) slip using a method described by Bakker, Pacejka, and Lidner [1]:

$$\sigma_{total} = \sqrt{\sigma_X^2 + \sigma_Y^2} \tag{10}$$

where

$$\sigma_X = -\frac{\kappa}{1+\kappa} + \frac{\delta\kappa}{1+\delta\kappa}, \qquad \sigma_Y = \frac{\tan(\alpha_{eff})}{1+\kappa} - \tan(\delta\alpha) \tag{11}$$

$\delta\kappa$ and $\delta\alpha$ are the side offset of tabular data for Fx and Fy, respectively (see Figure 24). The theoretical longitudinal slip $\sigma_x$ becomes equal to zero for $\kappa=\delta\kappa$ while the theoretical lateral slip $\sigma_Y$ will be zero for $\alpha_{eff}=\delta\alpha$ ($\kappa=0$ for pure side slip condition).

The tables of Equations 5 and 6 are adjusted for each offset in order that the characteristics go through the origin. The relation between the original and adjusted functions $FX_0$, $FY_0$ and $MZ_0$ are:

$$FX(\kappa+\delta\kappa, F_Z) = FX_0(\kappa, F_Z)$$

$$FY(\alpha+\delta\alpha, F_Z) = FY_0(\alpha, F_Z) \qquad MZ(\alpha+\delta\alpha_{mz}, F_Z) = MZ_0(\alpha, F_Z) \tag{12}$$

The theoretical slips are then normalized by peak slip values, $\sigma_{Xmax}$ and $\sigma_{Ymax}$, that cause peak or negative peak in the adjusted functions $FX_0$ and $FY_0$. For example, when the longitudinal slip is bigger than $\delta\kappa$, $\sigma_{Xmax}$ is calculated by the slip which causes the positive peak of $FX_0$ while the peak value is calculated by the absolute slip with the negative peak of $FX_0$ when the longitudinal slip is less than $\delta\kappa$. They are functions of $F_z$ and are obtained by a search algorithm. The total normalized slip is:

$$\sigma^*_{\text{total}} = \sqrt{\sigma^{*2}_{\text{X}} + \sigma^{*2}_{Y}} \tag{13}$$

where

$$\sigma^*_{\text{X}} = \frac{\sigma_{\text{X}}}{\sigma_{\text{Xmax}}} \qquad \sigma^*_{\text{Y}} = \frac{\sigma_{\text{Y}}}{\sigma_{\text{Ymax}}} \tag{14}$$

The equivalent longitudinal and lateral slips are calculated from the normalized total theoretical slip,

$$\kappa' = -\frac{\sigma^*_{total}\sigma_{Xmax}\,\text{sign}(\sigma_X)}{1+\sigma^*_{total}\sigma_{Xmax}\,\text{sign}(\sigma_X)}$$

$$\alpha' = \tan^{-1}[\sigma^*_{\text{total}}\sigma_{\text{Ymax}}\text{sign}(\sigma_{\text{Y}})] \tag{15}$$

Using the equivalent longitudinal and lateral slips, the so-called "base-curves" are obtained by means of the tabular data. Note that the normalized slip values are modified to include the friction ratio in support of the similarity method.

$$\begin{aligned} F_{X0} &= FX_0\left(F_Z, \frac{\mu_0}{\mu}\kappa'\right) \\ F_{Y0} &= FY_0\left(F_Z, \frac{\mu_0}{\mu}\alpha'\right) \end{aligned} \tag{16}$$

The base-curves are modified to account for the anisotropic properties of the tire-road friction.

$$F^*_{X0} = F_{X0} - \varepsilon(F_{X0} - F_{Y0})\left(\frac{\sigma^*_Y}{\sigma^*_{total}}\right)^2$$

$$F_{Y0}^{*} = F_{Y0} - \varepsilon(F_{Y0} - F_{X0})\left(\frac{\sigma_X^*}{\sigma_{total}^*}\right)^2 \tag{17}$$

where $\varepsilon = \sigma^*_{total}$ for $\sigma^*_{total} < 1$ and $\varepsilon = 1$ for $\sigma^*_{total} > 1$.

The longitudinal and lateral tire forces are then calculated by

$$F_X = F_{X0}^{*}\frac{\mu}{\mu_0}\cos(\lambda)sign(\sigma_X^*)$$

$$F_Y = -F_{Y0}^{*}\frac{\mu}{\mu_0}\sin(\lambda) \tag{18}$$

where

$$\lambda = \eta + \frac{2(\theta-\eta)}{\pi}\tan^{-1}(q_1{\sigma_{total}^*}^2)$$

$$\eta = \tan^{-1}\left(\frac{\sigma_Y^*}{|\sigma_X^*|}\right)$$

$$\theta = \tan^{-1}\left(\frac{\sigma_Y}{|\sigma_X|}\right) \tag{19}$$

λ defines the direction of the total horizontal force as shown in Figure 25. According to Bakker et al [1], $\sigma^*_{total}$ (denoted as $\sigma^*$ in Figure 25) should be used to determine the direction of the total horizontal force at small slip, while $\sigma_{total}$ (denoted as $\sigma_{tot}$ in the figure) should be used at large slip. The parameter ($q_1$) in equation 19 controls a smooth transition from the direction of the normalized slip to the total theoretical slip. If q1=0 (λ=η) the direction of the normalized slip is always applied, while if q1=∞ (λ=θ) the direction of the total theoretical slip is always applied. A $q_1$ value of 1.1 was used by Bakker et al [1]. Of course, the value of $q_1$ should be set to match the actual tire characteristics. If $q_1$ is too small, the model might be less accurate for large slip; if too large, the model might be less accurate for small slip (this can be seen by a dent in the $F_y$-$F_x$ friction ellipse).

**Notes** Equations 18 and 19 were added to the tire model in 2011 to more fully support the combined slip theory originally presented by Bakker et al [1]. Earlier versions used the direction of the total theoretical slip (λ=θ), as is now done by un-checking the box on the tire screen.

In the figure, ψ is an additional ply steer or conicity effect that is not used in CarSim or TruckSim.

The aligning moment is calculated by

$$M_Z = MZ_0(F_Z, \frac{\mu_0}{\mu}\alpha')\frac{\mu}{\mu_0}\sin(\lambda) \tag{20}$$

As described above, the longitudinal force ($F_x$), lateral force ($F_y$) and aligning moment ($M_z$) are derived by the combined slip theory in equations 10 to 20. In practice, the $F_x$, $F_y$ and $M_z$ values cannot be calculated in the same time, because that the shifted theoretical slip $\sigma_x$ and side slip $\sigma_Y$ (in equations 11) influence each other; thus, the forces and moment at pure slip condition cannot be reproduced.

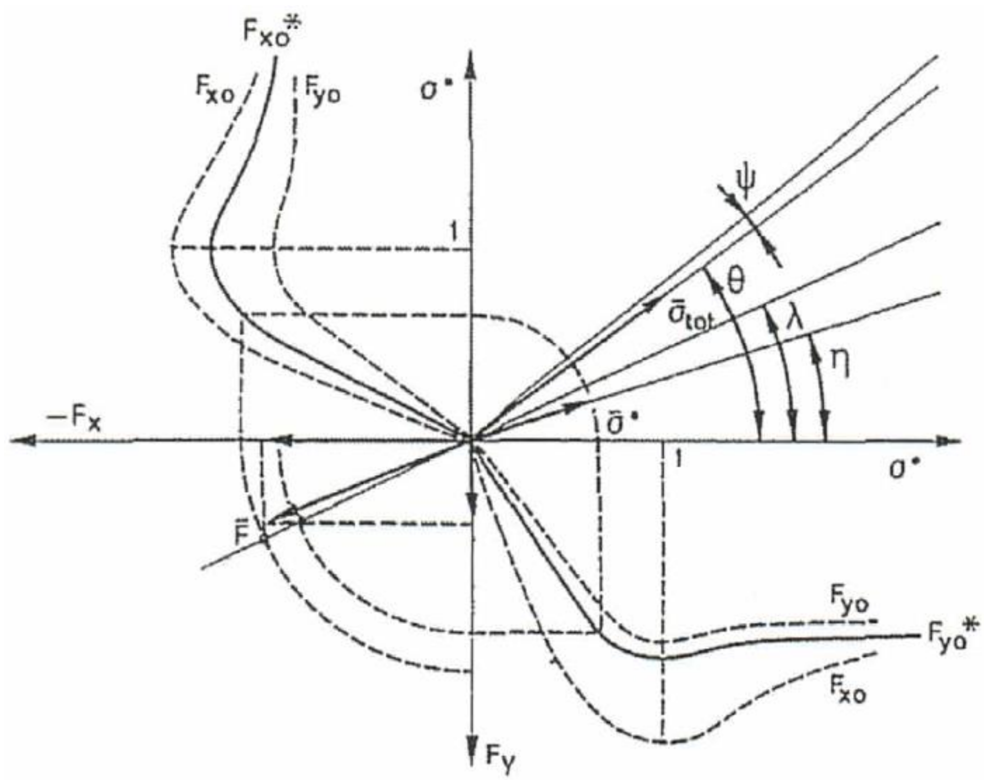



*Figure 25. Magnitude and direction of the total horizontal force using the normalized characteristics [1].*

In order to reproduce the pure slip condition, for example, the longitudinal force is calculated without side shifting the theoretical side slip ($\delta\alpha$ is set to zero in equation 11) while the lateral force is calculated without side shifting the theoretical longitudinal slip ($\delta\kappa$ is set to zero). Therefore, the combined slip calculation through equations 10 to 20 are repeated for each of $F_x$, $F_y$ and $M_z$ as shown in the flow chart of Figure 26.

## Extended Camber Effects

An extension to the basic tire model includes more effects of camber and the overturning moment $M_X$.

The added camber effects are supplied to the model as tables of lateral force, aligning moment, and overturning moment vs. inclination angle (at a zero slip angle and free rolling condition). Because they are defined this way, the camber effect is not included in the combined slip calculations for $F_Y$, $M_Z$, and $M_X$ due to slip. Therefore, the camber modification of slip angle (equation 9) adopted in the simple model is only applied for $F_X$ but not necessary for the other forces and moment, and the combined slip calculation (equation 11) uses pure slip angle, $\alpha$, instead of $\alpha_{eff}$ (otherwise, camber effects would be included twice). In this extension model, a force or moment component is added to the value from the combined slip equations *unadjusted for inclination* as follows:

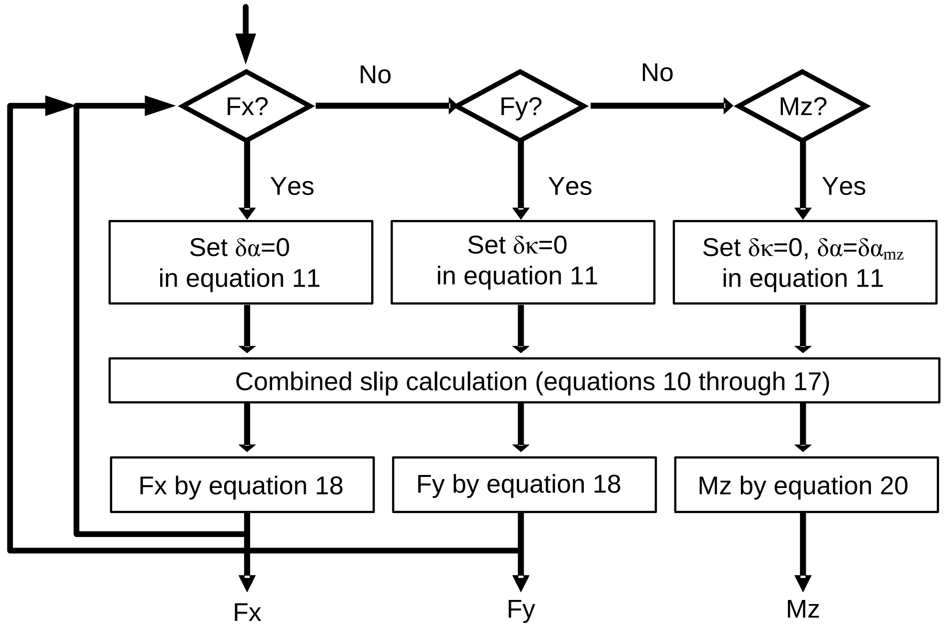



*Figure 26 Calculation flow chart for tire forces and moment with the combined slip.*

$$
\begin{aligned}
F_{YG} &= \frac{\mu}{\mu_0}\left\{ \mathrm{FYG}\left( F_Z , \frac{\mu_0}{\mu}\gamma \right) - \mathrm{FYG}\left(F_Z , 0\right) \right\} \cdot \frac{1}{1 + C_{RO}{\sigma^*_{total}}^2} \\
M_{XG} &= \frac{\mu_0}{\mu}\left\{ \mathrm{MXG}\left( F_Z , \frac{\mu}{\mu_0}\gamma \right) - \mathrm{MXG}\left(F_Z , 0\right) \right\} \cdot \frac{1}{1 + C_{RO}{\sigma^*_{total}}^2} \\
M_{ZG} &= \frac{\mu}{\mu_0}\left\{ \mathrm{MZG}\left(F_Z , \gamma\right) - \mathrm{MZG}\left(F_Z , 0\right) \right\} \cdot \frac{1}{1 + C_{RO}{\sigma^*_{total}}^2}
\end{aligned}
\qquad (21)
$$

where $F_{YG}$, $M_{XG}$, and $M_{ZG}$ are determined from the tables for the instantaneous values of Fz and γ. $C_{RO}$ is a parameter describing the rate at which each force or moment due to inclination diminishes with increasing total slip – the method has been suggested by Pacejka and Sharp [2]. The force and moments with γ=0 should be subtracted in order to avoid double counting the remained force and moments with α=0.

The value of the parameter $C_{RO}$ should be selected for each force and moment component to reproduce test data for combined slip and camber, if available. If combined test data is not available, a value of 1.0 is suggested for $F_y$, and 100.0 for $M_Z$. The overturning moment ($M_X$) due to camber is primarily a geometric effect, independent of slip, so a value of zero is suggested.

In equation 21, friction ratio ($\mu/\mu_0$) is used differently in each of force and moments: for lateral force ($F_{YG}$), the friction ratio is used as same way as described by previous sub-section, *Friction Similarity*; for aligning moment ($M_{ZG}$), the friction ratio is used as simply scaling down the moment; and for overturning moment ($M_{XG}$), the friction ratio is used inversely but the effect is compensated by modifying the side slip angle ($\alpha_{MXG}$) used by the table for the overturning moment due to slip angle described in the next sub-section.

Finally, $F_Y$, $M_X$, and $M_Z$ due to slip are added with $F_{YG}$, $M_{XG}$, and $M_{ZG}$, respectively.

## Overturning Moment

In the basic tire model, overturning moment $M_X$ is assumed to be zero. However, it is included in the extended model that includes detailed effects of camber.

Several mechanisms contribute to the development of an overturning moment. First, a lateral force causes a lateral distortion of the tire carcass. This distortion produces a lateral displacement of the effective center of tire contact. Second, inclination of the tire compresses one edge and unloads the other. This also causes a lateral displacement of the effective center of tire contact. The sum of these effects combines with the vertical load to produce an overturning moment.

Lateral force and inclination also may cause the belt of the tire to twist as it enters and leaves contact with the ground. This and other elastic effects may contribute to an overturning moment.

Although the tables for overturning moment with side slip and inclination are treated as independent, the saturation of overturning moment due to friction limit is treated as total effect by modifying the side slip angle ($\alpha_{MXG}$) such as:

$$\alpha_{MXG} = \tan^{-1}(\tau) - dM_{XG}/K_{MX\alpha} \qquad dM_{XG} = M_{XG} - MXG(F_Z, \gamma) \tag{22}$$

where $\tau$ is the tangent of the slip angle $\alpha$ and $K_{MX\alpha}$ is the partial derivative of $MX_0$ with respect to the side slip angle $\alpha$ (at $\alpha$=0).

The overturning moment due to side slip is calculated with modified side slip angle ($\alpha_{MXG}$) such as:

$$\begin{aligned} &M_X = MX_0(F_Z, \frac{\mu_0}{\mu}\alpha_{MXG}')\frac{\mu}{\mu_0}\sin(\lambda) \\ &\text{where} \\ &MX(F_Z, \alpha + \delta\alpha_{mx}) = MX_0(F_Z, \alpha), \end{aligned} \tag{23}$$

$\delta\alpha_{mx}$ is the side offset of tabular data for $M_X$ and $\alpha_{MXG}$' is calculated with $\alpha_{MXG}$ instead of $\alpha_{eff}$ by equations 10 through 15. The net overturning moment is the sum of the values of $M_X$ and $M_{XG}$.

Finally, the overturning moment and vertical force imply a net lateral displacement of the center of tire contact. This displacement and the longitudinal force $F_X$ contribute to the aligning moment $M_Z$.

**Notes** The data curve of $F_X$, $F_Y$, $M_Z$, and $M_X$ due to slip must cross the zero force or zero moment axis in order to specify the side offset $\delta\kappa$, $\delta\alpha$, $\delta\alpha_{mz}$ and $\delta\alpha_{mx}$, respectively. Otherwise, the calculation will stop with an error message.

As noted earlier, the tabular should include a column of zeros for a load Fz = 0. If not, the VS Solver will automatically add the column of zeros.

## Rolling Resistance and Power Balance

### *Rolling Resistance*

A free rolling tire (longitudinal slip ratio equal to zero), under load requires a force (at the wheel center) to roll it forward. This force is called *rolling resistance*. Although there are several mechanisms involved in the development of rolling resistance, it is thought to arise primarily from distortion of the loaded tire contact patch. When the loaded tire is rolled, the centroid of the distribution of vertical force moves forward of the geometric center of contact. The tire vertical load and the forward offset of the vertical tire force combine to contribute a moment in a direction that opposes rolling. As speed increase, the distortion may increase. The forward movement of the centroid of the vertical force is also influenced by factors due to the road surface such as the paving material and temperature.

The force due to rolling resistance is implemented as a moment that opposes rotation. It is nominally described by the equation:

$$M_{yRR} = F_Z \bullet R \bullet \texttt{Rr\_surf} \bullet (\texttt{Rr\_c} + \texttt{Rr\_v} \bullet v_X) \tag{24}$$

where $F_Z$ is the tire vertical load, R is the effective rolling radius, `Rr_surf` is a parameter describing the effect of the road surface, `rr_c` is the constant component of the rolling resistance coefficient, and `rr_v` is the speed varying component.

### *Power Balance*

The effective rolling radius `RRE` is usually described simply as the radius that relates the tire rotational speed to translational speed with little further discussion. It is observed by calculation from measured speeds, and is found to be less than the free radius but greater than the loaded radius, and it varies little over a wide range of vertical load.

A tire under vertical load assumes a flattened shape in the contact area, and of course, the loaded radius is less than the free radius. One might conclude that the loaded radius would be the effective radius relating translational and rotational speeds, but observations prove otherwise. Consider that, although the radius of the tire is reduced under load, the tire is no longer circular, and the length of the belt of the tire (its circumference) is essentially unchanged. During each revolution of the tire, the entire belt of the tire must pass through the contact patch. The result is an effective rolling radius that remains nearly constant over a wide range of load.

Tire forces occur at the interface between the tire and ground (at the loaded radius). They must to produce the correct suspension loading including roll, load transfer, and pitch effects, among many others. The drive torque might be determined from the longitudinal force and the loaded radius, or from the longitudinal force and the effective rolling radius. Using the loaded radius (rather than effective rolling radius) to determine torque results in a power unbalance (the torque is based on a smaller radius than the speed). However, the forces are physically produced at the loaded radius, and must be to interact properly with other vehicle properties.

To avoid a power unbalance, a moment due to Fx is calculated and added to the rolling resistance moment.

$$M_{yRR\_Fx} = Fx \bullet (R - \mathit{loaded\_radius}) \tag{25}$$

Where *loaded_radius* is determined from the unloaded radius, the vertical load, and the tire stiffness.

The sign is chosen so that it always opposes the direction of $F_X$, while rolling resistance always opposes the direction of rotation. This means, for example, it has the effect of increasing stopping distance. The power unbalance resulting from ignoring this amounts to a few percent, but can be important to many applications. On the other hand, you may wish to use only the effective rolling radius to describe the tire, so the capability to disable this calculation is provided on the **Tire** screen.

## Tire Transient Behavior (Lags and Relaxation Length)

Tires develop shear forces in response to deformation of the tire structure. The forces do not develop instantaneously, but build as the tire rolls [4, 5]. For example, Figure 27 shows how $F_Y$ builds in response to a step change in slip angle.

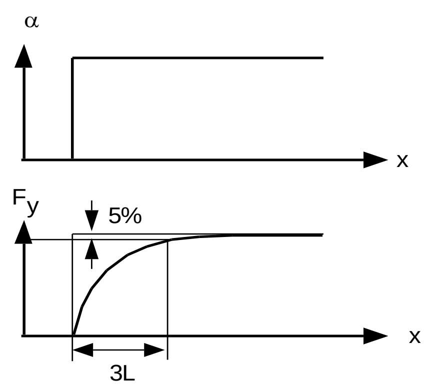



*Figure 27. Tire relaxation.*

### *Lagged Slip in VS Tire Models*

Two methods are commonly used for including the tire lag in a vehicle model:

1. A tire model with the dynamics built in, or
2. A static (steady-state) tire model with a separate filter to account for the lag.

The second approach is used in VS Math Models because it offers two practical advantages. First, it allows the use of any static tire model from the literature independently of the method used to introduce lag. Second, it simplifies the calculation of the kinematical variables used as inputs to the tire model. Lag is introduced into the slip angle such that the instantaneous response calculated for the lagged slip angle yields the lagged side force and aligning moment.

A method described by Bernard [5] is used to account for the lateral lag in tire response. In this method, a lagged slip angle, $\alpha_L$ is defined as the arc tangent of a state variable, $\tau$.

$$\alpha_L = \tan^{-1}(\tau) \tag{26}$$

The state variable $\tau$ is defined with the first-order differential equation:

$$\frac{d\tau}{dt} = \frac{|v_X|}{L_Y}(\tan(\alpha) - \tau)$$

$$\frac{d\tau}{dt} = \frac{|v_X|}{L_Y}\left(\frac{v_Y}{|v_X|} - \tau\right)$$

$$\frac{d\tau}{dt} = \frac{v_Y - \tau|v_X|}{L_Y} \qquad (27)$$

where $L_Y$ is the relaxation length for lateral slip.

The same approach is taken for defining a lagged longitudinal slip, $\kappa_L$, using a relaxation length $L_X$ for longitudinal slip. The lagged slip is then defined by the differential equation:

$$\frac{d\kappa_L}{dt} = \frac{|v_X|}{L_X}(\kappa - \kappa_L)$$

$$\frac{d\kappa_L}{dt} = \frac{\omega R - v_X - |v_X|\kappa_L}{L_X} \qquad (28)$$

In a VS Math Model, $\alpha_L$ and $\kappa_L$ are both state variables that are normally calculated by integrating the differential equations 27 and 28. Thus, these state variables have a dynamic lag similar to what is measured in physical tire tests. The static model (equations 5 - 25) is used with the replacement of the lagged slips for the instant slips (equations 3 and 4) that would be used for a static model.

### *Rebound and Overshoot Due to Lateral Relaxation Length*

The concept of dynamic lag based on forward travel assumes the tire is mainly moving forward. It leads to unrealistic behavior in situations where this is not true. For example, if there is significant slip angle and $v_X$ approaches zero, then the time constant ($L/v_X$) becomes very large, with the result that the lagged lateral slip angle follows the instant slip angle slowly, if at all. The simulated vehicle will come to zero lateral speed, and then "rebound" as if there is a big rubber spring between the tire and ground, slowly pulling the wheel back. The "rebound" or "overshoot" effect is caused because the lagged slip generates an unrealistic lagged side force.

Rill [6] and van der Jagt [7] describe relaxation data that vary with tire load and slip. Under large slip conditions, the lag is greatly reduced. These additions have been found to improve vehicle motion and stability under some conditions, such as sliding to a stop at very large slip angles. Relaxation lengths (both lateral and longitudinal) are defined internally using Configurable Functions of load ($F_Z$) and lagged slip (lateral or longitudinal) to provide sensitivity to operating conditions. When tabular data are used to reduce lateral relaxation length for high slip angles, the rebound/overshoot effect described above can be reduced to a negligible amount.

### *Overshoot Due to Longitudinal Relaxation Length*

There is a similar overshoot complication involving the interaction between slip and generated shear force that applies in the longitudinal direction but not in the lateral direction. There is a much tighter coupling between the generation of the force $F_X$ (via $\kappa_L$) and the application of the force to the wheel angular velocity $\omega$, which in turn defines the lagged slip $\kappa$. For typical tire properties and tire/wheel spin inertia values, there can be a high-frequency dynamic interaction.

At some speeds, the lagged slip can interact with the wheel spin inertia and acceleration to store energy in a manner that is not seen in physical test results. When there is high wheel-spin acceleration, as occurs when rapidly cycling the brakes under the control of ABS, release of the applied torque allows the wheel speed to not only return to a wheel-spin velocity close to the free rolling case, but in some cases, to overshoot. In this case, "overshoot" means that when the brake is released, the wheel spins up to a higher angular spin velocity than the free-rolling reference. It is as if the tire stored energy during the braking, and then put that energy into the spinning wheel.

In order to avoid this behavior, the VS Math Model supports an option to use the full lagged slip $\kappa_L$ only when $|\kappa|$ is increasing; when $|\kappa|$ is decreasing (defined as the case when $|\kappa| < |\kappa_L|$ and both $\kappa$ and $\kappa_L$ have the same sign) the model uses reduced lag. This option is selected by setting the parameter `OPT_TIRE_LAG_KAPPA` = 2. (This is also the default value for the parameter.)

### *Speed Thresholds for Tire Transient Behavior*

The tire models built into VS Math Models use the standard concepts of lateral and longitudinal relaxation length to provide realistic transient behavior at ranging from about 15 km/h and up using the equations proposed by Bernard and Clover [5] (see equations 27 and 28, page 37). When large ranges in slip and load are encountered, relaxation lengths should be defined as Configurable Functions of slip and load [6, 7]. However, will be described in the following subsections, the basic equations lead to some unrealistic behavior at speed conditions outside the range originally considered for the models. Especially in the cases of running real-time driving simulators that can cover any conceivable speed, driving behavior, or surface condition, modifications to the equations are provided to maintain feasible behavior.

Table 2 lists the parameters that were mentioned in this section on Tire Transient Behavior.

*Table 2. Tire parameters linking transient response to speed and special conditions.*

| Parameter | Default | Description |
|---|---|---|
| `OPT_TIRE_LAG_ALPHA` | 1 | Specifies how lag is handled at high speeds when the relaxation length defines a frequency that is too high |
| `OPT_TIRE_LAG_KAPPA` | 2 | |
| `VLOW_ALPHA` | 2 km/h | Minimum speed used in ODE for lagged slip, except at very low speeds where "drift" might occur |
| `VLOW_KAPPA` | 2 km/h | |
| `VLOW_DRIFT_X` | 0.1 km/h | Speed where vehicle can be considered "at rest" such that ODEs can allow lagged slip to prevent "drift" |
| `VLOW_DRIFT_Y` | 0.1 km/h | |
| `VLOW_DAMP_Y` | 0.5 km/h | Speed where lateral damping force is added to avoid oscillation that sometimes occurs when stopping |

### *Avoiding Lateral Tire Harshness at Very Low Speed*

When the vehicle is at rest, or moving at a very low speed, then a tire model for vehicle dynamics is not representing the behavior of the tire as structural element connecting the wheel the ground. In the case of lateral slip and force, a tire model without a transient lag in the lateral response acts as a very stiff lateral damper. Because the forward speed $v_X$ is zero, any perturbation in lateral velocity $v_Y$ generates a slip angle of ±90°, with maximum lateral force $F_Y$. The vehicle model is subject to a "bang-bang" oscillation.

VS Tire Models avoid this type of oscillation by three means:

1. The relaxation length differential equation (27) acts as a low-pass filter that reduces high-frequency behavior.
2. A low-speed limit is used in place of $v_X$ in the equations for $\alpha$ and $\alpha_L$ (if used), making the tire less sensitive at very low speeds.
3. A lateral force is added to the tire $F_Y$ opposing the instant $v_Y$ component of the wheel motions, to damp potential oscillations.

When $v_X$ and $v_Y$ are both very small, the response of the vehicle feels "harsh" when run in real-time driving simulators. Hence, the core definition of $\alpha$ is modified to include a low limiting speed:

$$\alpha = \tan^{-1}\left(\frac{v_y}{max(|v_X|, v_{low\alpha})}\right) \tag{29}$$

where $v_{low\alpha}$ is the low-speed limit, defined in the VS Math Model with the tire parameter VLOW_ALPHA. The differential equation for lagged slip that includes $v_{low\alpha}$ is:

$$\frac{d\tau}{dt} = \frac{v_Y - \tau\, max(|v_X|, v_{low\alpha})}{L_Y} \tag{30}$$

When coming to a stop in a driving simulator, there can be a lateral oscillation a near-zero speed. To prevent this, a lateral damping force is added to the lateral force produced by the steady-state tire model. The damping force is present only when the speed drops below a limit specified by the parameter VLOW_DAMP_Y. If this parameter is set to zero, the low-speed damping is disabled.

### *Avoiding Longitudinal Tire Instability at Very Low Speed*

The equation that defines longitudinal slip ratio (4) is singular when speed $v_X$ is zero, and leads to unstable tire behavior for very low values of $v_X$. To avoid this problem, a low-speed limit is included:

$$\kappa = \frac{\omega R - v_X}{max(|v_X|, v_{low\alpha})} \tag{31}$$

where $v_{low\kappa}$ is a low-speed limit, defined in the VS Math Model with the tire parameter VLOW_KAPPA. The differential equation for lagged slip ratio that includes $v_{low\kappa}$ is:

$$\frac{d\kappa_L}{dt} = \frac{\omega R - v_X - max(|v_X|, v_{low\alpha})\kappa_L}{L_X} \tag{32}$$

### *Preventing "Drift" for the Vehicle at Rest on a Grade*

One of the advantages of the original form of the lagged slip equations 27 and 28 mentioned by Bernard [5] is that they produce a static lagged slip at zero speed that keeps the vehicle from "drifting" when stopping on a grade. This behavior is not important in many vehicle simulation scenarios, but is helpful when used in driving simulators to provide confidence to drivers that the vehicle model is trustworthy. Unfortunately, the parameters $v_{low\alpha}$ (VLOW_ALPHA) and $v_{low\kappa}$ (VLOW_KAPPA), introduced in the previous subsections to reduce perceived "harshness" of the vehicle at very low speeds, prevent the lagged slips from reaching static values needed to avoid a very low drift (v < 0. 1 km/h).

To restore this capability, two vehicle parameters are included: VLOW_DRIFT_X and VLOW_DRIFT_Y to define extremely low speed limits. These are used to set VLOW_KAPPA and VLOW_ALPHA to zero, when the vehicle is considered “almost at rest.”

For the case of longitudinal movement, VLOW_KAPPA is temporarily set to zero if three conditions are satisfied:

1. $|v_x|$ < VLOW_DRIFT_X, and
2. $|v_y|$ < VLOW_DRIFT_X, and
3. $|\omega R|$ < VLOW_DRIFT_X

For the case of lateral movement, VLOW_ALPHA is temporarily set to zero if two conditions are satisfied:

1. $|v_x|$ < VLOW_DRIFT_Y, and
2. $|v_y|$ < VLOW_DRIFT_Y

The parameters VLOW_DRIFT_X and VLOW_DRIFT_Y both have default values of 0.1 km/h. They can be changed if needed using any miscellaneous yellow fields in the CarSim or TruckSim browser.

In order to avoid a discontinuity when the vehicle speed goes into or out of the limits defined by the drift parameters, a transition is used. For longitudinal drift, the transition covers speeds between VLOW_DRIFT_X and 2•VLOW_DRIFT_X; for lateral drift, the transition covers speeds between VLOW_DRIFT_Y and 2•VLOW_DRIFT_Y.

### *Limits at High Speed*

The numerical integration methods used in VS Solver programs to solve the equations of motion use a discrete time $\Delta T$ that must be much shorter than any response times in the math model. The response time associated with relaxation length is simply the length L divided by forward speed of the wheel center: $L/v_x$. As speed increases, the time constant associated with relaxation length diminishes, until the numerical integration method becomes unstable because $\Delta T$ is not small enough for the assumptions underlying the calculation method.

The problem of numerical instability is more serious with longitudinal relaxation length $L_x$, which is typically 50 mm or less. It has been found that the differential equations are well behaved if the time constant is greater than about 12 times the smallest time step $\Delta T$ used in the VS Math Model. For most of the numerical integration methods, the differential equations are applied twice each major time step, at a half of the parameter TSTEP. For example, if TSTEP is set to 0.001 s, the half step is 0.0005 s, and the time constant should not be less than 12•0.0005 = 0.006 s. If $L_x$ is 50 mm (0.05 m), then the limit is reached when $v_x$ reaches $L_x/(12\ \Delta T)$ = (0.05 m)/(0.006 s) = 8.3333 m/s => 30 km/h. For higher speeds, the tire transient response is so quick that the dynamics are outside the frequency range of the VS Math Model.

Although it is less common, lateral tire dynamic lag also becomes negligible at higher speed (e.g. 150 km/h), or at lower speeds if the relaxation length is small.

Two methods are available in VS Math Models to avoid numerical instabilities at higher speeds. One is to replace values of the lagged slip variables with instant variables (equations 3 and 4) when the relaxation length divided by $|v_X|$ approaches the limit of 12 $\Delta$T. The second method is to increase the relaxation length such that the time constant never drops below the limit of 12 $\Delta$T. The first method is recommended; the second method is provided for purposes of backward compatibility.

The choice between these two methods is set with a parameter `OPT_TIRE_LAG_KAPPA` for longitudinal lag and a parameter `OPT_TIRE_LAG_ALPHA` for lateral lag. If one of these parameters is set to zero, the corresponding relaxation length is increased at high speeds. If set to a non-zero value, the lagged slips are replaced with instant slips at high speeds, with a transition going from fully lagged slip when the time constant would be 12 $\Delta$T, to instant slip when the speed is doubled (i.e., the time constant would be 6 $\Delta$T if it were not replaced with instant slip).

If a relaxation length is set to zero (either lateral or longitudinal), then the VS Solver skips the corresponding differential equation and simply assigns the instant slip value to the lagged slip used in the static tire model. Be aware that if instant slip is used at very low speeds, the wheel will probably be numerically unstable in spin.

# Tire Interaction with 3D Road Geometry

Tire models make use of a point called the contact center for the origin of a coordinate system where the tire contacts the ground. The general solution of the location of contact center on an arbitrarily shaped surface requires an iterative solution at each time step. This section describes the method used in VS Math Models.

Figure 28 shows how the wheel plane intersects with a ground plane that is not necessarily level. The wheel plane is assumed neither vertical nor perpendicular to the ground plane.

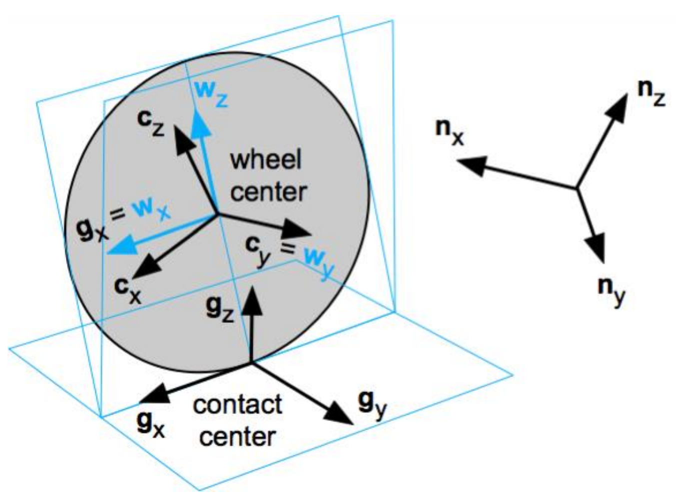



*Figure 28. Axis systems of the wheel carrier, the wheel, and the ground.*

The wheel carrier has a point that coincides with the wheel center and whose position is defined by global coordinates $x_{WC}$, $y_{WC}$, and $z_{WC}$. It has axis directions $\mathbf{c}_X$, $\mathbf{c}_Y$, and $\mathbf{c}_Z$. In the immediate area where the wheel plane intersects the ground, the ground has constant slopes in the global X and Y directions (dZ/dX and dZ/dY) that define a ground normal $\mathbf{g}_Z$:

$$\mathbf{g}_Z = \mathtt{dir}(\mathbf{n}_Z - (dZ/dX)\,\mathbf{n}_X - (dZ/dY)\,\mathbf{n}_Y) \quad (33)$$

where `dir` is a 3D vector function that sets the length of a vector to unity. The ground geometry is assumed to be defined such that the vertical coordinate $z_{gnd}$ and the two slopes dZ/dX and dZ/dY can be obtained as functions of the global coordinates $x_{gnd}$ and $y_{gnd}$. The inertial axis directions $\mathbf{n}_X$, $\mathbf{n}_Y$, and $\mathbf{n}_Z$ are used to define the ground normal, but are otherwise irrelevant to the geometry of the interaction between the wheel plane and the ground.

> **Note** This section uses the multibody convention of representing the tire/ground interface directions as $\mathbf{g}_X$, $\mathbf{g}_Y$, and $\mathbf{g}_Z$ and the wheel directions as $\mathbf{w}_X$, $\mathbf{w}_Y$, and $\mathbf{w}_Z$. These correspond to the directions expressed in the vehicle dynamics standards SAE J670 and ISO 8855 $\mathbf{X}_G$, $\mathbf{Y}_G$, and $\mathbf{Z}_G$, and $\mathbf{X}_W$, $\mathbf{Y}_W$, and $\mathbf{Z}_W$, respectively.

## Overview of Modeling Approach

The multibody vehicle model handles the describing equations for the rigid bodies, so at any instant of time all positions, angles, and associated velocities are known for all of the bodies in the system. Forces can be applied to any points defined on the bodies, in any directions. However, the calculations are reduced if the forces are applied on points that can be described easily and are fixed in the bodies. The calculations are also reduced if the directions of the applied forces and moments are in directions that have already been defined, such as the axis directions of the rigid bodies upon which the forces act.

The modeling approach involves separating the details of the 3D tire/ground connection from the main multibody vehicle model. The multibody model provides the position of the wheel center and orientation of a rigid body containing the wheel spindle. The multibody model is also ready to accept six forces and moments applied to the spindle, with the spin moment $M_Y$ applied to the spinning wheel, and the other five forces and moments applied to the non-spinning body containing the spindle.

Several functions were developed with the VehicleSim Lisp multibody code generator to handle details for:

1. defining the tire contact center,
2. developing the kinematical inputs for a tire model, and
3. the transformation of the tire model forces and moments from the contact center to the wheel center.

These calculations take place in five steps, involving 3D body kinematics, an external source of information about the road geometry, and an external tire model that provides six forces and moments at the contact center based on kinematical inputs.

## Step 1: Estimate X and Y Coordinates of the Contact Center

At a given instant of time, the state variables of the VS Math Model give the location of the wheel center and the orientation of the steered wheel carrier. The vector from the contact center to the wheel center is $R \bullet \mathbf{w}_Z$ where

$$\mathbf{w}_Z = \texttt{dir(dplane(}\mathbf{t}_Z, \mathbf{w}_Y\texttt{))} \tag{34}$$

and `dplane` is a 3D vector function that projects the first vector ($\mathbf{t}_Z$) onto a plane perpendicular to the second vector (the wheel plane, defined as being perpendicular to $\mathbf{w}_Y$). Computation of the location of the contact center is direct using equations 33 and 34, if values are known for the three scalars: R, dZ/dX, and dZ/dY. Although they are not known initially for a new position of the wheel center, values from the previous time step can be used to estimate $\mathbf{t}_Z$, $\mathbf{w'}_Z$, and the global X and Y coordinates of the contact center:

$$x_{gnd} = x_W - R\, \mathbf{w}_Z \bullet \mathbf{n}_X$$

$$y_{gnd} = y_W - R\, \mathbf{w}_Z \bullet \mathbf{n}_Y \tag{35}$$

A function that takes as input the global coordinates of the wheel center and Euler angles for the wheel carrier, along with old values for R, dZ/dX, and dZ/dY computes values of $x_{gnd}$ and $y_{gnd}$ based on that information.

## Step 2: Update Ground Variables Z, dZ/dX, dZ/dY

The values of $x_{gnd}$ and $y_{gnd}$ are used to obtain new values of $z_{gnd}$, dZ/dX, and dZ/dY from the road model along with possible imports from Simulink or other external code.

The new values of $z_{gnd}$, dZ/dX, and dZ/dY are assumed to define the correct road geometry. Although the values of $x_{gnd}$ and $y_{gnd}$ obtained do not exactly match the true contact center, they are located on a tilted plane in the near vicinity of the contact center.

Depending on the setting for the tire contact patch dimensions, the contact center properties ($z_{gnd}$, dZ/dX, and dZ/dY) may be based on a single point, two points, or four points (length and width). There are four cases covered in the math models:

1. If the dimensions are zero (contact is at a single point), then $z_{gnd}$, dZ/dX, and dZ/dY at the point of contact are used.
2. If the width is zero but the length is non-zero, then global X and Y coordinates are calculated for two points separated longitudinally by the instant length of the contact line, with the contact center at the midpoint. The elevations (Z) at the two points are averaged to obtain $Z_{gnd}$; the difference in two point elevations is divided by the instant length to obtain slope in the tire X direction; and the two lateral slopes (relative to the tire Y axis) are averaged to obtain lateral slope.
3. If the length is zero but the width is non-zero, then global X and Y coordinates are calculated for two points separated laterally by the instant width of the contact line, with the contact center at the midpoint. The elevations (Z) at the two points are averaged to obtain $Z_{gnd}$; the difference in two point elevations is divided by the instant width to obtain

slope in the tire Y direction; and the two longitudinal slopes (relative to the tire X axis) are averaged to obtain longitudinal slope.

4. If the length and width are both non-zero, then global X and Y coordinates are calculated for four points on a rectangle with the contact center at the center. The elevations (Z) at the four points are averaged to obtain $Z_{gnd}$; averages and differences of the point elevations are used with the dimensions to calculate the effective lateral and longitudinal slopes associated with the contact center.

## Step 3: Determine Kinematical Inputs for the Tire Model

The next step is to refine the location of the contact center and calculate the kinematical variables needed by any tire model that produces shear forces acting on the contact center.

In this stage, $\mathbf{g}_Z$ and $\mathbf{w}_Z$ are redefined using the updated values of dZ/dX and dZ/dY, and re-applying equations 33 and 34. Figure 29 shows how the distance R between the wheel center and the contact center is updated. In the figure, point 1 was defined using the old value of R and point 2 is the contact center based on the correct value. The value of $z_{gnd}$ associated with the new $x_{gnd}$ and $y_{gnd}$ values obtained in step 2 is shown in the figure directly below point 1 by a vertical interval of DZ.

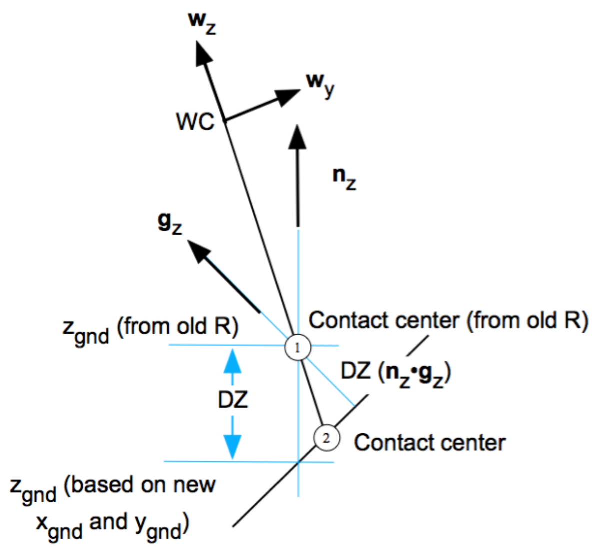



*Figure 29. Front view of wheel plane: adjusting R to improve contact center location.*

The height DZ in the $\mathbf{n}_Z$ direction can be projected into the ground normal $\mathbf{g}_Z$ to indicate a difference of DZ•($\mathbf{n}_Z$•$\mathbf{g}_Z$) in the $\mathbf{g}_Z$ direction. This distance can in turn be projected into the wheel plane using the inclination angle $\gamma$ between the ground normal and the wheel plane:

$$\gamma = \sin^{-1}(\mathbf{w}_Y \bullet \mathbf{g}_Z) \tag{36}$$

$$DR = DZ\ \mathbf{n}_Z \bullet \mathbf{g}_Z / \cos(\gamma) \tag{37}$$

$$R = R_{old} + DR \tag{38}$$

Tire forces $F_X$ and $F_Y$ are applied at the contact center in the directions $\mathbf{g}_X$ and $\mathbf{g}_Y$, defined as:

$$\mathbf{g}_X = \mathtt{dir}(\mathbf{w}_Y \times \mathbf{g}_Z) \tag{39}$$

$$\mathbf{g}_Y = \mathbf{g}_Z \times \mathbf{g}_X \tag{40}$$

Because $\mathbf{w}_Y$ and $\mathbf{g}_Z$ are not necessarily perpendicular, the 3D vector function `dir` is used to set the magnitude of $\mathbf{g}_X$ to unity. On the other hand, $\mathbf{g}_Z \times \mathbf{g}_X$ are perpendicular, so the `dir` function is not needed to define $\mathbf{g}_Y$.

Longitudinal speed of the contact center is the longitudinal speed of the wheel center plus the angular velocity crossed with the vector from the wheel center to the contact center:

$$v_{WCX} = (\mathbf{v}^{WC} - \boldsymbol{\omega}^{W} \times [R\ \mathbf{w}_Z]) \bullet \mathbf{g}_X \tag{41}$$

Lateral speed of the contact center $v_{CCY}$ is the lateral speed of the wheel center plus the angular velocity crossed with the vector from the wheel center to the contact center:

$$v_{CCY} = (\mathbf{v}^{WC} - \boldsymbol{\omega}^{W} \times [R\ \mathbf{w}_Z]) \bullet \mathbf{t}_Y \tag{42}$$

Any tire model that predicts forces and moments at the contact center will require as inputs the following:

1. the distance R between the wheel center and the contact center, as defined in equation 38,
2. the inclination $\gamma$, defined in equation 36,
3. lateral and longitudinal slips defined by velocities defined in equations 41 and 42, and
4. angular spin of the wheel, which is simply $\omega_Y$ of the non-spinning wheel carrier plus the wheel spin relative to the carrier.

The derivative of the radius R is also simple to define, being the component of the velocity of the wheel center in the wheel plane:

$$dR/dt = \mathbf{v}^{WC} \bullet \mathbf{w}_Z \tag{43}$$

The radius derivative can be used in some tire models to affect the vertical force. Also, it can be used in the multibody solver program to provide a more accurate estimate of R at the next time step.

The variables $x_{gnd}$, $y_{gnd}$, and $z_{gnd}$ are recalculated using the updated values of R with equation 35 and a similar equation for $z_{gnd}$:

$$z_{gnd} = z_W - R\,(\mathbf{w}_Z \bullet \mathbf{n}_Z) \tag{44}$$

## Step 4: Apply the Contact Center Tire Model

The kinematical variables produced in step 3 are passed as arguments to a tire model which in turn provides six forces and moments that are intended to be applied to the contact center using tire/ground axis directions $\mathbf{g}_X$, $\mathbf{g}_Y$, and $\mathbf{g}_Z$. The tire model used may be one that is built into the VS Solver, it may be provided by a third-party software company, or it may be a user-supplied function.

Details of the force and moment calculations were discussed in the previous section, **Tire Forces and Moments**.

## Step 5: Transform the Tire Forces and Moments

The six forces and moments provided by the tire model at the contact center are $F_x$, $F_y$, $F_z$, $M_x$, $M_y$, and $M_z$. The forces and moments desired at the wheel center are $F_{xwc}$, $F_{ywc}$, $F_{zwc}$, $M_{xwc}$, $M_{ywc}$, $M_{zwc}$. To define the transform, the scalar components are combined into vectors:

$$\mathbf{F} = F_x\,\mathbf{g}_x + F_y\,\mathbf{g}_y + F_z\,\mathbf{g}_z \tag{45}$$

$$\mathbf{M}_{cc} = M_x\,\mathbf{g}_x + M_y\,\mathbf{g}_y + M_z\,\mathbf{g}_z \tag{46}$$

$$\mathbf{M}_{wc} = \mathbf{M}_{cc} - (R\,\mathbf{w}_z) \times \mathbf{F} \tag{47}$$

The force vector and wheel center moment vector are then projected into the axis directions of the non-spinning wheel carrier:

$$F_{xwc} = \mathbf{F} \bullet \mathbf{c}_x \qquad F_{ywc} = \mathbf{F} \bullet \mathbf{c}_y \qquad F_{zwc} = \mathbf{F} \bullet \mathbf{c}_z \tag{48}$$

$$M_{xwc} = \mathbf{M}_{wc} \bullet \mathbf{c}_x \qquad M_{ywc} = \mathbf{M}_{wc} \bullet \mathbf{c}_y \qquad M_{zwc} = \mathbf{M}_{wc} \bullet \mathbf{c}_z \tag{49}$$

A VS library function continues the calculations, taking the six contact-center forces and moments as input arguments and providing the six wheel-center forces and moments as outputs.

## Import and Output Variables for the Tire Forces and Moments

The tire forces and moments at the wheel center and contact center can be imported from or exported to the external models, namely Simulink, VS commands, etc. The import and output variables for the tire forces and moments at the wheel center and contact center are summarized in Table 3 and Table 4, respectively.

*Table 3. Summary of import and output variables for the wheel center forces and moments.*

| **Notation of this document** | **Root keyword** | | **Description** |
|---|---|---|---|
| | **Import** | **Output** | |
| $F_{xwc}$ | `IMP_FX0` | `Fx_WC` | Longitudinal force at the wheel center |
| $F_{ywc}$ | `IMP_FY0` | `Fy_WC` | Lateral force at the wheel center |
| $F_{zwc}$ | `IMP_FZ0` | `Fz_WC` | Vertical force at the wheel center |
| $M_{xwc}$ | `IMP_MX0` | `Mx_WC` | Roll moment at the wheel center |
| $M_{ywc}$ | `IMP_MY0` | `My_WC` | Spin moment at the wheel center |
| $M_{zwc}$ | `IMP_MZ0` | `Mz_WC` | Yaw moment at the wheel center |

*Table 4. Summary of import and output variables for the tire contact forces and moments.*

| **Notation of this document** | **Root keyword** | | **Description** |
|---|---|---|---|
| | **Import** | **Output** | |
| $F_X$ | IMP_FX | Fx | Longitudinal force at the contact center |
| $F_Y$ | IMP_FY | Fy | Lateral force at the contact center |
| $F_Z$ | IMP_FZ | Fz | Vertical force at the contact center |
| $M_X$ | IMP_MX | Mx | Overturning moment at the contact center |
| $M_Y$ | IMP_MY | MyRR | Rolling resistance at the contact center |
| $M_Z$ | IMP_MZ | Mz | Aligning moment at the contact center |

As described in this chapter, the tire forces and moments at the contact center are transformed to the wheel center. Therefore, the imported tire forces and moments at the contact center are also transformed together with the internal model values to the wheel center. However, the imported forces and moments at the wheel center are not transformed back to the contact center.

If the tire model is third-party model (the details are described in the next chapter), the third-party model calculates the forces and moments at the wheel center that are fed to VS vehicle model. In this case, the tire forces and moment at the contact center are internally calculated in the third-party model and they are fed to VS model for just plotting purpose (not affecting to the vehicle motion), and those import variables are ignored.

Table 5. summarizes the effects of the import and output variables for tire forces and moments.

*Table 5. Summary of effects of the import and output variables for the tire forces and moments.*

| **Tire model option** | **Wheel center (WC)** | | **Contact center (CTC)** | |
|---|---|---|---|---|
| | **Import** e.g. IMP_FX0 | **Output** e.g. Fx_WC | **Import** e.g. IMP_FX | **Output** e.g. Fx |
| Internal tire models (OPT_TIRE_MODEL < 8) | Importable | Internal model output with imported values at WC and CTC | Importable | Internal model output with imported value at CTC |
| 3rd-party tire models (OPT_TIRE_MODEL ≥ 8) | Importable | 3rd-party model output with imported value at WC | Ignored | 100% from 3rd-party model output |

# Connection to third-party tire models

The VS vehicle models support third-party tire models such as MF-Tyre/MF-Swift, COSIN FTire, etc. If you select one of the third-party tire model options, then the tire model built into the VS Solver is disabled for the selected tire(s), and the third-party software calculates all tire forces and moments.

## Overview

Figure 30 shows a diagram of how a VS vehicle model works together with a third-party tire model. The VS Solver simulates the entire vehicle except the wheel/tire (above the wheel axles) and the tire model simulates wheel/tire dynamics including tire/ground contact (below the wheel axles).

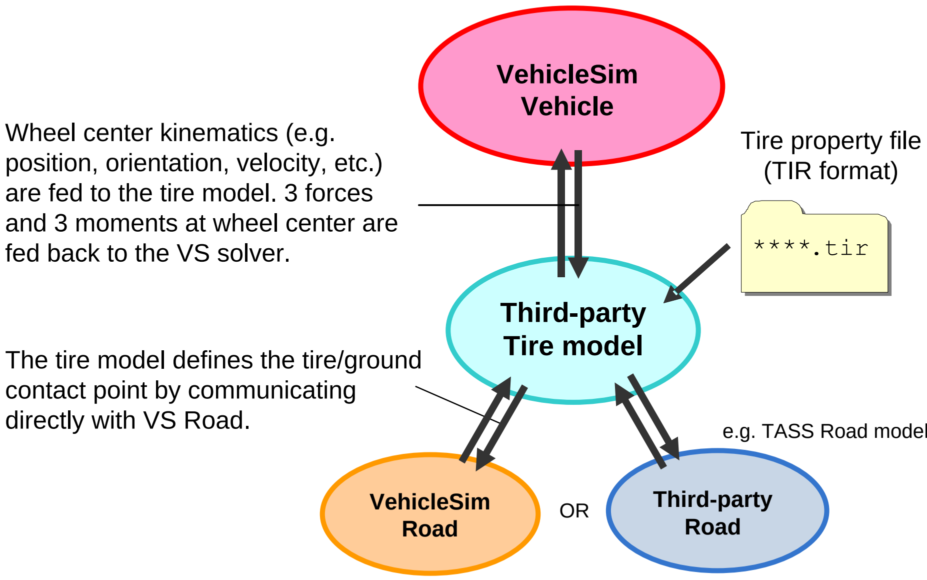



*Figure 30. Co-simulation with a VS Solver and third-party tire model.*

The VS Solver calculates kinematical variables at each wheel center such as position coordinates, velocity, orientation angle, angular speed, wheel rotation speed, and provides them to the tire model. The tire model calculates three forces and three moments at the wheel center that are sent back to the vehicle model.

The VS Solver provides road information as part of the VS API. As shown in Figure 30, third-party tire models can use the VS API road function to find the tire/ground contact point or contact surface. As an alternative to the VS road, you can also use the native road function within the third-party software.

### *VS STI Interface and Module*

The VS vehicle model and external tire models are implemented with separate program files that communicate each other through the *VS STI Interface* during the simulation. Figure 31 shows the program structure for interfacing between a VS Solver and external tire models, and corresponding program settings of **Tire (External)** screen.

The VS STI interface is a set of predetermined function prototypes written in C. A separate program with the VS STI interface is called a *VS STI Module*. The VS STI Module can either act as an interface between the VS Solver and an external tire model, or it can include an external tire model. Some sample programs and Microsoft Visual Studio projects are provided with CarSim and TruckSim to help users build custom VS STI modules. VS STI module can be compiled for a Windows DLL, dSPACE DS1006, SCALEXIO, or for Concurrent RT library file. More details about the VS STI interface is described in the separate document, *Connection to Third-party Tire Models using VS STI Interface and Module*.

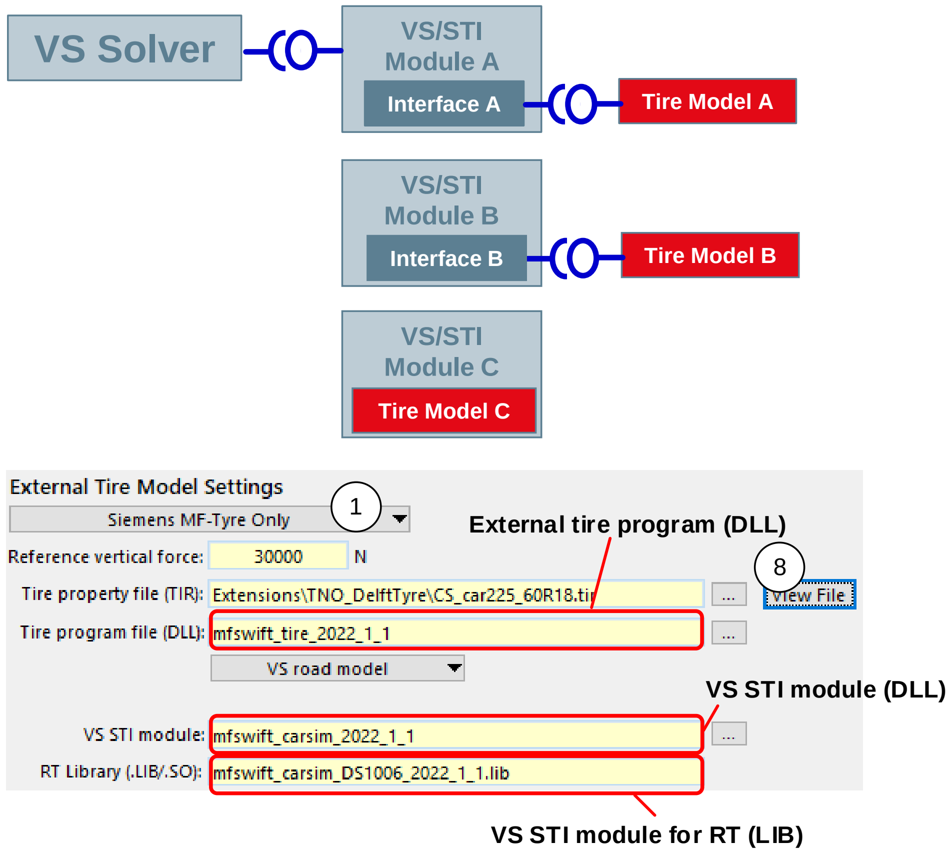



*Figure 31. Program structure for interfacing between a VS Solver and tire model (upper) and corresponding program settings on Tire (External) screen (below).*

**Note** The VS STI modules for MF-Tyre/MF-Swift and FTire are provided in precompiled DLL files. Source code for those commercial modules is not provided.

### *Tire Parameters*

The tire model parameters are typically specified in a separate file (`.tir`). You can click the view file button (8) in Figure 31) on the **Tire (External)** screen to open a text editor showing the tire properties. Alternatively, the tire parameters can be specified by a tabular data in VS format (root keyword: `STI_TYPARR`). The external tire model either reads a TIR file directly or CarSim reads the `STI_TYPARR` table and sends the array of parameters to the tire model depending on the cases.

For example, MF-Tyre/MF-Swift model is not able to access a text file when running on the dSPACE DS1006 system. Therefore, for on-line simulation (on the DS1006 system), the TIR file is automatically converted into VS format and written into a Parsfile (root keyword: `STI_TYPARR`) by the VS browser. The Parsfile is handled by the VS Solver and the tire data are sent into the TYPARR array of the MF-Tyre/MF-Swift model as shown in Figure 32.

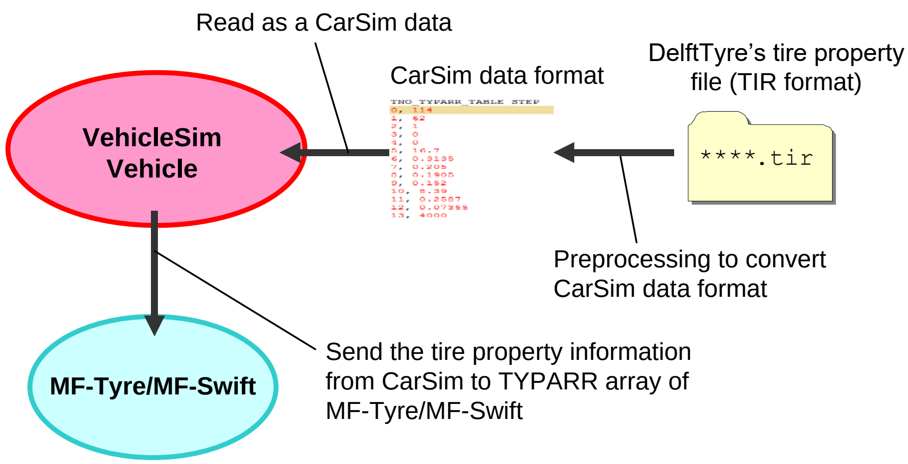



*Figure 32. Reading the tire property file for the real-time systems.*

### *Tire Model Option*

The **Tire (External)** screen can be alternatively used to communicate with the external tire models from the **Tire** screen for the internal tire model. The drop-down list of the tire model option involves three kinds of categories (Figure 33):

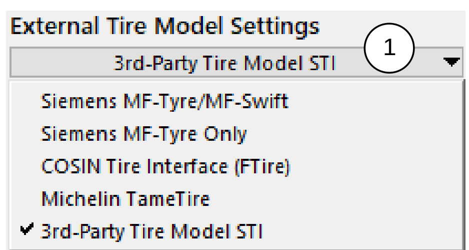



*Figure 33. External tire model option drop down list*

1. The first two options are MF-Tyre/MF-Swift models. One is MF-Tyre/MF-Swift model involving advanced models which require license from SIEMENS. MF-Tyre only runs with a CarSim/TruckSim license on Windows. MF-Tyre/MF-Swift model is able to run on dSPACE DS1006, SCALEXIO and Concurrent RT system (RT licenses are required by SIEMENS.)

2. The COSIN Tire interface option requires a license from COSIN Scientific Software to run the FTire model.

3. The Michelin TameTire option requires a license from Michelin to run the TameTire model.

4. The 3$^{rd}$-Party Tire Model STI option supports user defined tire models which are based on TYDEX STI (Standard Tire Interface) and the VS STI interface.

**Note** The **Tire** screen for the internal tire models has also legacy options for COSIN FTire.

## MF-Tyre/MF-Swift model options

There are two model options for MF-Tyre/MF-Swift: **Siemens MF-Tyre/MF-Swift** and **Siemens MF-Tyre Only**. Figure 34 shows the **Tire (External)** screen selected with **Siemens MF-Tyre/MF-Swift** option.

> **Note** Starting from the VehicleSim product version 2022.1, SIEMENS MF-Tyre/MF-Swift solver 9 and the VS STI module 18 supposed to be under `_prog\programs\solvers` location.
>
> TNO MF-Tyre/MF-Swift v6.2 model has been removed from VehicleSim products in 2021.

### *Basic Settings*

In the drop-down list of model options 1, select **Siemens MF-Tyre/MF-Swift** or **Siemens MF-Tyre Only**.

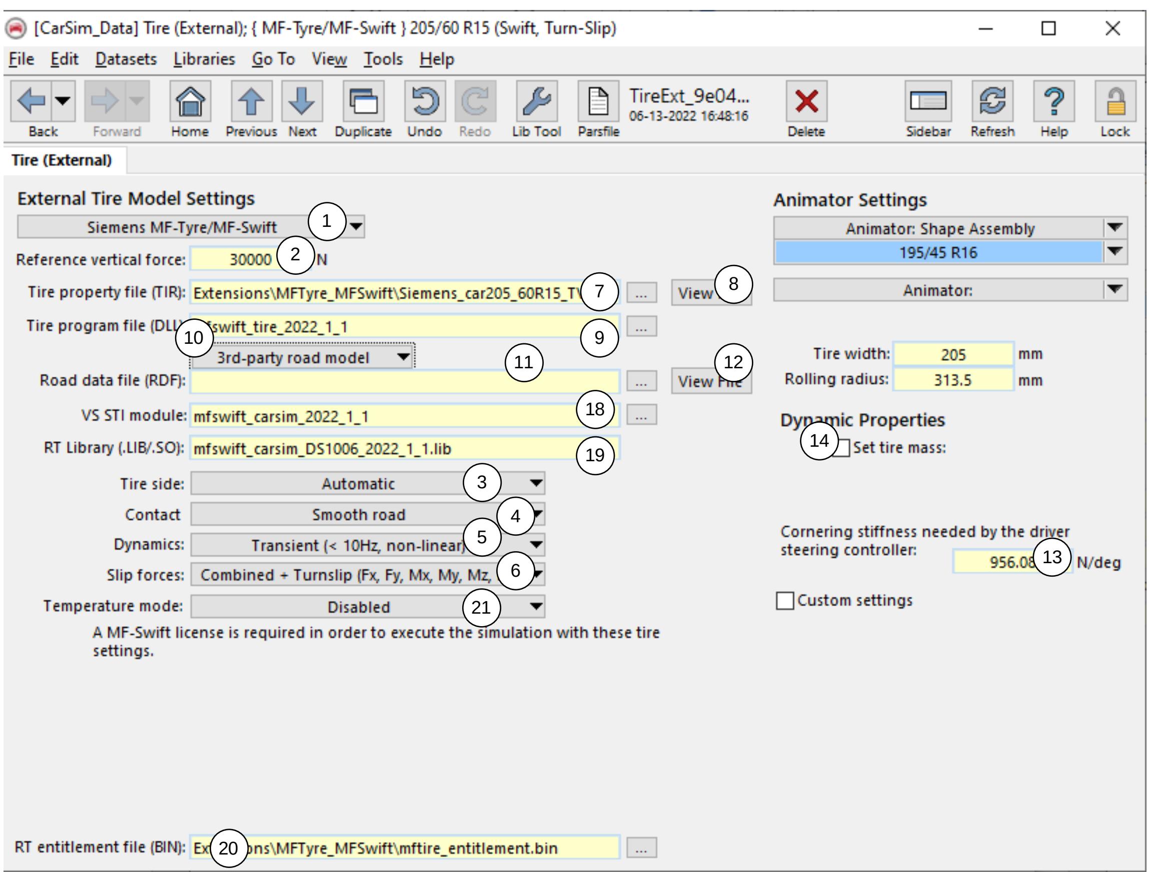



*Figure 34. Tire (External) screen with the MF-Tyre/MF-Swift selected.*

1 Model option drop-down list (keyword = `OPT_TIRE_MODEL`). Based on the selection, items on the screen that are not used are hidden. When the **Siemens MF-Tyre/MF-Swift** or **Siemens**

**MF-Tyre Only** is chosen, the screen has the appearance shown in Figure 34 and shows additional settings which are described in the next sub-section.

Further, if **Siemens MF-Tyre/MF-Swift** is chosen, a message will appear on this screen starting that you might have to have a MF-Swift license (issued by SIEMENS) in order to execute the simulation. The requirement for the MF-Swift license depends on how you select the options in the drop-down lists: 4, 5, and 6 and/or real-time application.

2 Reference vertical force (keyword = `FZ_REF`). This parameter is not used directly by the VS Math Model, but is provided as a standard property that can be used as a built-in reference for scaling other tire properties that are related to force.

3 Drop-down list to select the tire side (keyword = `DELFT_TYRE_SW`). The keyword is used with a 5-digit number (e.g.31114, the first digit is always "3".) This control specifies the second digit as follows:

| | | |
|---|---|---|
| Second digit: | 0 | automatic |
| | 1 | left |
| | 2 | right |
| | 3 | symmetric |
| | 4 | mirror |

4 Drop-down list to select the tire contact method (keyword = `DELFT_TYRE_SW`). This control specifies the third digit of the 5-digit number as follows:

| | | |
|---|---|---|
| Third digit: | 1 | smooth road |
| | 3 | moving flat surface |
| | 5 | enveloping contact |

5 Drop-down list to select the tire dynamics (keyword = `DELFT_TYRE_SW`). This control specifies the fourth digit of the 5-digit number as follows:

| | | |
|---|---|---|
| Fourth digit: | 0 | steady-state (< 1 Hz) |
| | 1 | transient (< 10 Hz, linear) |
| | 2 | transient (< 10 Hz, non-linear) |
| | 3 | rigid-ring (< 100 Hz, non-linear) |

6 Drop-down list to select the tire force evaluation (keyword = `DELFT_TYRE_SW`). This control specifies the fifth digit of the 5-digit number as follows:

| | | |
|---|---|---|
| Fifth digit: | 0 | none (Fz only) |
| | 1 | longitudinal (Fx, My, Fz) |
| | 2 | lateral (Fy, Mx, Mz, Fz) |
| | 3 | uncombined (Fx, Fy, Mx, My, Mz, Fz) |
| | 4 | combined (Fx, Fy, Mx, My, Mz, Fz) |
| | 5 | combined + turnslip (Fx, Fy, Mx, My, Mz, Fz) |

**Notes** Either MF-Tyre or MF-Swift is distinguished by the function switches of *Contact Method*, *Dynamics*, and *Force Evaluation* (4, 5, and 6, respectively).

MF-Tyre is able to run without any additional license on Windows. However, MF-Swift require specific license from SIEMENS. For example, if you select **Enveloping contact**, MF-Tyre/MF-Swift program requires the license for MF-Swift. Table 6 shows the option switches that are supported by VehicleSim license only or additional MF-Swift license.

*Table 6. MF-Tyre/MF-Swift function options and supported licenses.*

| Option item | MF-Tyre Only (VS license) | MF-Swift (Swift license) |
|---|---|---|
| **Contact** 4 | 1 | 1,5 |
| **Dynamics** 5 | 0,1,2 | 0,1,2,3 |
| **Slip forces** 6 | 0,1,2,3,4 | 0,1,2,3,4,5 |
| **Temperature** 21 | 0 | 0,1,2,3 |

7 Pathname and adjacent file browser button for a tire property file (`.tir`) (keyword = `DELFT_TYRE_DATA`).

**Note** When **Siemens MF-Tyre/MF-Swift** model option 1 is selected, the TIR file specified in this field is automatically converted to a VS format tabular data (root keyword: `STI_TYPARR`) by the VS Browser.

VS Solver reads the tabular data and sends the values to TYPARR array of the tire model instead of sending the pathname of the TIR file.

8 View file button to open text editor to show and edit the tire property file.

9 Tire program name without the file extension and adjacent file browser button (keyword = `DELFT_TYRE_DLL`). The tire program should locate in the vehicle solvers folder:

```
{Browser Installed folder}\Programs\solvers\
```

The file name of the tire program should be:

Windows 32bit program file name: xxx.dll32
Windows 64bit program file name: xxx.dll64
Linux 64bit program file name: xxx.so

**Note** CarSim/TruckSim includes MF-Tyre/MF-Swift v2022.1 (both 32-bit and 64-bit).

10 Drop-down list to select an external road data file (optional). If the external road is selected, the internal VehicleSim road is ignored and the external road description is used.

(11) Pathname and adjacent file browser button for a regular road data file (`.rdf`) or Open CRG data file (`.crg`) (keyword = `DELFT_TYRE_ROAD`).

(12) View file button to open text editor to show and edit the road data file.

(13) Cornering stiffness of tire, for use by driver path-follower model (keyword = `KY_TIRE`). The built-in driver model uses tire cornering stiffness to determine the steering wheel angle needed to follow a target path. (For details on the driver model, see the **Steering Controller (Driver Model)** technical memo that can be accessed from the **Help** menu.)

(14) Checkbox and data field for the spin and diametral moment of inertia and mass (keyword for checkbox = `OPT_TIRE_INERTIA_EXTERNAL`). If the box is not checked, the tire inertia and mass from the external tire file (`.tir`) (7) are used. If the box is checked, a data field for the inertia and mass are shown (Figure 35); the value from this field will be used to override the value from the external tire file (`.tir`).

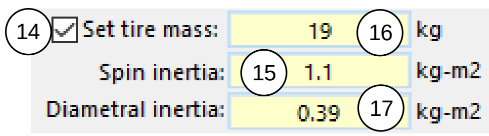



*Figure 35. Checkbox and data field for the tire inertia and wheel mass.*

> **Note:** If the box (14) is not checked and the drop-down list of the dynamics (5) is selected as rigid-ring, the spin and diametral moment of belt inertia and belt mass are subtracted from the total spin and diametral inertia and tire mass in the external tire file (`.tir`), respectively. Table 7 summarizes values used in the simulation for various cases.

(15) Spin moment of inertia of one tire/wheel (keyword = `IT`). The spin inertias of the tire, wheel, and other rotating components are lumped into one value.

(16) Mass of the tire and rim (keyword = `M_TIRE`).

(17) Diametral (yaw and roll) moment of inertia of one tire/wheel (keyword = `IT_XXZZ`). The diametral inertias of the tire, wheel, and other rotating components are lumped into one value.

(18) VS STI Tire module name without the file extension and adjacent file browser button for the VS STI module for running on Windows or Linux (`*.dll32, *.dll64, *.so`) (keyword = `EXTERNAL_TIRE_MODULE_DLL`). The file specified on this field should be compatible with the tire program file (9) if you have. These tire modules locate in the vehicle solvers folder:

```
{Browser Installed folder}\Programs\solvers\
```

The file name of VS STI tire module should be:

Windows 32bit module file name: xxx.dll32
Windows 64bit module file name: xxx.dll64
Linux 64bit module file name: xxx.so

*Table 7. Wheel/tire properties adopted in the simulation for various cases.*

| Checkbox (14) status | **Non-rigid-ring dynamics** (drop-down list (5)) | **Rigid-ring dynamics** (drop-down list (5)) |
|---|---|---|
| not checked | **Spin inertia**: `IYY` in the external tire file (7) (`.tir`)<br>**Diametral inertia**: `IXX` in the external tire file (7) (`.tir`)<br>**Mass:** `MASS` in the external tire file (7) (`.tir`) | **Spin inertia:** `IYY-BELT_IYY` in the external tire file (`.tir`)<br>**Diametral inertia:** `IXX-BELT_IXX` in the external tire file (`.tir`)<br>**Mass:** `MASS-BELT_MASS` in the external tire file (`.tir`) |
| checked | **Spin inertia**: value specified in the data field (15)<br>**Mass:** value specified in the data field (16)<br>**Diametral inertia**: value specified in the data field (17) | |

### *Real-Time Settings (CarSim only)*

(19) File name of the RT library for running on real-time dSPACE DS1006 system (`***.lib`), dSPACE SCALEXIO (`***.so`), and Concurrent RT (`***.so`) (keyword = `STI_RT_LIB`). This field specifies only the file name without path. This library file is supposed to be a VS STI module together with MF-Tyre/MF-Swift tire program implementation.

(20) Pathname and adjacent file browser button for an entitlement file which is only needed to run SIEMENS MF-Tyre/MF-Swift on RT systems (keyword = `*ENTITLE_BIN`).

> **Notes** The entitlement file specified in field (20) is automatically converted from binary format to VS text format tabular data (root keyword: `MFTYRE_ENTITLEMENT`) and written in the PAR file by VS browser.

### *Temperature and velocity Option*

The temperature and velocity model can be activated and specified the option by the drop-down list (21). The model is also activated through the `TV_MODEL` parameter in the tire property file (7) . The drop-down list (21) is higher priority and it overrides the `TV_MODEL` parameter in the tire property file.

(21) Drop-down list to select the temperature and velocity model (keyword = `DELFT_TYRE_TVM`). This control specifies the following four options with single digit number:

| | |
|---|---|
| 0 | disabled |
| 1 | static |
| 2 | dynamic without the inflation pressure change |
| 3 | dynamic with the inflation pressure change |

| | |
|---|---|
| **Note** | In order to run with the temperature and velocity model, an additional MF-SwiftTV license is required from SIEMENS. |

### *Detailed Information for MF-Tyre/MF-Swift Model*

Detailed information is available for the current MF-Tyre/MF-Swift v2022.1 model in the *MF-Tyre/MF-Swift User Manual*, available from the **Help** menu.

## COSIN FTire model option

In the drop-down list of model option on **Tire (External)** screen (Figure 36), select **COSIN Tire Interface (FTire)** 1.

| | |
|---|---|
| **Note** | If this COSIN Tire Interface (FTire) option is chosen, a message will appear on the screen stating that you need a COSIN FTire license in order to execute the simulation. |

1 Model option drop-down list (keyword = `OPT_TIRE_MODEL`). Based on the selection, items on the screen that are not used are hidden. When the **COSIN Tire Interface (FTire)** is chosen, the screens have the appearance shown in Figure 36.

2 Reference vertical force (keyword = `FZ_REF`). This parameter is not used directly by the VS Math Model, but is provided as a standard property that can be used as a built-in reference for scaling other tire properties that are related to force.

3 Pathname and adjacent file browser button for a tire property file `(***.tir)` (keyword = `FTIRE_DATA`).

4 View file button to open text editor or COSIN/TIRETOOLS to show and edit the tire property file 3.

5 Pathname and adjacent file browser button for a tire program file (keyword = `FTIRE_CTI_DLL`).

| | |
|---|---|
| **Note** | The FTire programs are not installed with CarSim or TruckSim; they must be installed separately in order to co-simulate FTire with CarSim or TruckSim. |

6 Checkbox to select an external road data file (optional). If this box is checked, the internal VehicleSim road is ignored and the external road description is used.

7 Pathname and adjacent file browser button for a road data file `(***.rdf)` (keyword = `FTIRE_ROAD`). This field is not shown unless the checkbox for the external road is checked 6.

8 View file button to open text editor to show and edit the road data file. This button is not shown unless the checkbox for the external road is checked 6.

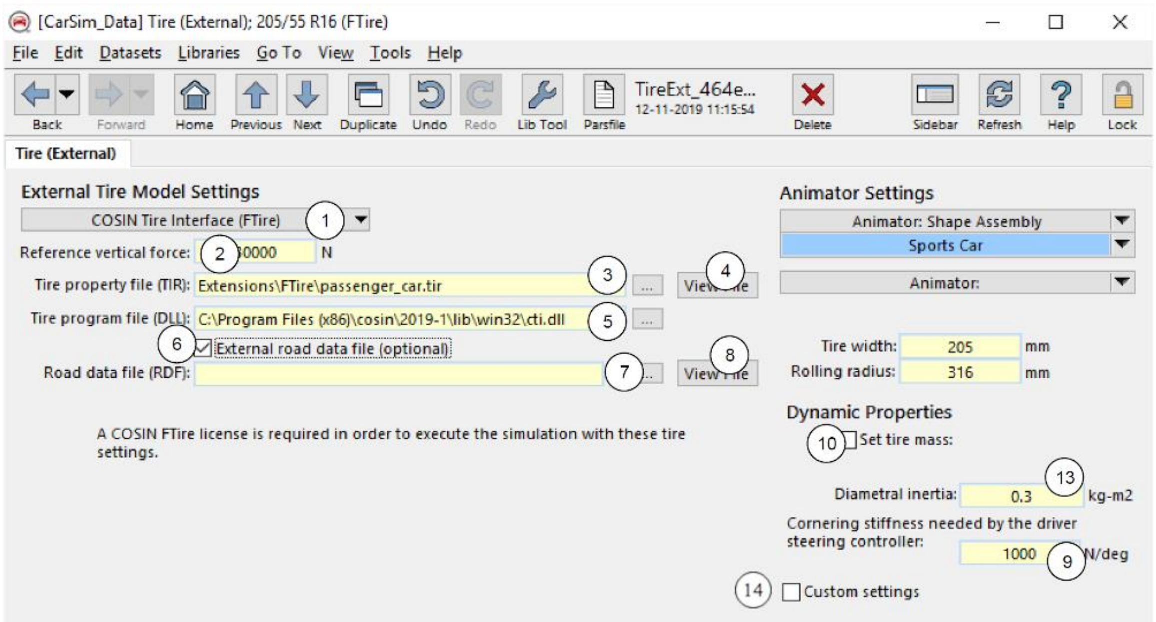



*Figure 36. Alternate Tire (External) screen with the COSIN Tire Interface (FTire) selected.*

9 Cornering stiffness of tire, for use by driver path-follower model (keyword = `KY_TIRE`). The built-in driver model uses tire cornering stiffness to determine the steering wheel angle needed to follow a target path. (For details on the driver model, see the **Steering Controller (Driver Model)** technical memo that can be accessed from the **Help** menu.)

10 Checkbox and data field for the spin moment of inertia and mass (keyword for checkbox = `OPT_TIRE_INERTIA_EXTERNAL`). If the box is not checked, the spin moment of inertia and wheel mass from the external tire file (`.tir`) 3 are used. If the box is checked, a data field for the spin inertia and wheel mass are shown (Figure 37); the value from this field will be used to override the value from the external tire file (`.tir`).

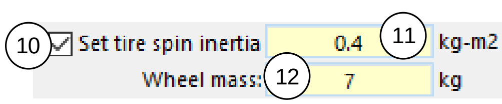



*Figure 37. Checkbox and data field for the tire spin moment of inertia and wheel mass.*

11 Spin moment of inertia of one tire/wheel (keyword = `IT`). The spin inertias of the tire, wheel, and other rotating components are lumped into one value.

12 Wheel mass (keyword = `M_TIRE`). This mass is added with the front fork mass set on **Suspension: Fork** screen. The sum of those masses is the total front unsprung mass. As far as the total unsprung mass is non-zero, the calculation is executed without any numerical problem.

13 Diametral (yaw and roll) moment of inertia of one tire/wheel (keyword = `IT _XXZZ`). The diametral inertias of the tire, wheel, and other rotating components are lumped into one value.

14 Checkbox for custom settings. Enabling this checkbox will display a miscellaneous yellow field and a blue link field, potentially useful for custom settings.

### *Suspension and steering compliances with FTire*

The CarSim and TruckSim suspension and steering compliance data establish a quasi-static deflection of the unsprung mass relative to its location determined by the suspension kinematics functions. *Quasi-static* means that the position and orientation variables of the unsprung mass are modified by the compliances, but the time-derivatives of these variables are not. This approach works well, but results in some differences between the velocity of the unsprung mass implied by differentiating the position variables and the velocity variables themselves.

The FTire model necessarily takes as input the rim center position, rim reference frame orientation, rim center translational velocity, and rim frame angular velocity. These terms include a dependency on the unsprung mass kinematics. Including large compliance values causes the inputs to the FTire model to differ enough for FTire to fail with the following error message appearing in the FTire log file: "…rim motion states sent to CTI are inconsistent; velocities and differentiated positions do not match". For compatibility with FTire, the CarSim or TruckSim quasi-static suspension and steering compliances should be reduced in magnitude. One way to confirm this is the root cause is to disable the compliances entirely by setting them all equal to zero.

For convenience, Table 8 and Table 9 summarize the quasi-static suspension and steering compliance keywords which need be assessed when using FTire. As a practical matter, the values for these keywords may be overwritten from the external tire screen by using the custom settings checkbox ⑭ to establish a link to an appropriate Generic VS Commands dataset.

*Table 8. Summary of quasi-static suspension compliance keywords.*

| Suspension compliance configurable functions… | Indexed by… | Independent and virtual steering axis suspensions? | Steered solid axle suspensions? | Non-steered solid axle suspensions? |
|---|---|---|---|---|
| `CT_FX` | Wheel | x | x | |
| `CS_FY` | Wheel | x | x | |
| `CS_MZ` | Wheel | x | x | |
| `CC_FX` | Wheel | x | x | |
| `CI_FY` | Wheel | x | x | |
| `CI_MZ` | Wheel | x | x | |
| `C_LONG` | Wheel | x | | |
| `C_LAT` | Wheel | x | | |
| `CD_MY` | Wheel | x | | |
| `C_LONG_AXLE` | Axle | | x | x |
| `C_LAT_AXLE` | Axle | | x | x |

*Table 9. Summary of quasi-static steering compliance keywords.*

| Steering compliance keywords… | Type… | Indexed by… | Independent and virtual steering axis suspensions, steered solid axle suspensions with symmetric compliance option (`OPT_CS=0`)? | Steered solid axle suspensions, asymmetric compliance option (`OPT_CS>0`)? |
|---|---|---|---|---|
| `STEER_COMP` | Configurable function | Axle | x | |
| `CS_MZ_ROD` | Parameter | Axle | | x |
| `CS_MZ_SHAFT` | Parameter | Axle | | x |
| `C_WRAP` | Parameter | Axle | | x |

**Note** The independent suspension model in CarSim supports a dynamic longitudinal suspension compliance which is not subject to the position/velocity discrepancy. This feature is controlled by the command `DEFINE_SUSP_X_DOF`. The dynamic compliance replaces the quasi-static compliance established by `C_LONG`. For more information, see the *Suspension Systems* help manual.

### *Detailed Information for FTire Model*

The detailed information for FTire model is available in the COSIN Scientific Software home page: www.cosin.eu/prod_FTire.

## Michelin TameTire Model Options

The **Tire (External)** screen with the **Michelin TameTire** option ① is shown in Figure 38.

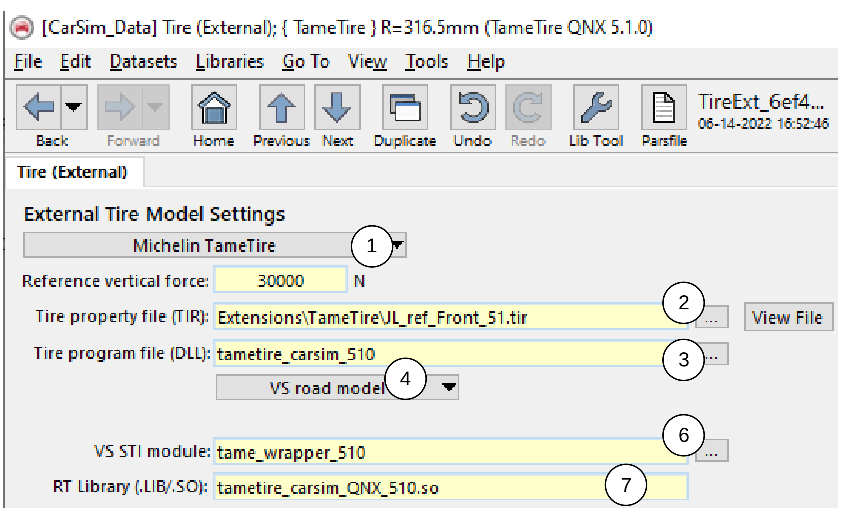



*Figure 38. Tire (External) screen with the Michelin TameTire model option selected.*

| | |
|---|---|
| **Note** | Starting from the VehicleSim product version 2022.1, Michelin TameTire solver 3 and the VS STI module 6 supposed to be under `_prog\programs\solvers` location. |

1 Model option drop-down list (keyword = `OPT_TIRE_MODEL`).

2 Pathname and adjacent file browser button for a tire property file `(***.tir)` (keyword = `USER_STI_DATA`).

| | |
|---|---|
| **Note** | When **Michelin TameTire** model option 1 is selected, the TIR file specified in this field is automatically converted to a VS format tabular data (root keyword: `STI_TYPARR`) by the VS Browser.<br><br>VS Solver reads the tabular data and sends the values to TYPARR array of the tire model instead of sending the pathname of the TIR file.<br><br>Earlier distribution of TameTire RT for CarSim from Michelin required manually convert and link to a tabular data. However, this **Tire (External)** screen in CarSim v2021.1 and newer version automatically converts the TIR file to a VS format table with a conversion program which checks the license from Michelin. |

3 Tire program name without the file extension and adjacent file browser button (keyword = `USER_STI_DLL`). The tire program should locate in the vehicle solvers folder:

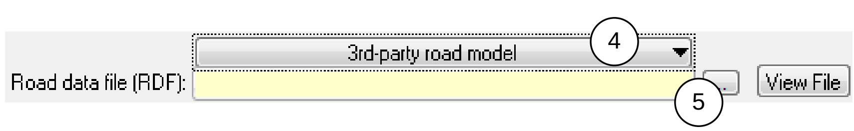
`{Browser Installed folder}\Programs\solvers\`

The file name of the tire program should be:

Windows 32bit program file name: xxx.dll32
Windows 64bit program file name: xxx.dll64

Linux 64bit program file name: xxx.so

4 Drop-down list to select 3rd-party road model (optional). If the 3rd-party road is selected, the internal VS road is ignored and the external road description is used.




5 Pathname and adjacent file browser button for a road data file `(***.rdf)` (keyword = `USER_STI_ROAD`). This field is not shown unless the drop-down list for 3rd-party road is selected 4.

6 VS STI Tire module name without the file extension and adjacent file browser button for a TameTire wrapper running on Windows or Linux `(*.dll32, *.dll64, *.so)` (keyword = `EXTERNAL_TIRE_MODULE_DLL`). The file specified on this field should be

compatible with the tire program file (3) if you have. These tire modules locate in the vehicle solvers folder:

```
{Browser Installed folder}\Programs\solvers\
```

The file name of VS STI tire module should be:

Windows 32bit module file name: xxx.dll32
Windows 64bit module file name: xxx.dll64

Linux 64bit module file name: xxx.so

(7) File name of the RT library. This library file is supposed to be a VS STI module together with tire model implementation for dSPACE SCALEXIO (`***.so`) system (keyword = `STI_RT_LIB`).

### *Detailed Information for TameTire Model*

Detailed information is available for TameTire model in the *TameTire User Manual*.

## Custom 3rd-Party Tire Model Options

The **Tire (External)** screen with the **3rd-Party Tire Model STI** option (1) is shown in Figure 39. This option can be used to connect custom external tire models through the VS STI interface.

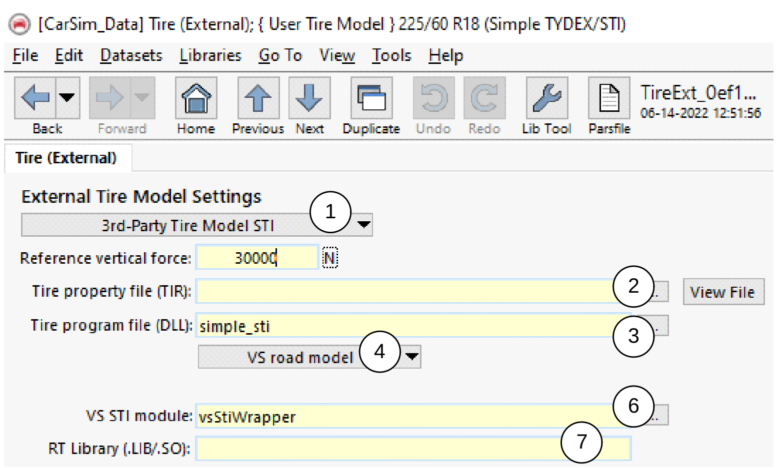



*Figure 39. Tire (External) screen with the 3rd-Party Tire Model STI option selected.*

**Note** Starting from the VehicleSim product version 2022.1, 3rd-party tire model solver (3) and the VS STI module (6) supposed to be under `_prog\programs\solvers` location.

(1) Model option drop-down list (keyword = `OPT_TIRE_MODEL`).

(2) Pathname and adjacent file browser button for a tire property file (`***.tir`) (keyword = `USER_STI_DATA`).

3 Tire program name without the file extension and adjacent file browser button (keyword = `USER_STI_DLL`). The tire program should locate in the vehicle solvers folder:

`{Browser Installed folder}\Programs\solvers\`

The file name of the tire program should be:

Windows 32bit program file name: xxx.dll32
Windows 64bit program file name: xxx.dll64

Linux 64bit program file name: xxx.so

4 Drop-down list to select 3rd-party road model (optional). If the 3rd-party road is selected, the internal VS road is ignored and the external road description is used.


5 Pathname and adjacent file browser button for a road data file `(***.rdf)` (keyword = `USER_STI_ROAD`). This field is not shown unless the drop-down list for 3rd-party road is selected 4.

6 VS STI Tire module name without the file extension and adjacent file browser button for an external tire module file running on Windows or Linux (`*.dll32, *.dll64, *.so`) (keyword = `EXTERNAL_TIRE_MODULE_DLL`). The file specified on this field should be compatible with the tire program file 3 if you have. These tire modules locate in the vehicle solvers folder:

`{Browser Installed folder}\Programs\solvers\`

The file name of VS STI tire module should be:

Windows 32bit module file name: xxx.dll32
Windows 64bit module file name: xxx.dll64

Linux 64bit module file name: xxx.so

7 File name of the RT library. This library file is supposed to be a VS STI module together with tire model implementation (keyword = `*STI_RT_LIB`).

If you specify an external tire model in the tire program file field 3, you must also specify the VS STI module 6, which interfaces between the VS Solver and the external tire model. There are two sample C programs provided under `CarSim_Data\Extensions\User_Tire`.

## Animator and custom settings

The visual appearance of the tire and wheel in the animator can be specified with a link (1 in Figure 40) to an animator dataset with one or more objects that will define the tire and wheel assembly. Alternatively, the wheel can be shown as a simple cylinder. Either way, the animator sizes the wheel shape in proportion to the radius and width specified on this screen.

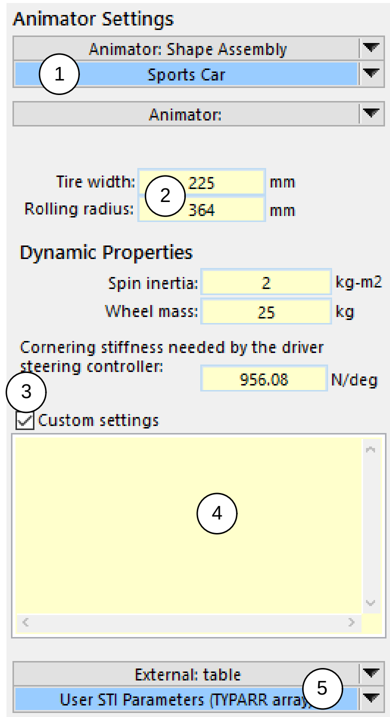



*Figure 40. Animator and custom settings.*

① Optional link to animator data to show the tire wheel assembly in the animation. This link is visible only when the adjacent checkbox is checked.

② Tire width and radius (keyword = `SET_THICKNESS` and `SET_RADIUS_SGUI`). These dimensions are used only to generate animations of the wheels; they have no influence on the simulated vehicle response.

> **Note** Alternatively, the wheel can be shown in more detail in COSIN FTire option by COSIN/graphics, such as tire deformation, tire contact surface(s), tire temperature, tire pressure (Figure 41).

③ **Custom settings** checkbox. This checkbox adds a miscellaneous data field ④ and a data link ⑤ (see Figure 40).

④ Miscellaneous data field. You can overwrite the data settings using this field (e.g. `DELFT_TYRE_SW 31124`).

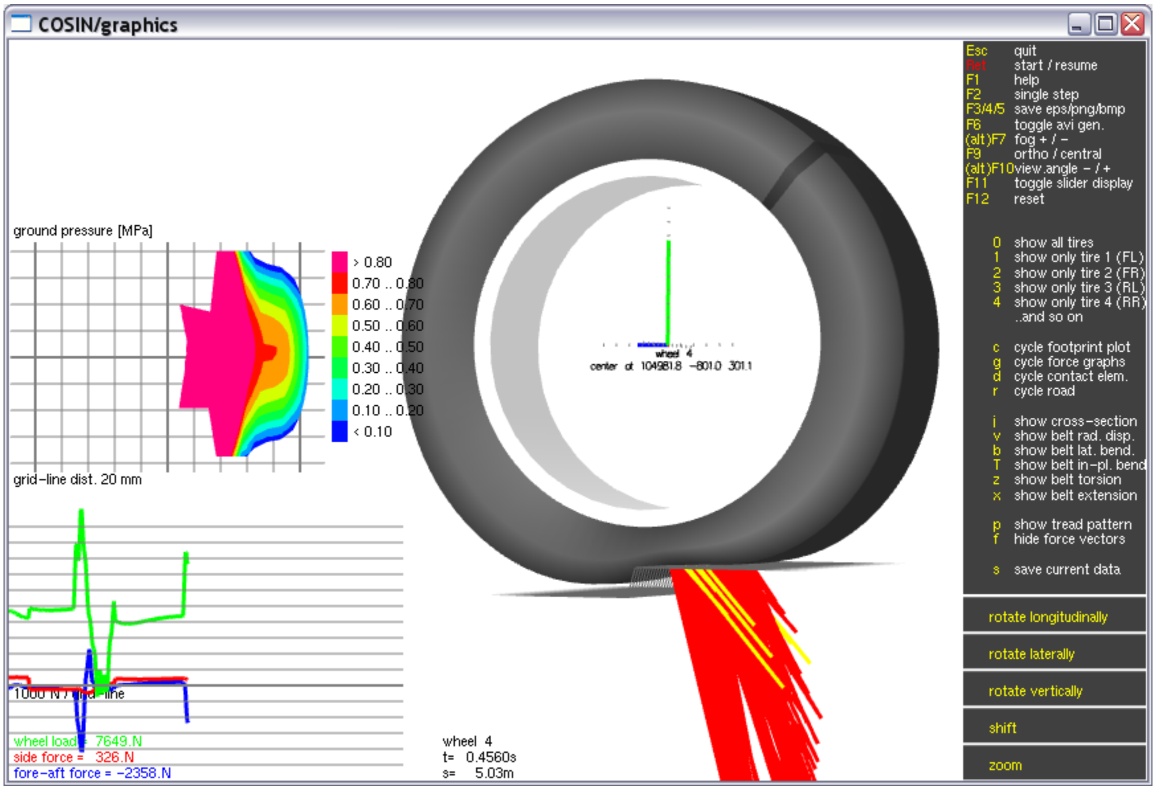



*Figure 41. COSIN/graphics showing tire deformation (3D), tire forces graphs and ground contact pressure.*

5 Miscellaneous data link.

# Appendix: Comments on PAC 2002

Pacejka Magic Formula (Delft Tyre) tire models in CarSim and TruckSim are produced by TASS International (SIEMENS) and distributed by Mechanical Simulation under license. In our view, these represent the “gold standard” among Magic Formula tire models, as their development was overseen by Dr. Pacejka himself through the end of his career there.

The SIEMENS / Delft Tyre package supports many revisions as outlined in the MFTyre-MFSwift manual. This manual can be found under the **Help** menu item in the Reference Manuals section. The description of fit types is found in section 5.3.4 “Backward Compatibility”

In reviewing the chart there, note the lack of reference to PAC2002. There are several implementations of PAC2002 models offered by various sources, based on the work published in Dr. Pacejka’s book, *Tire and Vehicle Dynamics* [3]. This model is not directly supported by SIEMENS / Delft Tyre and is not available in CarSim and TruckSim. The several PAC2002 implementations available from various sources are not identical. We have chosen not to develop our own based on the book. When you obtain parameters for a Magic Formula tire, be sure to direct the source performing the curve fit to use one of the types listed in the manual.

# References


1. Bakker, E., Pacejka, H.B., and Lidner, L., "A New Tire Model with an Application in Vehicle Dynamics Studies," 4th Autotechnologies Conference, Monte Carlo, 1989, SAE 890087.

2. Pacejka, H., Sharp, R.S. "Shear Force development by Pneumatic Tyres in Steady State Conditions: A Review of Modelling Aspects" *Vehicle System Dynamics*, 20 (1991), pp. 121-176, Equations 88-93.

3. Pacejka, Hans B., *Tire and Vehicle Dynamics*, (2006, 2nd edition), SAE International, Warrendale, PA, USA

4. Loeb, J.S. et. al., "Lateral Stiffness, Cornering Stiffness and Relaxation Length of the Pneumatic Tire," SAE Paper No. 900129, 1990.

5. Bernard, J. E., Clover, C. L., "Tire Modeling for Low–speed and High–speed Calculations," SAE Paper No. 950311, 1995.

6. Rill, Georg, *First Order Tire Dynamics*, III European Conference on Computational Mechanics, 2006.

7. van der Jagt, P.: The Road to Virtual Vehicle Prototyping; new CAE-models for accelerated vehicle dynamics development. PhD-Thesis, Tech. Univ. Eindhoven, Eindhoven 2000, ISBN 90-386-2552-9 NUGI 834.