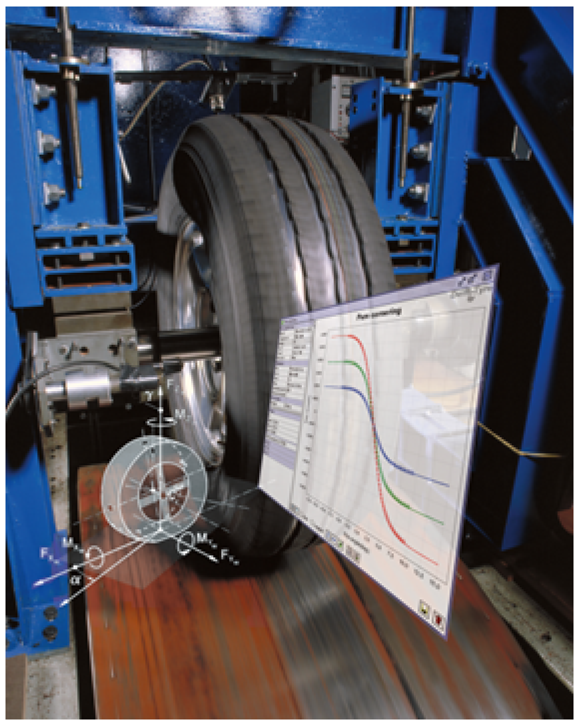
MF-Tyre/MF-Swift 6.1.2

Help Manual



TNO Automotive
The Netherlands
http://www.delft-tyre.nl
http://www.tno.nl

Document revision: 6-7-2010

Delft-Tyre

MF-Tool, MF-Tyre and MF-Swift are part of the Delft-Tyre product line, developed at TNO Automotive, Helmond, The Netherlands.

Printed: December, 2010 in Helmond, The Netherlands

Delft-Tyre

Publisher

TNO Automotive
PO Box 756
5700 AT Helmond
The Netherlands

Table of Contents

1 Release Notes

1.1 Contents of the 6.1.2 Release

This chapter describes the most important **changes** between the 6.1.1 and the 6.1.2 Release of MF-Tyre/MF-Swift.

New Features

- **OpenCRG** 100
  MF-Tyre and MF-Swift 6.1.2 support OpenCRG, a unified approach to represent 3D road data in tyre simulation. With OpenCRG large measured 3D road surfaces can be handled easily and efficiently. Next to the support of OpenCRG in the tyre model, MATLAB routines for data manipulation and generation are provided with the tyre model installation.

Enhancements

- Increased accuracy of the dynamic longitudinal force simulation on relatively small **drums** (< 2 m) when using the DRUM_RADIUS keyword in the road data files of Polyline 78 and Plank road 77.
  Further, a physical drum road 80 is made available as internal road of the tyre model; this makes simulations of a vehicle on drums possible. Note that the preferred way of simulating fixed spindle cleat experiments on a drum is still using the DRUM_RADIUS keyword in the road data files of Polyline and Plank road.

- Specific ADAMS 87 **road data files** are shipped with the MF-Tyre/MF-Swift such that Adams can deal with the TNO roads 76.

- Increased accuracy of the road contact 45 for 2D and 3D roads (i.e. enveloping model). The number of points that can be dealt with internally in the road contact for 3D roads has been increased, making calculation of high obstacles more accurate. For the road contact for 2D roads, the number of contact points per specified road increment is made equal to the number of points used in the road contact for 3D roads.

Bug Fixes

- Several bugs have been **solved** in the road contact for 2D and 3D roads resulting in increased robustness.

- The **libraries** libifcorem.dll and libmmd.dll are now shipped with the MF-Tyre/ MF-Swift for users without (implicit) Fortran runtime libraries installed, such as CarSim.

- **Visual C++ 2005 redistribution packages** are now shipped with the MF-Tyre/ MF-Swift for users without (implicit) C++ libraries installed.

Known Issues

- When using a fixed-step solver for the Dynamics mode 47 **tyre relaxation behaviour (< 10 Hz, nonlinear)**, the time-step of the simulation should be chosen small enough for the simulation to produce correct results. A variable-step solver will automatically reduced the time-step when required.

- When using an OpenCRG 100 data file in Windows format, warning messages will appear stating `WARNING: parseFileHeader: ignoring line XX`. This can be ignored. In Linux format no warning message will appear.

2 Installation Guide

Installation Guide

MF-Tyre 37 and MF-Swift 38 are plug-ins to Multi-body Simulation Packages 87 (MBS). This installation guide applies to the following MBS Packages only:

- MATLAB/Simulink (MathWorks)
- ADAMS (MSC)
- DADS (LMS)

The installation of MF-Tyre/MF-Swift for Microsoft Windows consists of the following steps:

- MF-Tyre/MF-Swift installation 11
- licensing the MF-Tyre/MF-Swift 11
- connecting MF-Tyre/MF-Swift to MBS Packages 22

The compatibility of MF-Tyre/MF-Swift to various MBS packages is described in a Compatibility Table 10.

As MF-Tyre/MF-Swift is focussed on Windows, the main installation steps are described for the Windows operating system. If you would like to use Linux please contact 34 our sales representative, and follow the instructions of Linux Installation 30 .

2.1 System Requirements

(Minimum) system requirements for MF-Tyre/MF-Swift are:

- Windows 2000, XP or Vista (32-bit), or Windows XP Professional x64 Edition (64-bit)
- Pentium 4 or higher
- 512 Mb of RAM
- MF-Swift: 25 Mb of disk space

Note 1: **Windows XP 64-bit Edition** (for **Intel Itanium** processors) is **not** supported.

Note 2: The MBS Package 87 MF-Tyre/MF-Swift is connected to may have specific system requirements.

Technical note on 64-bit: On a Windows 64-bit operating system, it is possible to run 32-bit and 64-bit applications:

- The MF-Tyre/MF-Swift 64-bit installation contains libraries for both 32-bit and 64-bit applications.
- The licensing mechanism 11 is a 32-bit application, but can be installed on a 64-bit operating system without problem.

2.2 Compatibility Table

MF-Tyre and MF-Swift are available for a wide variety of multi-body simulation packages. We may distinguish between:

Implementations by TNO

MATLAB/Simulink (MathWorks)
ADAMS (MSC)
DADS (LMS)

Implementations using a shared library

Recurdyn (FunctionBay)
CarSim/TruckSim/BikeSim (MSc)
Modelon (Dymola)
AVL software (AVL)

Native implementations

Virtual.Lab (LMS)
SIMPACK (SIMPACK AG)
Madymo (TASS)

The corresponding compatibility table is shown below.

Multi-body package	Version	Win32 **	Win64 ***	Linux
ADAMS (MSC)	2003	x		
	2005	x		x
	2005r2	x		x
	2007r1	x		x
	2008r1	x	x	
SIMPACK (SIMPACK AG)	*	*	*	*
MATLAB/Simulink	6.5	x		
	2006a and up	x	x	
Recurdyn (FunctionBay)	*	*	*	

Multi-body package	Version	Win32 **	Win64 ***	Linux
CarSim/TruckSim/BikeSim	*	*	*	
Modelon (Dymola)	*	*	*	
AVL software (AVL)	*	*	*	
Virtual.Lab (LMS)	*	*	*	*
DADS (LMS)	9.6	x		
MADYMO (TASS)	*	*	*	*

*: Availability depends on the implementation in the respective multi-body packages.
**: Win32 includes Windows 2000, XP and Vista
***: Win64 includes Windows XP Professional x64 Edition

2.3 Windows Installation

This chapter describes the steps to follow to install MF-Tyre/MF-Swift on a Windows system.

2.3.1 MF-Tyre & MF-Swift Installation

Note 1: To install the products you need to have administrator rights on the machine (for installing the program and setting the environment variables).

Note 2: On **Vista** machines, do **not** install MF-Tool, MF-Tyre or MF-Swift in C:\Program Files\

To install MF-Tyre/MF-Swift run the following application. Setup wizards will guide you through the installation process.

```
setup_MFTyre_MFSwift_xxx.exe
```

On Windows 64-bit editions run:

```
setup_MFTyre_MFSwift_xxx_x64.exe
```

After installing the tyre models, MF-Tyre will work immediately. For MF-Swift you have to activate your license. Please follow the steps described in the License Manual 11.

2.3.2 License Manual

This chapter describes the licensing mechanism for the **TNO Delft-Tyre** software products. Licensing is used in two ways in this chapter: licensing regarding terms and conditions, and licensing as a mechanism to protect the software from unauthorized use. The context will reveal the meaning.

The terms and conditions governing the licensing of **MF-Tyre** consist solely of those set forth in the document titled 'License conditions of Mf-tyre software'. The terms and conditions governing the licensing of **MF-Swift**, **MF-Tool** and other Delft-Tyre software consist solely of those set forth in the **written contracts** between TNO and its customers.

MF-Tyre/MF-Swift includes libraries of OpenCRG, licensed under the Apache License, Version 2.0.

The software is protected / licensed with Safenet Sentinel. Licensed products include:

- MF-Swift 38
- MF-Tool

Note: For MF-Tyre no license is required.

License Types

The Delft-Tyre software products are available with two types of licenses:

1. Stand-alone license
With a stand-alone license you can run Delft-Tyre products only on one specific computer, i.e. the license is restricted to this single computer. In the Delft-Tyre products the stand-alone license is implemented as a "network license", but the server and client are on the same computer.

2. Floating license
With a floating license multiple users ("clients") are allowed to share one or more licenses for a Delft-Tyre product. Access to the licenses is controlled by a license server, which monitors active licenses and releases them when users close their software. When the maximum number of floating licenses has been reached, access by new users will be rejected. When one user finishes using the software, another user can start using it.

Note: This licensing scheme applies to both MF-Tool and MF-Swift. Please note that the tyre model MFTyre does not require a license.

Installation Procedure

A number of steps have to be taken to **install** and **activate** the **licensing software**, they are described in the next chapters. The table below shows the required steps per license type. After installing the license, the license system may be tested 19.

Step	Description	License Type		
		Stand-alone	Floating	
			Server	Client
1	Install the License Architecture 13	v	v	
2	Obtain machine ID 13	v	v	
3	Install the License Manager 16	v	v	
4	Activate the license 17	v	v	
5	Install clients 19			v

2.3.2.1 Installing the License Architecture

This step is **required** for:

- Stand-alone license
- Floating license: Server side

Install the licensing infrastructure on your computer by running the following application:

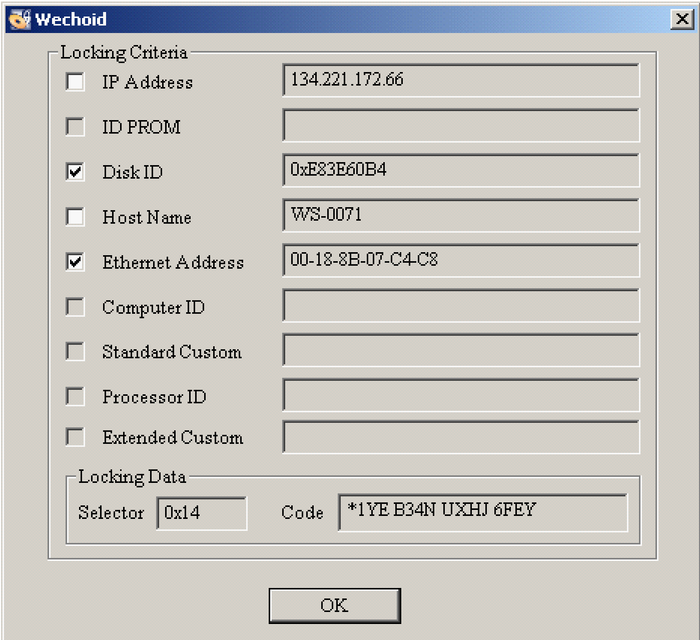
```
setup_DelftTyre_licensing_810.exe
```

2.3.2.2 Obtain required computer information

This step is **required** for:

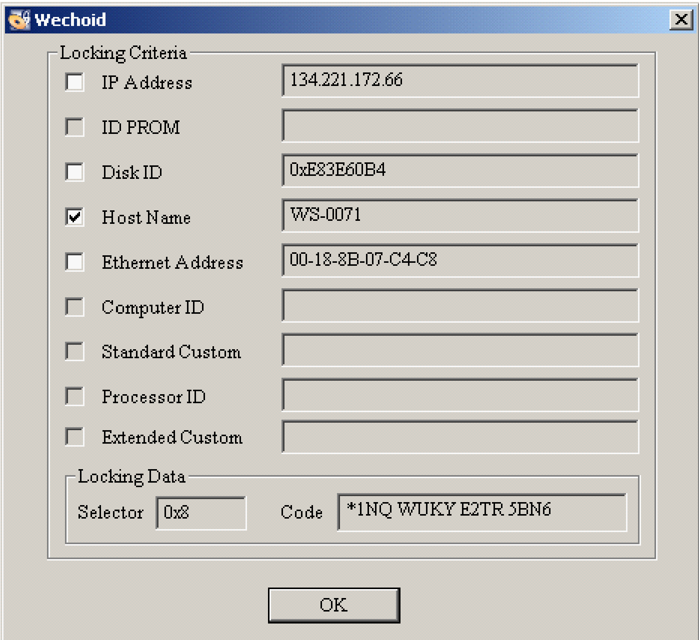
- Stand-alone license
- Floating license: Server side

The TNO Delft-Tyre license will be locked to a specific computer (a stand-alone machine or license server). Therefore TNO needs some information to identify this computer. To obtain this information, click in the **Licensing 8.1.0** submenu on **Obtain Machine ID (wechoid)**. In the screen that appears, make sure that ONLY "Disk ID" and "Ethernet Address" are checked, see the figure below.



Copy the information at the bottom of the window inside the “Locking Data” panel into a mail or document. You can do this by selecting the code with your mouse and pressing CRTL+C. So, in this example: Selector = 0x14, Code = *1YE B34N UXHJ 6FEY.

Then, make sure that ONLY “Host Name” is checked.



Again, the information needed is at the bottom of the window inside the “Locking Data” panel. **Copy** this data and **paste** it into a mail or document. You can do this by selecting the code with your mouse and pressing CRTL+C.

So, in this example: Selector = 0x8, Code = *1NQ WUKY E2TR 5BN6. You can close the Wechoid tool.

Send both **selectors** and **codes** to your TNO representative, in this example:

- Selector = 0x14, Code = *1YE B34N UXHJ 6FEY
- Selector = 0x8, Code = *1NQ WUKY E2TR 5BN6

Note: Please send both **selectors** and **codes** to your TNO representative using **copy** and **paste** to avoid typing errors at generating the license.

License File

TNO will then generate a license file. The information in the license file can look for example like this:

*B
9kPsGlormgwdxBd,K8Zqf0PmfD4AaVBXxzmi5PIhTcfkp7fdX,lSjhhafNrhSdPgV5wZH
A67hx9eXyBLIQtiRAVaL03qzk9Dj:Rn1La3bHrXf9sEllvrwgM7Jj7CuCamfDV6RUzkuc
wHQoVjDkiqlLKAfXX:zfzV,RjqrkWCm5fpPyLC# "MFFitSwiftFitSA" version "6.1", expires Midnight of Dec 31, 2009, exclusive

*B
AIT9:DtBiLIPoFAmBhZS3EQA5u:TYvKE4zHKspkH5JVBoCPaRVyUjrekZNQg3Qd6V
GBKtp1ZW7YENOhIqcOW2S3eAWgHJ8VA3KHFs4DfmqRNVQIPjqm01iCyh,wZmN
wcoOVJ5JDW6Ru9RhoNnLuSrcZX3H# "MFToolSA" version "6.1", expires Midnight of Dec 31, 2009, exclusive

*B
Plg34PWINfGQeFYGrw4EVzBshrr4cWjyUo0UC6dlc7LXoTZkB,c5fEhPJu4DWocY36
A1veRqf9Z434Wfmjjn8cLcVMHSMAQpZOsl88,BBLBvVGDQUKv9WTjxC4q,jhI5o3Vq
JA05PHvSXWciD3r5iuodfzp1jrOA3EsolvNdgiURPKLA# "MFSwiftTyreModelSA" version "6.1", expires Midnight of Dec 31, 2009, exclusive

2.3.2.3 Installing the License Manager

This step is **required** for:

- Stand-alone license
- Floating license: Server side

To install the License Manager, select **Install Sentinel RMS License Manager** from the Start Menu (All Programs > TNO Delft-Tyre > Licensing 8.1.0). A wizard is started that will guide you through the installation process.

Firewall

When installing the License Manager you will be asked to unblock the Windows Firewall for the Sentinel RMS License Manager on a Windows XP system with SP2 or higher. You should select 'Yes' to unblock the firewall. When using a different firewall you may also get messages like “Sentinel RMS License Manager is trying to access the network”.

Note that in order for the License Manager to operate and communicate with client machines your firewall should not block the License Manager. For a stand-alone licenses you may also have to unblock your firewall for the License Manager, since communication between the server and its clients or visa versa may be seen as network access even if the server and client are actually one single computer.

MF-Tool only

This section is valid for MF-Tool only. Finally, some firewalls may give “network access” warnings when parts of the Delft-Tyre software, like mffit.exe, swift_fitter.exe, etc., try to get license information from the server. In order for the software to function properly you have to unblock your firewall for these Delft-Tyre software components.

Administrator rights

You need to have administrator rights on your machine to install the License Manager. In case you encounter a problem that looks like the figure below, you probably do not have administrator rights on the machine. In this case, choose “Yes”. The Sentinel RMS License Manager is NOT installed. Contact your network administrator to obtain administrator rights and try to install the License Manager again.

> **Note**: You need to have **administrator rights** to be able to install MF-Tyre/MF-Swift

2.3.2.4 Activate the license

This step is **required** for:

- Stand-alone license
- Floating license: Server side

Time Tampering

> **Note**: Please do not change the clock time of your computer!

The License Manager guards against reuse of expired licenses. If for example the system clock has been set back to an earlier date than when a license has expired, the License Manager will detect this and will give a “time tampering” error. If such a problem occurs due to accidentally changing the time settings of your computer, please contact TNO to “clean” your computer to allow you to “reuse” your license.

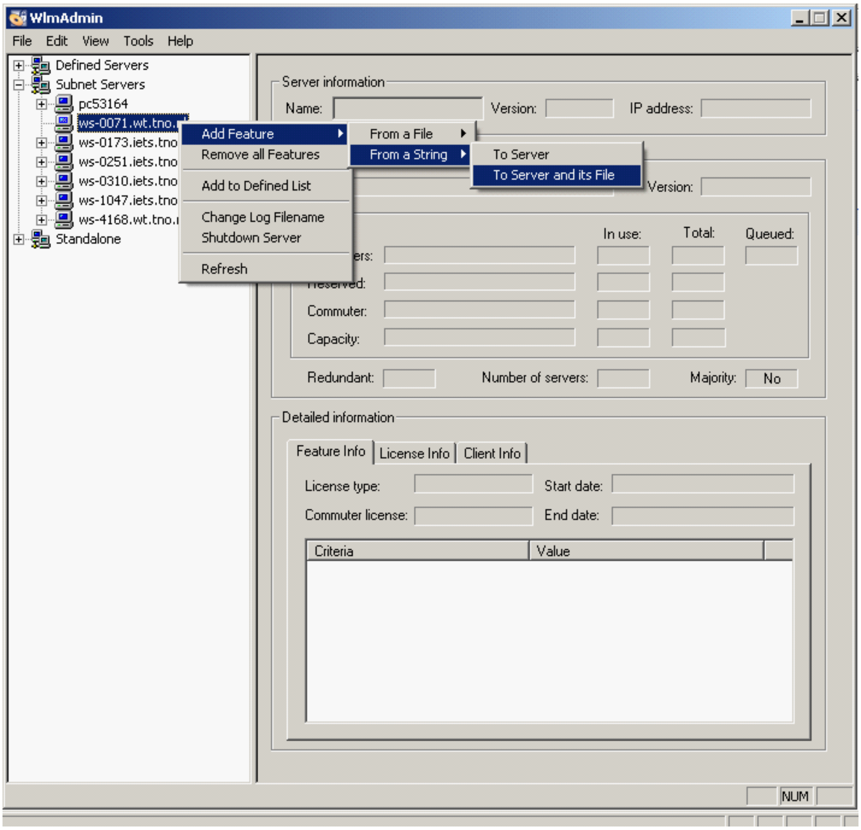
Activating the License

You have now obtained a license file from TNO and want to activate it. To activate the license, run **License Management** from the **Licensing 8.1.0** menu. Open **Subnet**

Severs by clicking the '+'-sign next to it, and you will get a list of license servers on the subnet. If you installed the **Sentinel License Manager** as described in Installing the License Manager 16 your machine should be in the list.

Right click on the name of your machine, and from the pop-up menu that appears select: **Add Feature** => **From a File** => **To Server and its File**, as shown below.

Select the license file that you received from your TNO representative. The program will confirm the features that are added by showing a window for each feature. Just click on the OK-button for each window. Now, the license is activated. The License Management (WlmAdmin) can be closed and your Delft-Tyre application can be run.



Miscellaneous

To check which features are installed, go to your machine and click on the "+" to show the features available.

Available license features

The following license features exist:

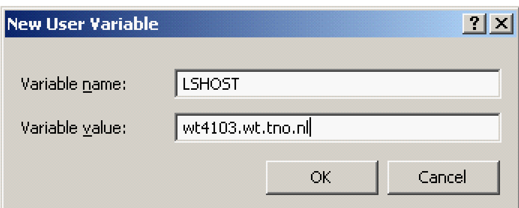
- MFToolSA — MF-Tool GUI (stand-alone)
- MFToolFl — MF-Tool GUI (floating)
- MFFitSwiftFitSA — fit software (stand-alone)
- MFFitSwiftFitFl — fit software (floating)
- MFSwiftTyreModelSA — MF-Swift (stand-alone)
- MFSwiftTyreModelFl — MF-Swift (floating)

2.3.2.5 Install clients

This step is **required** for:

- Floating license: Client side

To be able to reach the license server, on the client computer(s) the **environment variable LSHOST** has to be set and should refer to the license server. To do this, open the “Environment Variables” menu of your Windows installation (right click “My Computer”, select “Properties”, select the “Advanced” tab). Press the “Environment Variables” button and create a new User Variable. The variable name must be LSHOST and the variable value must be the server name or IP address, see figure below.



When this environment variable has been set, the client can use the floating licenses on the server.

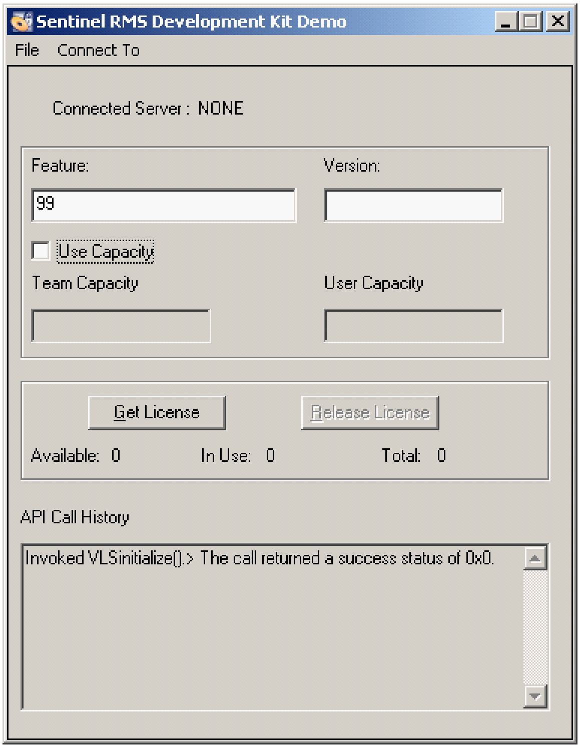
This procedure should be repeated for all client computers, where the TNO Delft-Tyre software (MF-Tool, MF-Tyre/MF-Swift) will be used in the future.

2.3.2.6 Testing the license system

To test the license system without the specific tyre software, go to the folder `<TNO Delft-Tyre>\Licensing 8.1.0` and run:

```
slmdemo.exe
```

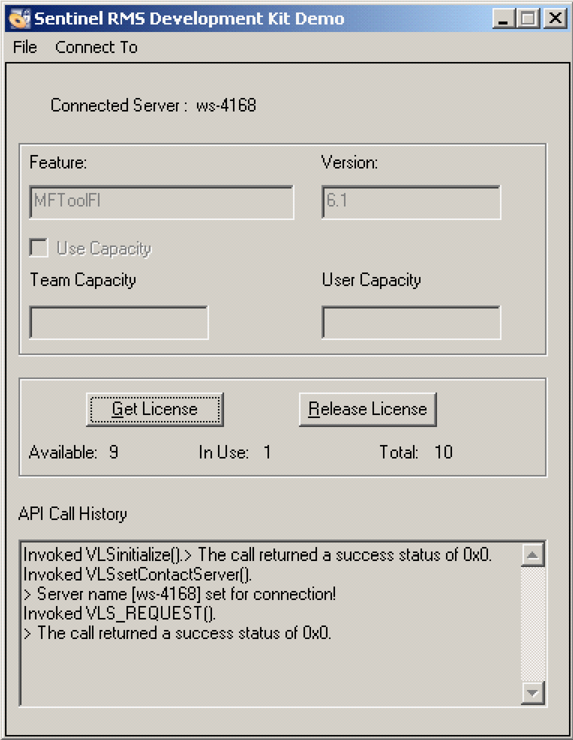
The following window appears:



In the menu use “Connect to” to specify the license server name. To test for example the MF-Swift 6.1.2 license type MFSwiftTyreModelSA in the Feature box and 6.1 in the Version box. For other license types see Available License Features 19.

By pressing “Get License” you can obtain a license for the MF-Swift feature; pressing “Release License” will give the license back to the server.

The figure below shows an example. In this example the total number of available licenses on the server is 9 and 1 license is successfully obtained by pressing “Get License”.



WlmAdmin

By using the License Management (WlmAdmin) 17 one can also check that a license is obtained or released.

> **Note**: In WlmAdmin you have to refresh the list to see changes taking effect.

2.3.2.7 Common License Issues

This chapter describes the most common issue regarding licensing, and their solutions

No license on host no-net

Full error

```
Sentinel RMS Development Kit: Error[18]: No license is available
for feature MFSwiftTyreModelFl ver 6.1 on host no-net.
```

Explanation

1. Environment variable LSHOST is set to the computer the license server is running on. At simulation, the MF-Tyre/MF-Swift looks for the license server on the network. When no license on the network is found due to e.g. a loss of the network connection or time-out, Sentinel is trying to find a standalone license. When it does not find this standalone license, it will give the error message that no license is found on host no-net.
2. Computer is not connected to the network.

Solution

1. Besides environment variable LSHOST, set variable LSFORCEHOST referring to the license server. Sentinel will now normally find the license server.
2. Connect computer to the network.

Failed to resolve host

Full error

```
Sentinel RMS Development Kit: Error[3]: Failed to resolve the
server host "<server_name>"
```

Explanation

- This error occurs if the network connection is lost during simulation. In principle the simulation does not fail if the license is found again within 60 s. If not the above mentioned error occurs.

Solution

- Contact your network administrator.

2.3.3 Connection to MBS Packages

Implementations by TNO

MATLAB/Simulink (MathWorks) 26
ADAMS (MSC) 22
DADS (LMS) 27

2.3.3.1 ADAMS

> **Note**: To use the TNO Delft-Tyre model in combination with ADAMS a private solver is required. See ADAMS documentation for details.

For each supported version of ADAMS a private solver is shipped with

MF-Tyre/MF-Swift. The directory where these solvers are located, can be found via the Windows “Start” menu (TNO DelftTyre > MFTyre & MF-Swift 6.1.2 > ADAMS > Private solvers). In this directory:

- `private_solverXX.dll` is an extension of the standard ADAMS solver
- `acar_solverXX.dll` is an extension of the ADAMS/Car solver

where XX refers to the ADAMS version.

Installation and MF-Swift license verification

> **Note**: At this point it is assumed the license system of MF-Tyre/MF-Swift is working correctly, see Testing the license system 19 for details.

To install and check the licensing mechanism, use the “Start” menu and select: TNO Delft-Tyre > MFTyre & MF-Swift 6.1.2 > ADAMS > Verify installation. In this directory a number of batch files can be found. Depending on the version of your ADAMS installation, select the corresponding batch file:

```
testsolversXX.bat
```

When executing the batch file, a short simulation using the MF-Swift tyre model is performed with both the standard ADAMS solver and ADAMS/Car solver. The simulations should complete without errors, this can be checked by looking at the ADAMS message files (private_solver.msg and acar_solver.msg respectively).

> **Warning**: When running the `testsolversXX.bat` the `acar_solver.dll` in `%HOMEDRIVE%%HOMEPATH%\acar_private\acar_solver.dll` will be overwritten.

Using the private solver in ADAMS/Car

ADAMS/Car expects that the private solver is located in a special folder and that the private solver has a fixed name. The location typically is:

```
%HOMEDRIVE%%HOMEPATH%\acar_private\acar_solver.dll
```

Example: `C:\Documents and Settings\<username>\acar_private\acar_solver.dll`

Once the file `acar_solver.dll` is found at this location, it will be used in the current and future simulations with ADAMS/Car.

You can also check this by looking at the ADAMS message file (*.msg) after doing a simulation. Starting at about line number 170 you will see the message:

```
============================================
==                                        ==
==      Copyright 1996-2010               ==
==      TNO Automotive                    ==
==      P.O. Box 756                      ==
==      5700 AT Helmond                   ==
==      The Netherlands                   ==
==      www.delft-tyre.nl                 ==
==                                        ==
============================================
DELFT-TYRE MF-Tyre/MF-Swift 6.1.2
============================================
```

This confirms that the TNO tyre model is being used. Once you have run the `testsolversXX.bat` file successfully, there is no need to copy the `acar_solverXX.dll` to the special location by hand. However, if for any reason you have to copy the solver dll file by hand to the special location, please rename the file so that its name is `acar_solver.dll` (so remove the ADAMS version specific information).

Warning: When running the `testsolversXX.bat` the `acar_solver.dll` in `%HOMEDRIVE%%HOMEPATH%\acar_private\acar_solver.dll` will be overwritten.

Note 1: For introducing a MF-Tyre/MF-Swift model in your ADAMS model, see ADAMS 87.

Note 2: Two simple demo models can be found under the "Start" menu, `<TNO Delft-Tyre> > MFTyre & MF-Swift 6.1.2 > ADAMS > Demo`. The models cover vehicle handling using MF-Tyre and ride using MF-Swift. You can use these models with ADAMS/View and the standard ADAMS/Solver.

Using the private solver with the standard ADAMS/Solver

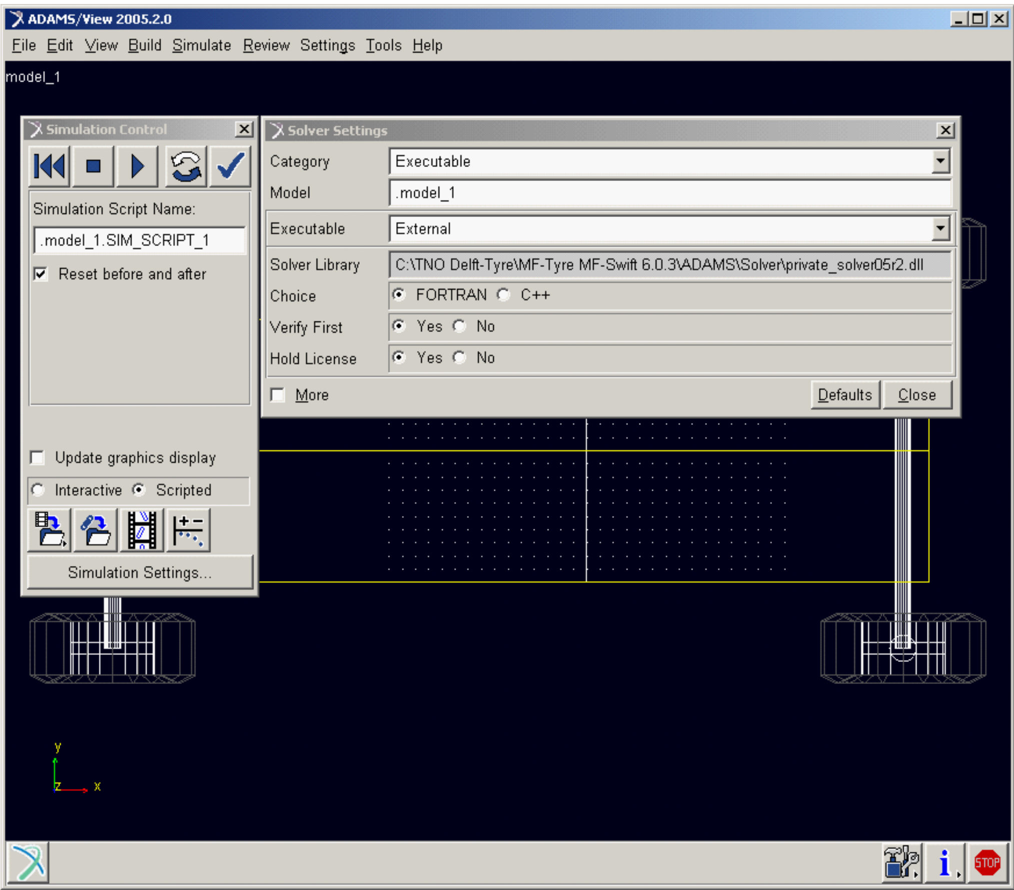
To start a simulation from the DOS prompt, using ADM/ACF files and in this case ADAMS 2005r2, use the following command:

```
adams05r2 ru-us private_solver05r2 mymodel.acf exit
```

In this case, the file `private_solver05r2.dll` should be in the same directory as the ADM/ACF files or the full path to the private solver dll should be specified, e.g.:

```
adams05r2 ru-us c:\mysolvers\private_solver05r2.dll mymodel.acf
exit
```

The private solver dll can also be called from within ADAMS/View. In ADAMS/View select: Simulate > Scripted Controls and then push “Simulation settings” button. In the box “Solver Settings” select “Executable” in the “Category” drop down menu. Then select “External” in the “Executable” box. Finally in the “Solver Library” box, browse to the location of the private solver dll. After that you can close the Solver Settings window and the simulation can be started. Please note that a simulation script is required when using an external executable. The settings are also shown in the figure below.



From the “Settings” menu you can use the “Close” and “Default” buttons to save or restore the changes made to the solver executable selection.

Note 1: For introducing a MF-Tyre/MF-Swift model in your ADAMS model, see ADAMS 87.

Note 2: Two simple demo models can be found under the “Start” menu, `<TNO Delft-Tyre> > MFTyre & MF-Swift 6.1.2 > ADAMS > Demo`. The models cover vehicle handling using MF-Tyre and ride using MF-Swift. You can use these models with ADAMS/View and the standard ADAMS/Solver.

Building your own solver

Next to the Delft-Tyre model, you may want to add other user subroutines to the ADAMS solver. In that case you will have to build a private ADAMS solver yourself. The following components (object and library) are required for the TNO Delft-Tyre model, and may be found in the folder
`<TNO-DelftTyre>\MF-Tyre/MF-Swift_6.1.2\ADAMS\Source\AdamsXX`:

- `tyr815.obj`
- `dtemsg.obj`
- `troad.obj`
- `TNO_DelftTyre.lib`

An example build script is available per Adams version (e.g. link_acarXX.bat). These scripts may need to be modified to reflect your specific needs.

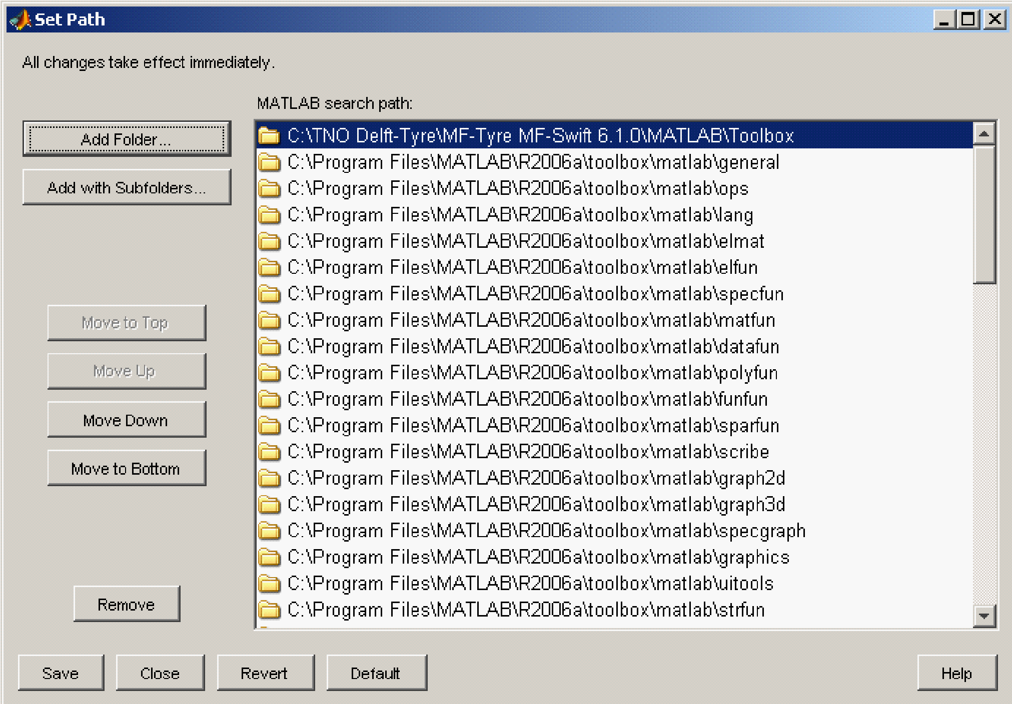
Note 3: Objects and Library for Adams03, Adams05 and Adams05r2 are compiled in Intel Fortran 6.6B; Objects and Library for Adams07r1 and Adams08r1 are compiled in Intel Fortran 9.1.

Note 4: TNO_DelftTyre.lib requires the following dlls: TNO_DelftTyre.dll, TNO_UserRoad.dll, dformd.dll, dforrt.dll, libifcoremd.dll and libmmd.dll. These dlls are part of the MF-Tyre/MF-Swift installation.

2.3.3.2 MATLAB

To use the TNO tyre model MF-Tyre/MF-Swift with MATLAB, Simulink and SimMechanics, the "Toolbox" directory has to be on the MATLAB search path. Typically this directory has the name:

```
C:\TNO Delft-Tyre\MF-Tyre MF-Swift 6.1.2\MATLAB\Toolbox
```



The following steps should be taken:

- From the MATLAB menu select:
  File > Set Path…
- A new window showing the MATLAB search path will appear (see figure above). In this window click on "Add Folder…"
- Browse for the Toolbox directory, followed by "ok" in the "Browse For Folder" window.
- Finally in the "Set Path" window click on "Save" and "Close". From now on the TNO Delft-Tyre model is available in every MATLAB session.

Some examples of using the tools are given in the Start Menu: All Programs > TNO Delft-Tyre > MF-Tyre & MF-Swift 6.1.2 > MATLAB > Simulink demo's (and SimMechanics demo's, respectively).

2.3.3.3 DADS

To use the full potential of the TNO tyre model MF-Tyre/MF-Swift in combination with DADS 9.6 three items are needed:

1. Customized solver: ndads3d.exe (required)

This executable can be found under the "Customised solver" menu item. The file manager shows the directory, where the file ndads3d.exe is placed by default after installation. Note when you want to use the TNO tyre models in combination with your own DADS simulation models, the file `ndads.exe` has to be copied to the directory from which the solver is called. If you are unsure if the TNO software is actually being used, check the *.inf file. In the header you should find a message like:

```
============================================
==                                        ==
==      TNO Automotive                    ==
==      P.O. Box 756                      ==
==      5700 AT Helmond                   ==
==      The Netherlands                   ==
==      www.delft-tyre.com                ==
==                                        ==
==  (c) Copyright 1996-2010               ==
==                                        ==
============================================
DELFT-TYRE MF-Tyre/MF-Swift 6.1.2
===========================================
```

Also when a vehicle appears to fall through the road surface this is generally an indication that the TNO tyre model is not used.

The ndads3d.exe delivered by TNO may not reflect the latest patch level of DADS, or you may want to add other user routines to the DADS solver. In this case you have to build the customised ndads3d solver yourself. For the tyre model the following items need to be linked: dtyre.obj, dtemsg.obj and TNO_DelftTyre.lib. A makefile (makefile.pc) and build script (build_ndads.bat) are included, but they may need modifications to reflect your specific needs.

2. Customisation of the tyre GUI (file: tirsti.pdf, optional)

The menu used for introducing the tyre model in DADS can be customised, which makes the TNO tyre model more intuitive to use. To achieve this, the file "tirsti.pdf" has to be copied into the installation directory of DADS, typically `C:\Program Files\DADS 9.6\dadsaux\def\`. Please rename the existing file tirsti.pdf first. Note that, despite the extension, this is not a regular binary Adobe pdf file.

3. Customisation of the plot GUI (file: tire3d_875.xdf, optional)

The TNO tyre model provides more and different outputs compared to the regular DADS tyre models. To get a clear overview of the signals available for plotting, the file tire3d_875.xdf has to be copied to the installation directory of DADS, typically:

```
C:\Program Files\DADS 9.6\dadsaux\post\xdf\
```

Again please rename the existing file tire3d_875.xdf first.

Not-customized Plot GUI

In the following table a translation list can be found when the plot GUI is **not** customized.

Name in the plot GUI	TNO Delft-Tyre output signal
"fx_chassis_body"	longitudinal tyre force Fx
"fy_chassis_body"	lateral tyre force Fy
"fz_chassis_body"	vertical tyre force Fz
"tx_chassis_body"	overturning moment Mx
"ty_chassis_body"	rolling resistance moment My
"tz_chassis_body"	self-aligning moment Mz
"fmg_chassis_body"	force magnitude
"tmg_chassis_body"	moment magnitude
"defl"	longitudinal slip (kappa)
"strs"	side slip angle (alpha)
"rlls"	camber angle (gamma)
"camb"	turnslip
"a"	forward velocity Vx
"fn"	- not used -
"flng"	effective rolling radius
"flat"	vertical tyre deflection
"fmag"	contact length
"fmax"	pneumatic trail
"almt"	friction coeff longitudinal direction
"frr"	friction coeff lateral direction
"fx_tire_body"	relaxation length longitudinal
"fy_tire_body"	relaxation length lateral
"fz_tire_body"	Vsx
"tx_tire_body"	Vsy
"ty_tire_body"	vertical deflection velocity
"tz_tire_body"	- not used -
"unx"	kappa dynamic

"uny"	alpha dynamic
"unz"	- not used -
"vn"	travelled distance
"vlng"	- not used -
"vlat"	- not used -
"tau"	x coordinate contact point
"v32"	y coordinate contact point
"v33"	z coordinate contact point
"v34"	x road normal
"v35"	y road normal
"v36"	z road normal
"v37"	effective plane height
"v38"	effective plane angle
"v39"	effective plane curvature
"v40"	effective banking angle

2.4 Linux Installation

To install MF-Tyre/MF-Swift unpack the file MFTyre_MFSwift_611_xxx.tar.gz:

```
tar -tzvf MFTyre_MFSwift_611_xxx.tar.gz
```

A directory structure is created.

ADAMS installation

The files for ADAMS can be found in the Adams subdirectory.

Demo

`Demo` contains demo models to be used with ADAMS/View

Solver_*

The folder `Solver_Linux32` contains the files for the private solvers.

The naming convention of the solvers follows the different Adams releases. For example:

- `private_solver05r2.exe` is the private solver to be used with ADAMS/View 2005r2
- `libacar_solver05r2.sl` is the private solver for ADAMS/Car 2005r2.

The object files (*.o) and libary (*.a) allow to create your own private solver. Typically the commands to do that are:

Private solver:

```
adams05r2 -c cr-user n @objects.lst -n private_solver05r2.exe
exit
```

ADAMS/Car solver:

```
adams05r2 -c acar  cr-solverprivate n @objects.lst -n exit
```

Note: When you create an ADAMS/Car private solver it will be placed in a special directory:

```
$HOME/acar_private/rh_linux/libacar_solver.so
```

Note: From these solver names it is unclear for which version of ADAMS it was build.

Verify

The Verify directory contains some sample scripts to check the installation.

For example the command:

```
./test_adams05r2_Linux32.sh
```

will test the 2005r2 version of the private solvers for ADAMS and ADAMS/Car.

Note: This script automatically copies the ADAMS/Car solver to the right directory.

Licensing

When you use MF-Tyre 37 only, no license is required. When using rigid ring dynamics or short wavelength road contact a MF-Swift 38 license is required. This license can be obtained from TNO Automotive. The license is node locked, but one license file can contain entries for different computer systems.

For the licensing 2 steps are required:

1) Obtain the MachineID

Run one of the test scripts in the **Verify** directory. This will start a short ADAMS simulation. For example:

```
./test_adams05r2_Linux32.sh
```

The Computer ID is written in the ADAMS message file. You can obtain it with the following Unix command:

```
grep MachineID *.msg
```

This will return lines which include the MachineID, for example:

```
private_solver.msg:       MachineID: tecra8-8323328
```

This line should be send to TNO by E-mail. TNO can then generate a license file for you. If you want to run the model on different computers, first collect the Computer ID of the different machines.

2) Install the license file

You have obtained a license file from TNO, e.g. TNO_2008.lic

This file should be placed at a location where it can be seen from the machines running ADAMS. Futhermore the environment variabele `DT_LICENSE_FILE` should be set, which points to this license file. On **Linux** add the following line to the file `.bashrc` :

```
DT_LICENSE_FILE="/home/,username>/TNO_2007.lic";export
DT_LICENSE_FILE
```

3 Technical Support

Support is provided to those who have a support contract.

Categories

Support categories (among others):

- Bugs
- Request for enhancements / new features
- Installation help
- Application help

Supported multi-body packages

TNO provides (direct) support for the following multi-body packages

- ADAMS
- MATLAB / Simulink / SimMechanics
- DADS

For other multi-body packages support can be requested at the respective multi-body package used. If required the multi-body package will contact TNO to solve the issue. Requests for enhancements and / or new features may also be asked directly.

Checklist

Before asking support, make sure you followed all steps as described in the Simulation Guidelines 41.Especially step 9), checking model messages, is encouraged to carry out.

Required information

To be able to provide you support, we need to be able to reproduce your issue. When asking support, please provide the following:

- MF-Tyre/MF-Swift and multi-body package release number
- Brief summary of the issue
- Reproduction path: what steps did you take for the issue to occur
- Error message: exact error message, preferably complete text or screenshot
- If allowed: Models* used for the issue to occur (don't forget the include files)

* **Note**: We understand your hesitation to share your models with us, since they may contain **property knowledge** or project / company **restricted information**. TNO will always treat your information as confidential unless otherwise stated.

Support

Finally, send your support request to the following mail address, or your current contact at the Delft-Tyre group:

Mail: Support-DelftTyre@tno.nl

3.1 Contact Information

Other contact information.

Sales

Contact information can be found on our website or may be reached by e-mail.
Website www.delft-tyre.nl
Mail DelftTyre@tno.nl

Support

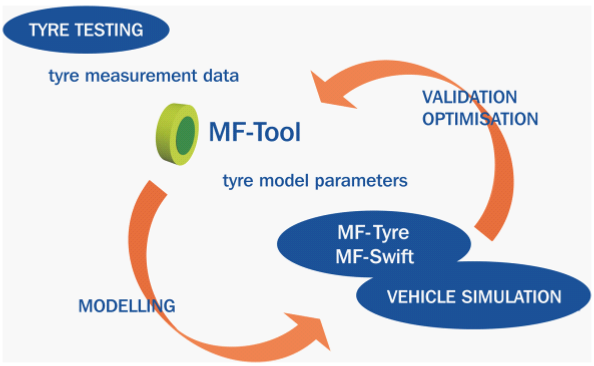
Support may be reached by e-mail:
Mail: Support-DelftTyre@tno.nl

4 User Manual

This chapter is the User Manual of MF-Tyre/MF-Swift.

4.1 Introduction

The contact interaction between tyres and the road largely affects the driving performance of vehicles. Automotive engineers are optimising the tyre-road interaction so that the vehicle handles well and operates both safely and comfortably under any circumstance. To analyse the influence of tyre properties on the dynamic behaviour of vehicles, the engineer requires an accurate description of the tyre-road contact phenomena. TNO Delft-Tyre provides a complete chain of tools and services for detailed assessment and modelling of vehicle-tyre-road interaction.



TNO Delft-Tyre chain of tools for tyre analyses.

The tyre models MF-Tyre 37 and MF-Swift 38 can be used in vehicle dynamics simulations in all major simulation packages to efficiently and accurately represent tyre behaviour for applications ranging from steady-state to complex high frequency dynamics. MF-Tyre and MF-Swift contain the latest implementation by Delft-Tyre of Pacejka's renowned 'Magic Formula' tyre model.

With MF-Tyre you can simulate steady-state and transient behaviour, making it a suitable tyre model for:

- vehicle handling,
- control prototyping, or
- rollover analysis.

With MF-Swift you can simulate tyre dynamic behaviour up to about 100 Hz, which is particularly useful for

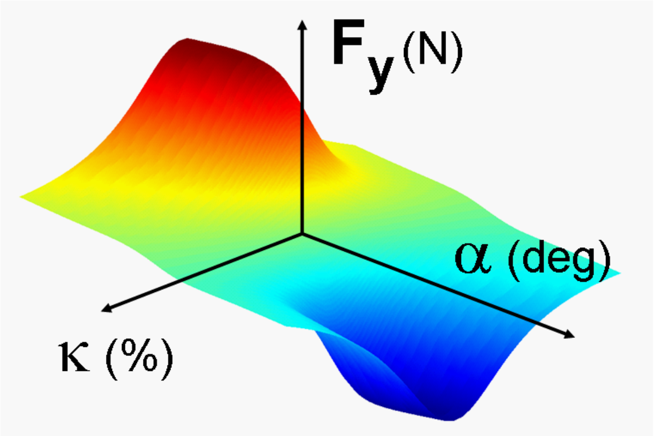
- vehicle comfort,
- durability,
- vehicle control prototyping, or
- vibration analysis.

Special attention has been paid to include behaviour necessary for special applications such as motorcycles (regular and racing), motorsport (e.g. Formula 1) or aircraft tyres.

TNO Delft-Tyre's MF-Tyre and MF-Swift are available for all major simulation packages, see Compatibility Table 10. TNO Delft-Tyre makes sure that the tyre model implementation and simulation results are identical and that the same set of tyre model parameters can be used for all these packages. Further, MF-Tyre and MF-Swift are fully compatible with all previous 'official' TNO Delft-Tyre releases, see Backward compatibility 56.

4.1.1 MF-Tyre

MF-Tyre is TNO Delft-Tyre's implementation of the world-standard Pacejka Magic Formula tyre model, including the latest developments by TNO and Prof. Pacejka [1] and [2]. MF-Tyre's semi-empirical approach enables fast and robust tyre-road contact force and moment simulation for steady-state and transient tyre behaviour. MF-Tyre has been extensively validated using many experiments and conditions. For a given pneumatic tyre and road condition, the tyre forces and moments due to slip follow a typical characteristic. These steady-state and transient characteristics can be accurately approximated by MF-Tyre.



Steady –state tyre lateral force as function of longitudinal and lateral slip, calculated using MF-Tyre.

MF-Tyre calculates the forces (Fx, Fy) and moments (Mx, My, Mz) acting on the tyre under pure and combined slip conditions on arbitrary 3D roads, using longitudinal, lateral and turn slip, wheel inclination angle ('camber') and the vertical force (Fz) as input quantities.

MF-Tyre is valid for large slip angles (typically over 30 degrees), longitudinal slip (+/-

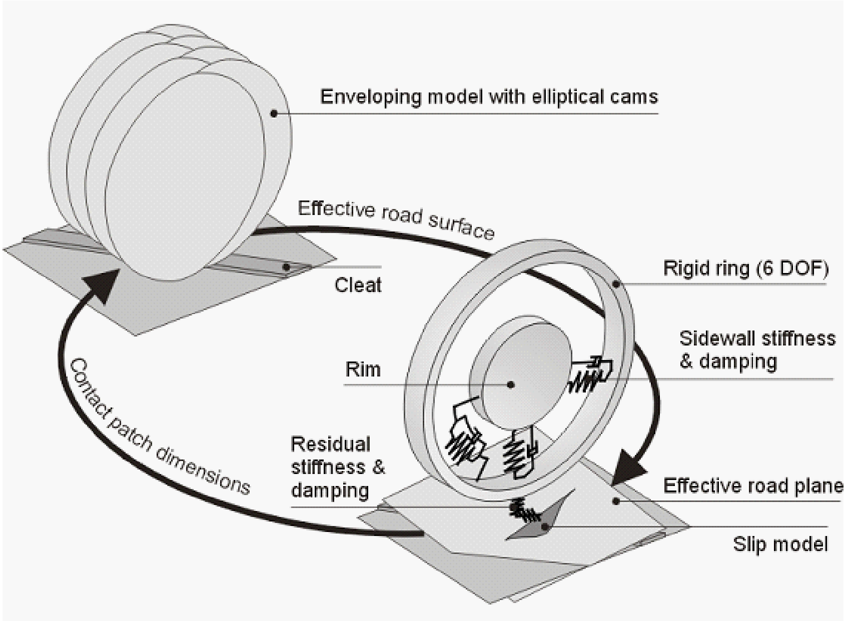
100%), large load variations (including truck tyre loads) and large camber angles (including motorcycle camber angles; MF-Tyre 6.x includes the functionality of MF-MCTyre). It can handle road undulations that have a wavelength larger than the tyre circumference and is typically applied for vehicle handling simulation.

4.1.2 MF-Swift

In addition to the Magic Formula description in the MF-Tyre 37 part of the model, **MF-Swift** uses a rigid ring model in which the tyre **belt** is assumed to behave like a rigid body. This means that the model is accurate in the frequency range where the bending modes of the tyre belt can be neglected, which, depending on the tyre type, is up to 60 – 100 Hz. MF-Swift has been validated using measurements of a rolling tyre (7 to 40 m/s) containing frequencies up to 120 Hz. The model includes essential gyroscopic effects.

The tyre model functionality is primarily based on [1] to [6]. TNO has made several crucial changes and enhancements in cooperation with Prof. Pacejka to the models as described in [1] in order to improve functionality, robustness, calculation times, user-friendliness and compatibility between various operating modes.

MF-Swift uses an efficient single point contact for slip calculation which results in full compatibility with MF-Tyre. Due to the introduction of a so-called phase leading network for the pneumatic trail, MF-Swift is suitable for path curvature with a wavelength in the order of two times the contact length. For braking/traction applications, wavelengths as small as half the contact length are well described. The transient slip behaviour is well described up to full sliding, due to modelling of decrease in relaxation length for increased slip levels.



Graphical representation of the MF-Swift model.

Five main elements of the model structure can be distinguished:

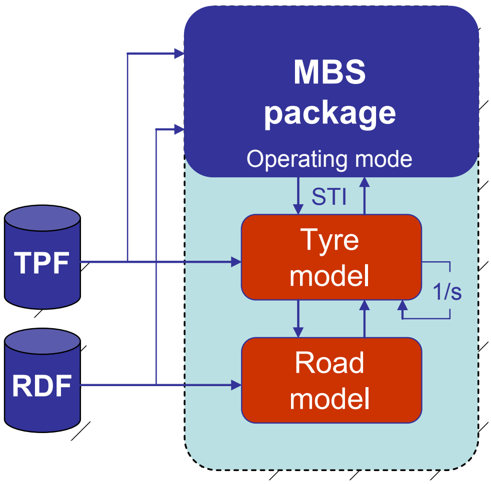
1. Elastically suspended **rigid ring** (6 degrees of freedom): represents the tyre sidewalls and belt with its mass and inertia properties. The rigid ring describes the primary vibration modes of the tyre belt.
2. Residual stiffness & damping: have been introduced between contact patch and rigid ring to ensure that the total quasi-static tyre stiffnesses in vertical, longitudinal, lateral and yaw directions are modelled correctly. The total tyre model compliance is made up of the carcass (ring suspension) compliance, the residual compliance (in reality a part of the total carcass compliance) and the tread compliance.
3. Contact patch model: features horizontal tread element compliance and partial sliding. On the basis of this model, the effects of the finite length and width of the footprint are approximately included.
4. Generic 3D obstacle enveloping model: calculates effective road inputs to enable the simulation of the tyre moving over an uneven road surface with the enveloping behaviour of the tyre properly represented. The actual three-dimensional profile of the road is replaced by a set of four effective inputs: the effective height, the effective forward and camber slopes of the road plane and the effective forward road curvature (that is largely responsible for the variation of the tyre effective rolling radius).
5. Magic Formula steady-state slip model (MF-Tyre): describes the nonlinear slip force and moment properties. This enables an accurate response also for handling manoeuvres.

For more details on the MF-Swift tyre model, please refer to [1] and [6].

4.2 Model Usage

Tyre Simulation

MF-Tyre/MF-Swift is a plug-in to Multi-body simulation packages (MBS) 87. The communication between the MBS Package and the tyre model during simulation is presented below.



The MBS package is communicating with the tyre model following the Standard Tire Interface format [7 115]. The tyre model in its turn is communicating with the Road model. The multi-body simulation package, and the Tyre and Road model are fed by the Tyre Property File (TPF) 53 and Road Data File (RDF) 74. The MBS package specifies the Operating mode 45 of the tyre model.

To set up a simulation in a MBS package using MF-Tyre/MF-Swift, the user is advised to follow the Simulation Guidelines 41.

Note: It is assumed that a user of MF-Tyre/MF-Swift is familiar with the use of a Multi-body simulation package.

4.2.1 Simulation Guidelines

The following steps are recommended to take when setting up a vehicle dynamics simulation with MF-Tyre/MF-Swift:

1) Add the TNO Delft-Tyre model to your simulation model, see chapter Multi-Body Simulation Packages 87 (MBS) for details.

2) Set Tyre Property File 53

3) Select road surface 74 (MBS, TNO, UserRoad)
   a) Set the Road Data File 74

4) Select the correct operating mode 45 (ISWITCH 48) for the tyre model
   a) Advised operating modes (format: ABCD):
      i) 0104 for steady-state MF-Tyre simulations only (e.g. steady-state circling)
      ii) 0124 or 114 for general MF-Tyre simulations (e.g. slalom, ISO lane change, J-turn)
      iii) 0134 for MF-Swift on flat or smooth road surfaces (e.g. shimmy)
      iv) 0434 for MF-Swift on 2d road surfaces (e.g. ride comfort and durability simulations)
      v) 0534 for MF-Swift on 3d road surfaces (e.g. ride comfort and durability simulations)

5) Set the correct mounting side 45 of the tyre model, all left tyres must be specified as left and all right tyres must be specified as right in the MBS Packages.

6) Masses and inertia 50:
   a) MF-Tyre: tyre model is massless; mass and inertia of tyre have to be added to rotating wheel body in MBS program;
   b) MF-Swift: tyre model contains mass and inertia of tyre belt; mass and inertia of tyre except mass and inertia of belt have to be added to rotating wheel body in MBS program.

7) When using a short wavelength road contact method 45:
   a) 2d contact method (B =4):
      i) Set ROAD_INCREMENT properly. Typically use a road increment that corresponds with the sample interval of the measured road profile.
      ii) Set ROAD_DIRECTION properly.
   b) 3d contact method (B = 5):
      i) Set ROAD_INCREMENT 46 properly. Typically use a road increment that corresponds with the sample interval of the measured road profile.
      ii) Set the value of ELLIPS_MAX_STEP 46 sufficiently larger than the obstacle height for discrete obstacle impacts or larger than the difference in height of two neighbouring road points for arbitrary/ measured road profiles. For arbitrary road surfaces always check for error and warning messages like:
         (1) 'Too many ellipse points, increase parameter ROAD_INCREMENT, or reduce ELLIPS_MAX_STEP'
         (2) 'Road data may require to extend ELLIPS_MAX_STEP'
      iii) If these messages occur please change the values of ROAD_INCREMENT and ELLIPS_MAX_STEP according to suggestion

provided in the error or warning message till these messages disappear.

iv) The number of elliptical cams in the contact can be specified with the parameters ELLIPS_NWIDTH and ELLIPS_NLENGTH. The default settings for ELLIPS_NWIDTH and ELLIPS_NLENGTH are respectively 10 and 10. This should be sufficient for sharp obstacles like oblique cleats. The number of ellipses can be increased or decreased depending on the simulation requirements (accuracy vs. computational effort).

> **Note**: The minimal distance between the contact points on the cams is 1 mm. Thus for highest accuracy, set the *ROAD_INCREMENT* to 1 mm for simulations on sharp obstacles.

8) Set the simulation time step.
   a) Variable-step solver: solver will determine correct time step itself
   b) Fixed-step solver: user needs to determine the time step so that the simulation (tyre and vehicle) stays stable. Typical values for each Dynamics mode 47 are:
      i) Steady-state, relaxation behaviour, linear: 10^{-2} - 10^{-4}
      ii) Relaxation behaviour, non-linear: 10^{-3} - 10^{-5}
      iii) Rigid Ring: 10^{-3} - 10^{-5}

9) Perform a short simulation and check the tyre model messages.
   a) Is the TNO tyre model used?
   b) Are indeed the correct tyre property file, road surface, use mode, etc. used?
   c) Are there any error or warning messages related to the tyre model?
   d) Was the vehicle positioned correctly on the road? Are the vertical tyre forces correct?

4.2.2 Dynamics Mode

Depending on the frequency range of interest, the **Dynamics Mode** of the Delft-Tyre model family may be chosen. With increasing frequency, more details in the dynamic behaviour of the tyre are included. Five modes exist for the Delft-Tyre model, separated over the two components:

- MF-Tyre 37
- MF-Swift 38

> **Note**: For MF-Swift a license is required. For MF-Tyre, it is not.

MF-Tyre/MF-Swift Operating Modes

The **Dynamics Modes** are:

Dynamics mode	Frequency range	Component
Steady-state	< 1 Hz	MF-Tyre
Relaxation behaviour, linear	< 10 Hz, linear	MF-Tyre
Relaxation behaviour, non-linear	< 10 Hz, non-linear	MF-Tyre
Rigid ring	< 100 Hz, non-linear	MF-Swift (requires license)
Rigid ring + initial statics	< 100 Hz, non-linear*	MF-Swift (requires license)

* Same as Rigid Ring, but with finding static equilibrium at the start of the simulation

Steady-State

In the case of a steady-state evaluation no dynamic behaviour is included.

Relaxation Behaviour, linear

"Linear transient effects" indicates that the tyre relaxation behaviour is included using empirical relations for the relaxation lengths.

Relaxation Behaviour, non-linear

In the "Nonlinear transient effects" mode, a physical approach is used in which the compliance of the tyre carcass is considered to determine the lag. This approach correctly accounts for the tyre property that the lag in the response to wheel slip and load changes diminishes at higher levels of slip. This approach is fully compatible with the MF-Swift theory.

Rigid Ring Dynamics

"Rigid ring dynamics" refers to a detailed dynamic model (MF-Swift), where the tyre belt is modelled as a separate rigid body.

Rigid Ring Dynamics with Initial Statics

Finally, "initial statics" refers to finding the static equilibrium of the tyre belt (rigid ring/body) at the start of the simulation.

Note: When selecting rigid ring dynamics the belt is not massless anymore. The mass and moments of inertia of the tyre belt have to be subtracted from the total wheel + tyre mass and moments of inertia.

4.2.3 Tyre model operating modes

The main operating modes of the MF-Tyre/MF-Swift model are described below. This information should always be provided by the Tyre Property File or by the MBS package 87.

MF-Tyre/MF-Swift is set up in a modular way and allows a user to independently set the operating mode of the Magic Formula, tyre dynamics and contact method.

A: Tyre side - Magic Formula mirroring

If a tyre has asymmetric behaviour, caused by e.g. conicity or plysteer, using the same characteristics on the left and right hand side of a vehicle will result in incorrect results. For this reason the side of the vehicle a tyre is mounted on should be specified.

We may select one of the following values for the tyre side in the **MBS Package**:

- 0/1 tyre is mounted on the left side of the car
- 2 tyre is mounted on the right side of the car
- 3 symmetric tyre characteristics (asymmetric behaviour is removed)

In the **Tyre Property File**, it should be specified how the tyre measurement was executed: in other words, if the left or right tyre was tested. In the Tyre Property File [MODEL]-section, the keyword TYRESIDE can be set to either "LEFT" or "RIGHT" (when missing: "LEFT" is assumed).

If "TYRESIDE" is "LEFT" and the tyre is mounted on the right side of the vehicle (A=2), mirroring will be applied on the tyre characteristics and the total vehicle will behave symmetrically. It is also possible to remove asymmetrical behaviour from an individual tyre (A=3).

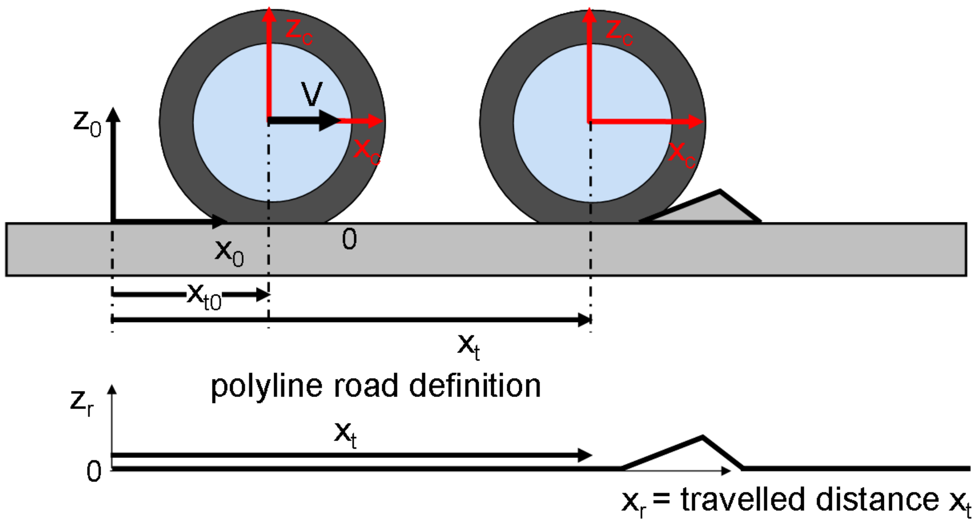
B: Contact Method

Various methods are available to calculate the tyre–road contact point. Smooth road contact should only be used on a smooth road surface profile containing a minimum wavelength larger than twice the tyre radius. For short obstacles (e.g. cleats/bumps, discrete steps, potholes) or road surfaces containing wavelength smaller than twice the tyre radius, either the road contact for 2D or 3D roads should be selected. The road contact for 3D roads works on both 2D and 3D road surfaces, but it is computationally more expensive than the road contact for 2D roads that works only with 2D road profiles. The moving road is to be used for simulation of a four poster test rig. It is available in a limited number of simulation packages (e.g. MATLAB/Simulink, SIMPACK 8.700 and up)

The following values may be selected for the contact method in the **MBS Package**:

0/1 smooth road contact, single contact point
2 smooth road contact, circular cross section (motorcycle tyres)
3 moving road contact, flat surface
4 road contact for 2D roads (using travelled distance)
5 road contact for 3D roads

All contact methods use global coordinates to obtain the road height. Exception is the road contact for 2D roads where the travelled distance is used to obtain the road height. The travelled distance is the distance the wheel centre has travelled with respect to the origin of the global axis system (x^0, y^0, z^0). By default, the travelled distance is positive in the direction of the x^c-coordinate system. The travelled distance changes sign when the origin of the global coordinate system is passed.



Road contact for 2D and 3D roads

There are two variables in the road contact that need some special attention:

- Increment of the road sampling: step size with which the contact algorithm is evaluating the road surface
- Maximum allowed discrete step in road height (3D contact only)

The increment of the road sampling is the step size with which the contact algorithm is evaluating the road surface. In general, a smaller road sampling increment will give more accurate results but a slower simulation as more contact points are evaluated. The contact algorithm will try to use at least 3 contact points of the elliptical cams in one road increment. But there are some limitations:

- The distance between the contact points on the cams is at least 1 mm;
- Due to the limited memory space reserved for the calculation of the effective road height and slope, the distance between the contact points may become larger. The tyre model will generate warning messages when a memory problem has occurred. For 2D contact the message “Road increment adjusted to minimum possible increment of: (value) m” appears and for 3D contact the message “Too many ellipse points, increase parameter ROAD_INCREMENT, or reduce ELLIPS_MAX_STEP” appears.

A typical value for ROAD_INCREMENT is 0.01 m, and should be defined in the `[MODEL]` section of the Tyre Property File 53.

```
ROAD_INCREMENT = 0.01
```

The maximum allowed discrete step in road height should be set by the user larger than the highest obstacle in the road surface. It should be set in the Tyre Property File 53 in the `[CONTACT_PATCH]`, e.g.:

```
ELLIPS_MAX_STEP = 0.025
```

For a detailed description of these parameters see the Tyre model settings 58 section.

C: Dynamics

We may select one of the following values for the Dynamics Mode 43:

0 Steady-state evaluation (< 1 Hz)
1 Transient effects included, tyre relaxation behaviour (< 10 Hz, linear)
2 Transient effects included, tyre relaxation behaviour (< 10 Hz, nonlinear)
3 Rigid ring dynamics included (< 100 Hz, nonlinear) *(requires MF-Swift license)*
4 Rigid ring dynamics + initial statics (same as 3, but with finding static equilibrium) *(requires MF-Swift license)*

Known issue: When using a fixed-step solver for **tyre relaxation behaviour (< 10 Hz, nonlinear)**, the time-step of the simulation should be chosen small enough (typically 10^{-5}) for the simulation to produce correct results. A variable-step solver will automatically reduced the time-step when required.

D: Slip forces - Magic Formula evaluation

When evaluating the Magic Formula it is possible to switch off parts of the calculation. This is useful when e.g. debugging a vehicle model, or if only in-plane tyre behaviour is required. The following values may be selected for D:

0 no Magic Formula evaluation (Fz only)

1 longitudinal forces/moments only (Fx,My)
2 lateral forces/moment only (Fy,Mx,Mz)
3 uncombined forces/moment (Fx,Fy,Mx,My,Mz)
4 combined forces/moment (Fx,Fy,Mx,My,Mz)
5 combined forces/moment (Fx,Fy,Mx,My,Mz) + turnslip

Note 1: In principle all combinations are possible, although some make more sense than others. Typically you do not use road contact for 2D or 3D roads without activating rigid ring dynamics. On the other hand you may want to use rigid ring dynamics on a flat road surface e.g. in case of ABS/ESP or shimmy analysis. Obviously the choice of the operating mode will affect the calculation times.

Note 2: Don't forget to specify the mass of the wheel in the multi-body package.

4.2.3.1 ISWITCH

Although most packages use a Graphical User Interface (GUI) to supply the operating mode to the tyre model, on a lower level the operating modes are combined to a single variable called **ISWTCH**, see Standard Tyre Interface for details [7 115]. The following format is used: ISWTCH = ABCD. For example, ISWTCH = 1134 stands for:

- A = 1: left tyre;
- B = 1: smooth road contact, single contact point
- C = 3: Rigid ring dynamics
- D = 4: combined slip forces/moments

Note: In ADAMS 87 the operating mode must be set using the parameter **USE_MODE** in the [MODEL] section of the Tyre Property File.

4.2.4 Supported operating modes

The next table lists the operating modes that are supported by MF-Tyre and MF-Swift licenses.

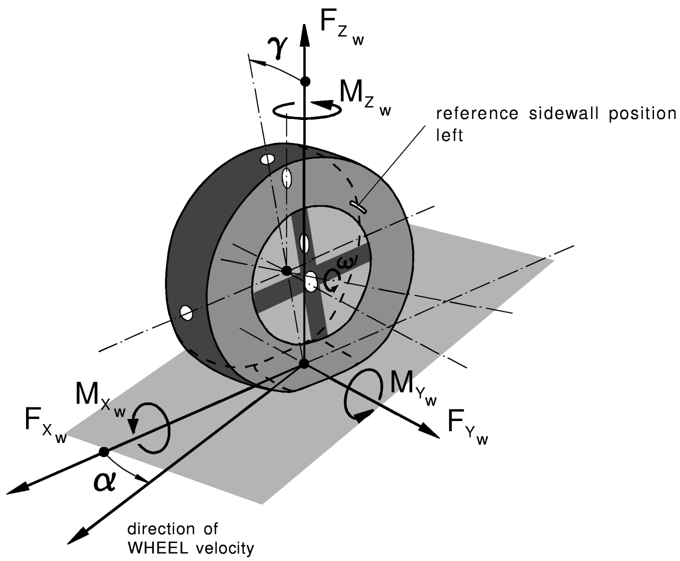
	MF-Tyre	MF-Swift
Slip forces - Magic Formula evaluation (number D)	0,1,2,3,4	0,1,2,3,4,5
Dynamics (number C)	0,1,2	0,1,2,3,4
Contact Method (number B)	0,1,2,3	0,1,2,3,4,5
Tyre side - Magic Formula mirroring (number A)	0,1,2,3	0,1,2,3

4.2.5 Conventions

In this section the axis system and units, used in MF-Tyre/MF-Swift, are explained.

Axis System

MF-Tyre/MF-Swift uses the ISO sign conventions as shown in the figure below.



ISO sign conventions.

The longitudinal slip κ and sideslip angle α are defined as:

$$\kappa = -\frac{V_{sx}}{V_x}$$

, (note: κ = -1 means braking at wheel lock),

$$\tan(\alpha) = \frac{V_{sy}}{|V_x|}.$$

In these equations:

- V_x is the x-component (in the wheel centre plane) of the wheel contact centre horizontal (i.e. parallel to road) velocity V;
- V_s is the wheel slip velocity (with components V_{sx} and V_{sy}), which is defined as the horizontal velocity of the slip point that is thought to be attached to the wheel at a distance that equals the effective rolling radius below the wheel centre in the wheel centre plane.

Units

The output of the tyre model is always in SI units (m, N, rad, kg, s).

The Tyre Property File uses SI units by default (m, N, rad, kg, s); this is always the case when it is generated by TNO Automotive. It is allowed to use a different set of units (e.g. mm or inch for length). The specification in the `[UNITS]` section file applies to all parameters in the Tyre Property File.

The tyre model expects SI units to be passed via the interface between tyre model and the multi-body simulation program, as defined in the specification of the Standard Tyre Interface (STI) [8] 115. However many multi-body codes do not use units internally and leave the choice of a consistent set of units to the user. In many cases this implies that the vehicle model has to be defined using SI units to avoid unit conversion problems.

Mass

An MF-Tyre 37 model does **not** contain any mass, and an MF-Swift 38 model does **only** contain a mass for the belt 38. In the multi-body packages 87 used, the mass should be defined as rim + tyre mass. For MF-Swift the belt mass should be subtracted from this mass. The mass definition is summarized in the table below.

Model	Tyre model mass	Mass definition in MBS Package
MF-Tyre	None	Rim + Tyre
MF-Swift	Belt mass	Rim + Tyre - Belt mass

Note: The Tyre Mass definition in the Tyre Property File is ignored in all multi-body packages, except by the "SimMechanics Wheel + tyre" block provided with MF-Tyre/MF-Swift, in which a complete wheel (including a tyre) is modelled.

4.2.6 Tyre model output

MF-Tyre/MF-Swift is offered as a force element which can be connected to a Multi-body simulation (MBS) package 87.

Various signals are available for post-processing (sometimes called varinf). The availability may be dependent on the implementation in the MBS package. Depending on the implementation they are selected by means of a keyword, signal number or other methods.

tyre contact forces/moments in the contact point:

# Variable	Description	Unit
1 Fx	longitudinal force Fx_w	[N]
2 Fy	lateral force Fy_w	[N]
3 Fz	vertical force Fz_w	[N]
4 Mx	overturning moment Mx_w	[Nm]
5 My	rolling resistance moment My_w	[Nm]
6 Mz	self aligning moment Mz_w	[Nm]

slip quantities:

7 kappa	longitudinal slip kappa	[-]
8 alpha	sideslip angle alpha	[rad]
9 gamma	inclination angle	[rad]
10 phi	turn slip	[1/m]

additional tyre outputs:

11 Vx	wheel contact centre forward velocity	[m/s]
13 Re	effective rolling radius	[m]
14 defl	tyre deflection	[m]
15 contact_length	tyre contact length	[m]
16 tp	pneumatic trail	[m]
17 mux	longitudinal friction coefficient	[-]
18 muy	lateral friction coefficient	[-]
19 sigma_x	longitudinal relaxation length	[m] (not always available)
20 sigma_y	lateral relaxation length	[m] (not always available)
21 Vsx	longitudinal wheel slip velocity	[m/s]

22 Vsy	lateral wheel slip velocity	[m/s]
23 Vz	tyre compression velocity	[m/s]
24 psidot	tyre yaw velocity	[rad/s]
28 s	travelled distance	[m] (not always available)

tyre contact point:

31 xcp	global x coordinate contact point	[m]
32 ycp	global y coordinate contact point	[m]
33 zcp	global z coordinate contact point	[m]
34 nx	global x component road normal	[-]
35 ny	global y component road normal	[-]
36 nz	global z component road normal	[-]
37 w	effective road height	[m] (not always available)
38 beta_y	effective forward slope	[rad] (not always available)
39	effective road curvature	[1/m] (not always available)
40 beta_x	effective road banking/road camber angle	[rad] (not always available)

Note: The wheel spindle forces and moments are in general obtained from the multibody package 87.

4.3 Tyre Property File

The **Tyre Property File** contains the parameters of the tyre model.

- The different **sections** of the Tyre Property File are described in Overview 53.
- If not all parameter are specified, MF-Tyre/MF-Swift has a built-in procedure 54 to estimate the missing parameters.
- Scaling factors 55 may be specified to manipulate and tune tyre characteristics.
- MF-Tyre/MF-Swift is backward compatible 56 for all commercially released versions of it.
- The full set of MF-Tyre/MF-Swift parameters is described in Parameters in the Tyre Property File 63.

- **Sample Tyre Property Files** are provided with the installation in the directory `<TNO Delft-Tyre>MF-Tyre MF-Swift 6.1.2\Tyre property files`.

4.3.1 Overview

The Tyre Property File (*.tir) is subdivided in various sections indicated with square brackets. Each section describes a certain aspect of the tyre behaviour. The next table gives an overview (a full description is given in Parameters in the Tyre Property File 63):

General and Swift parameters:

[UNITS]	units system used for the definition of the parameters
[MODEL]	parameters on the usage of the tyre model
[DIMENSION]	tyre dimensions
[OPERATING_CONDITIONS]	operating conditions like inflation pressure
[INERTIA]	tyre and tyre belt mass/inertia properties
[VERTICAL]	vertical stiffness; loaded and effective rolling radius
[STRUCTURAL]	tyre stiffness, damping and eigenfrequencies
[CONTACT_PATCH]	contact length, obstacle enveloping parameters

Input limitations

[INFLATION_PRESSURE_RANGE]	minimum and maximum allowed inflation pressures
[VERTICAL_FORCE_RANGE]	minimum and maximum allowed wheel loads
[LONG_SLIP_RANGE]	minimum and maximum valid longitudinal slips
[SLIP_ANGLE_RANGE]	minimum and maximum valid sideslip angles
[INCLINATION_ANGLE_RANGE]	minimum and maximum valid inclination angles

Magic Formula:

[SCALING_COEFFICIENTS]	Magic Formula scaling factors 55
[LONGITUDINAL_COEFFICIENTS]	coefficients for the longitudinal force Fx
[OVERTURNING_COEFFICIENTS]	coefficients for the overturning moment Mx
[LATERAL_COEFFICIENTS]	coefficients for the lateral force Fy
[ROLLING_COEFFICIENTS]	coefficients for the rolling resistance moment My
[ALIGNING_COEFFICIENTS]	coefficients for the self aligning moment Mz
[TURNSLIP_COEFFICIENTS]	coefficients for turn slip, affects all forces/moments

Though at first sight the number of coefficients may seem extensive, Delft-Tyre has established two methods to significantly facilitate tyre model parameterisation:

1. **MF-Tool**: this is an automated parameter identification tool to determine the tyre model parameters and manipulate the resulting characteristics [8]. Fitting Magic Formula coefficients is a well established process within the vehicle industry.Furthermore, MF-Tool features a generic method for identifying MF-Swift parameters from standardised measurements such as loaded radius, contact length and cleat/drum tests.
2. Reduced Input Data Requirements 54

4.3.2 Reduced Input Data Requirements

If no (or limited) measurement data is available, it is also allowed to omit coefficients in the Tyre Property File. Built-in procedures will be used to provide a reasonable estimate for the missing data and only a small number of coefficients are needed. The next table gives the minimum required coefficients.

When using a reduced parameter file, detailed effects such as combined slip, tyre relaxation effects and enveloping behaviour on short wavelength road obstacles are included, although the related parameters are not explicitly specified (see for the location the Parameters in the Tyre Property File 63).

Coefficient	Description
FITTYP	Magic Formula version number
UNLOADED_RADIUS	Free tyre radius
MASS	Tyre mass
GRAVITY	Gravity acting on belt in Z direction
FNOMIN	Nominal wheel load

VERTICAL_STIFFNESS	Tyre vertical stiffness
VERTICAL_DAMPING	Tyre vertical damping
LONGITUDINAL_STIFFNESS	Tyre overall longitudinal stiffness
LATERAL_STIFFNESS	Tyre overall lateral stiffness
PDX1	Longitudinal friction Mux at Fznom
PKX1	Longitudinal slip stiffness Kfx/Fz at Fznom
PDY1	Lateral friction Muy
PKY1	Maximum value of stiffness Kfy/Fznom
PKY2	Load at which Kfy reaches maximum value

Minimum set of parameters.

Tip: When **extrapolating** to (very) low friction values, the use of "estimated combined slip" possibly improves the performance of the tyre model . "Estimated combined slip" can be turned on by setting the combined slip coefficients in the Tyre Property File to zero or by omitting them.

4.3.3 Scaling factors

Tyre force and moment testing is often done in a laboratory environment (e.g. using a MTS Flat Trac or a drum). The artificial road surface on the tyre test machine may be quite different from a real road surface. Combined with other factors as temperature, humidity, wear, inflation pressure, drum curvature, etc. the tyre behaviour under a vehicle may deviate significantly from the results obtained from a test machine. Differences of up to 20 % in the friction coefficient and cornering stiffness have been reported in literature for a tyre tested on different road surfaces compared to lab measurements.

For this purpose scaling factors are included in the tyre model, which allow the user to manipulate and tune the tyre characteristics, for example to get a better match between full vehicle tests and simulation model. Another application of the scaling factors is that they may be used to eliminate some undesired offsets or shifts in the Magic Formula.

The most important scaling factors are:

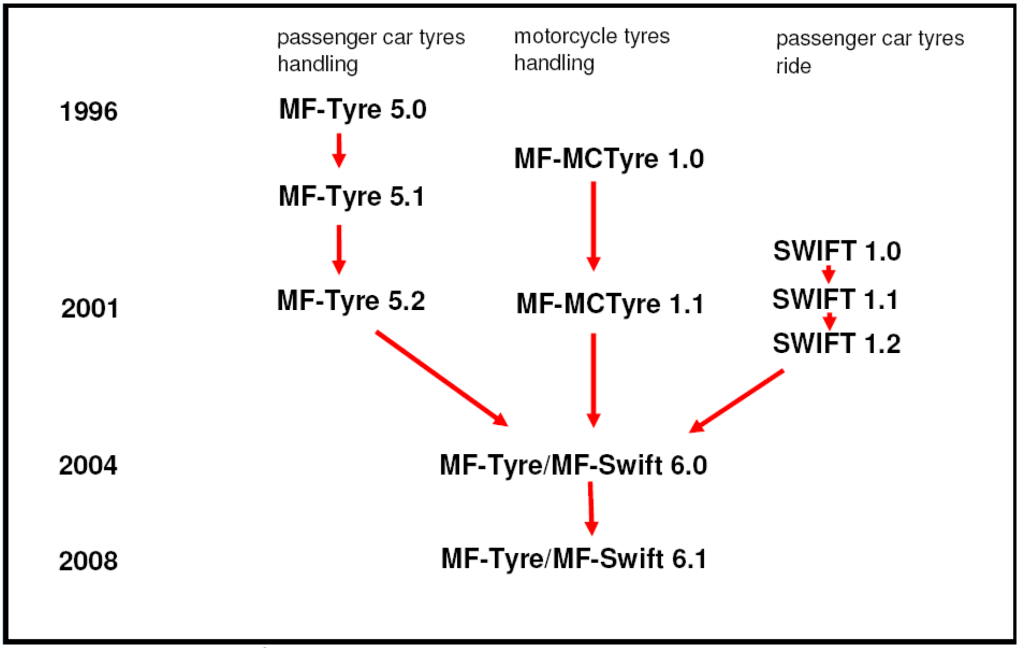
- LMUX longitudinal peak friction coefficient (Fx)
- LKX longitudinal slip stiffness (Fx)
- LMUY lateral peak friction coefficient (Fy)

- LKY cornering stiffness (Fy)
- LKYC camber stiffness (Fy)
- LTR pneumatic trail (Mz)
- LKZC camber moment stiffness (Mz)
- LMP parking moment at standstill (Mz)

Normally when processing the tyre measurements these scaling factors are set to 1, but when doing a validation study on a full vehicle model they can be adjusted to tune the tyre behaviour. The scaling factors are defined in the [SCALING_COEFFICIENTS] section of the Tyre Property File (see section 4.3.1).

4.3.4 Backward compatibility

To be able to use old Tyre Property Files, and test data, MF-Tyre/MF-Swift is backward compatible with older versions (MF-Tyre 5.x, MF-MC-Tyre 1.x , SWIFT 1.x and MF-Tyre/MF-Swift 6.0.x). Tyre Property Files generated for these tyre models will work with MF-Tyre/MF-Swift 6.1 and will give the same simulation results as before.



Backward compatibility of Tyre Property Files.

Due to the built-in estimation procedure 54 it is possible to use for example an existing MF-Tyre 5.2 Tyre Property File and perform simulations including turn slip, rigid ring dynamics and tyre enveloping behaviour, thus already benefiting from the new functionality available in MF-Tyre/MF-Swift 6.1.2.

FITTYP

The selection of the appropriate set of Magic Formula equations is based on the parameter FITTYP in the [MODEL] section of the Tyre Property File. The following conventions apply:

- FITTYP = 5 MF-Tyre 5.0, 5.1 Magic Formula equations
- FITTYP = 6 MF-Tyre 5.2 Magic Formula equations
- FITTYP = 21 MF-Tyre 5.2 Magic Formula equations
- FITTYP = 51 MF-MCTyre 1.0 Magic Formula equations
- FITTYP = 52 MF-MCTyre 1.1 Magic Formula equations
- FITTYP = 60 MF-Tyre 6.0 Magic Formula equations
- FITTYP = 61 MF-Tyre 6.1 Magic Formula equations

MF-Tyre/MF-Swift 6.1.2 accepts all these values for the parameter FITTYP. It is recommended not to change the value of the parameter FITTYP unless you are sure that the model parameters in the Tyre Property File are meant for that specific Magic Formula version!

Differences

However some differences may occur at very low speeds when relaxation behaviour is included combined with a forward velocity below the value specified with the parameter VXLOW in the [MODEL] section. Due to new formulations the tyre behaviour is much more realistic for these operating conditions.

In the case of MF-Swift minor differences may occur between the 1.x, 6.0.x and 6.1 versions due to a different formulation of the contact patch dynamic behaviour. These differences can be observed in the tyre contact forces and slip values, whereas at wheel axle level the differences remain small.

Compatibility

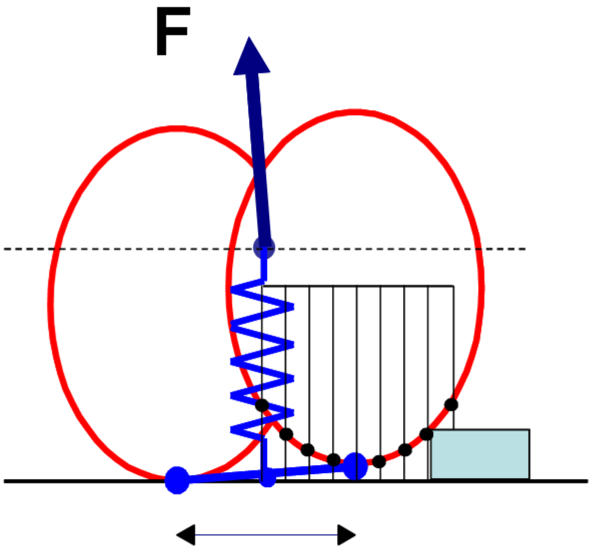
- Former MF-MCTyre users explicitly will have to select "smooth road contact, circular cross section" as Contact Method 45 to get the same results using MF-Tyre 6.1 with their MF-MCTyre datasets.
- Former SWIFT-Tyre 1.x users will have to select "road contact for 2D roads" as Contact Method 45 and "rigid ring dynamics" as Dynamics Mode 47 to get the same results as before.

Discontinued functionality

The camber angle scaling factors LGAX, LGAY and LGAZ are not supported anymore. The camber influence in MF-Tyre/MF-Swift 6.x can now be more conveniently controlled by the new parameters LKYC (Fy) and LKZC (Mz). These parameters allow explicit scaling of the camber stiffness and camber moment stiffness. These new parameters also have to be used in combination with MF-Tyre 5.x and MF-MCTyre 1.x datasets.

4.3.5 Tyre model settings

The MF-Swift model uses elliptical cams to determined the effective road profile, as is shown in the figure below. Each elliptical cam is discretised and each point is evaluated for the road height. Using the effective plane height and angle the direction and magnitude of the forces are determined. This paragraph discusses several setting, which have effect on this effective road profile.



When using road contacts for 2D and 3D roads, the following points can be changed by the user and are explained below:

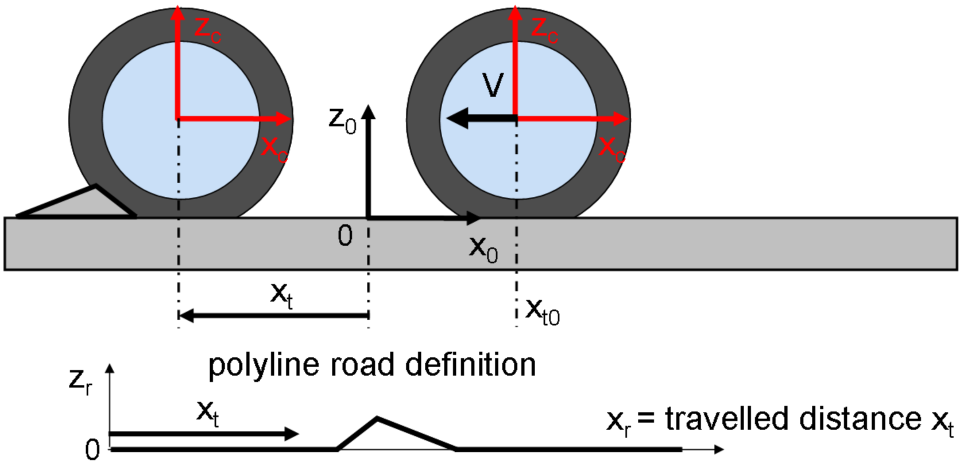
- When using the road contact for **2D roads**:
  - ROAD_DIRECTION 59 (direction of travelled distance, 1 (default) or -1 (reverse)).
  - ROAD_INCREMENT 60 (= increment in road sampling).
- When using the road contact for **3D roads**:
  - ROAD_INCREMENT 60 (= increment in road sampling).
  - ELLIPS_MAX_STEP 60 (= maximum allowed discrete step in road height).

- ELLIPS_NWIDTH 61 (=number of parallel tandem ellipsoids)
- ELLIPS_NLENGTH 61 (=number of successive cams at both sides)

In the following paragraphs more information about above tyre model settings are given:

ROAD_DIRECTION (only: road contact for 2D roads)

The travelled distance x_t is the distance the wheel centre has travelled with respect to the origin of the global coordinate system (x^0, y^0, z^0). By default, the travelled distance is positive in the direction of the x^c-coordinate system. However, the keyword *ROAD_DIRECTION* in the tyre property file can change the sign of the travelled distance for 2D road contact if it is set to -1.



In the figure above an example is given. When using the road contacts for 2D roads the cleat will only be seen if *ROAD_DIRECTION* = -1. Travelled distance x_t then is positive when moving to the left. Note that x_{t0} is negative, since the zero of the travelled distance measure is the origin of the global coordinate system Thus the travelled distance is initially negative and becomes positive when the origin of the global coordinate system is passed.

> **Note**: If the carrier axis system is initially rotated 180° about z-axis (default in ADAMS/Car), the conversion is done automatically, thus then use *ROAD_DIRECTION* = 1.

Note: For Adams users the road reference marker must have to same position and orientation as the global reference marker.

ROAD_INCREMENT

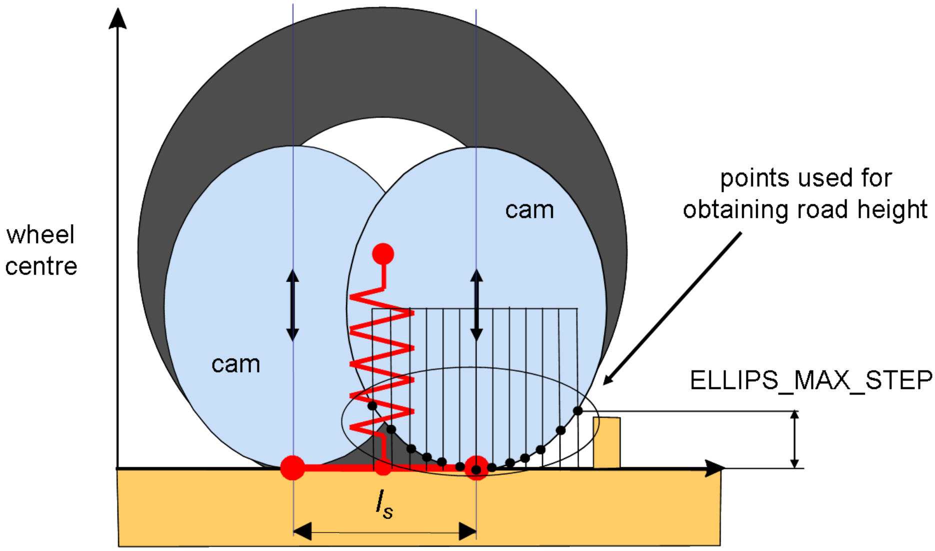
The *ROAD_INCREMENT* parameter is only used in combination with the enveloping model with elliptical cams. It affects the number of points used on the ellipse for calculation of the effective road height & slope. In general a smaller *ROAD_INCREMENT* will give more accurate results, because of more contact points, but more contact evaluations means slower simulation.

Note: Only a limited memory space is reserved for the calculation of the effective road height and slope. If the message “Too many ellipse points, increase parameter ROAD_INCREMENT, or reduce ELLIPS_MAX_STEP” appears when 3D contact method is used, a memory problem has occurred. If the message “Road increment adjusted to minimum possible increment of: (value) m” appears when 2D contact method is used, a memory problem has occurred.

Note: The minimal distance between the contact points on the cams is 1 mm. Thus for highest accuracy, set the *ROAD_INCREMENT* to 1 mm for simulations on sharp obstacles.

ELLIPS_MAX_STEP

For faster simulation the number of points on the elliptical cams should be limited. This can be controlled by the tyre property parameter *ELLIPS_MAX_STEP*. This parameter has to be larger than the obstacle height to prevent extreme slopes and high forces, see below.



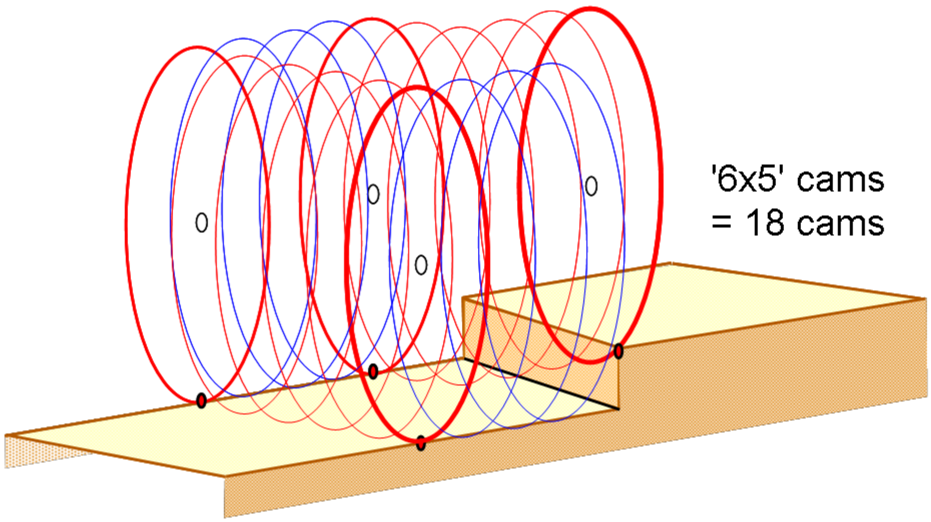
ELLIPS_NWIDTH

For faster simulation the number of parallel tandems (multi-track) should be limited. This can be controlled by the tyre property parameter *ELLIPS_NWIDTH*. For sharp obstacle the default value of 10 parallel ellipsoids generally is sufficient for an accurate simulation. However, with more smooth roads or with cleats oriented perpendicular to the X-axis this value can be limited.

ELLIPS_NLENGTH

For faster simulation the number of successive cams at both sides should be limited. This can be controlled by the tyre property parameter *ELLIPS_NLENGTH*. For sharp obstacle the default value of 10 successive ellipsoids generally is sufficient for an accurate simulation. However, with more smooth roads or with cleats oriented perpendicular to the X-axis this value can be limited.

In the figure below an example of 6 parallel cams in the **front** & **rear** row and 5 successive cams at **both** sides can be seen.



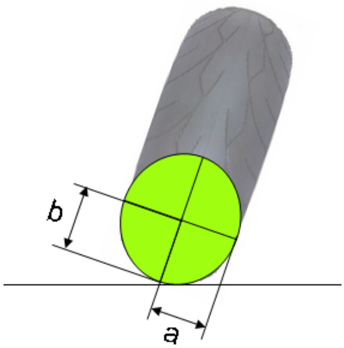
RIGID_RING_DYNAMICS

During dynamic simulation it is advised to select rigid ring dynamics of the model for the most realistic simulations. Then the tyre force element is not massless anymore! The mass and moments of inertia of the tyre belt have to be subtracted from the total wheel + tyre.

4.3.6 Miscellaneous

Motorcycle Contour Ellipse

The motorcycle contour ellipse is the contact ellipse of a motorcycle tyre. It is the ellipse over which the motorcycle tyre rolls, as explained in the figure below.



The dimension of the ellipse can be set using the dimensionless parameters 63 in the Tyre Property File MC_CONTOUR_A and MC_CONTOUR_B,in section [VERTICAL], according to the following formulas:

```
MC_CONTOUR_A = a / tyre width
MC_CONTOUR_B = b / tyre width
```

> **Note**: These parameters are only used by the tyre model in the *smooth road contact, circular cross section (motorcycle tyres)* Contact Method 45, so only for a flat road surface.

4.3.7 Parameters in the Tyre Property File

The following table lists the required and optional parameters for each tyre model version. For convenience, a comparison is made with the previous model versions.

x: required parameter
(x): optional parameter

Tyre Property File		**MF-Tyre 6.1**	**MF-Swift 6.1**	**MF-Tyre 6.0**	**MF-Swift 6.0**	**MF-Tyre 5.2**	**SWIFT 1.2**	**MF-MC Tyre 1.1**
[MODEL]								
FITTYP	Magic Formula version number	61	61	60	60	6	21	52
+	Moment representation 0=ground frame 1=wheel frame	x	x					
TYRESIDE	Position of tyre during measurements	x	x	x	x	x	x	x
LONGVL	Reference speed	x	x	x	x	x	x	x
VXLOW	Lower boundary velocity in slip calculation	x	x	x	x	x	x	x
ROAD_INCREMENT	Increment in road sampling		x		x		x	
ROAD_DIRECTION	Direction of travelled distance		x		x		x	

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
PROPERTY_FILE_FORMAT	Tyre model selection (ADAMS only)	(x)	(x)	(x)	(x)	(x)	(x)	(x)
USE_MODE	Tyre use mode switch (ADAMS only)	(x)	(x)	(x)	(x)	(x)	(x)	(x)
HMAX_LOCAL	Local integration time step (ADAMS only)		(x)		(x)		(x)	
TIME_SWITCH_INTEG	Time when local integrator is activated (ADAMS only)		(x)		(x)		(x)	
[DIMENSION]								
UNLOADED_RADIUS	Free tyre radius	x	x	x	x	x	x	x
WIDTH	Nominal section width of the tyre	x	x	x	x	x	x	x
RIM_RADIUS	Nominal rim radius	x	x	x	x	x	x	x
RIM_WIDTH	Rim width	x	x	x	x	x	x	x
ASPECT_RATIO	Nominal aspect ratio	x	x	x	x	x	x	x
[OPERATING_CONDITIONS]								
INFLPRES	Tyre inflation pressure	x	x					
NOMPRES	Nominal pressure used in (MF) equations	x	x					
[INERTIA]								
MASS	Tyre mass	x	x	x	x		x	
IXX	Tyre diametral moment of inertia	x	x	x	x			
IYY	Tyre polar moment of inertia	x	x	x	x			
BELT_MASS	Belt mass		x		x			
BELT_IXX	Belt diametral moment of inertia		x		x			
BELT_IYY	Belt polar moment of inertia		x		x			
GRAVITY	Gravity acting on belt in Z direction		x		x			
M_B	Portion of tyre mass of tyre belt part						x	
I_BY	Normalized moment of inertia about Y of tyre belt part						x	
I_BXZ	Normalized moment of inertia about XZ of tyre belt part						x	
C_GRV	Gravity constant						x	
[VERTICAL]								
FNOMIN	Nominal wheel load	x	x	x	x	x	x	x
VERTICAL_STIFFNESS	Tyre vertical stiffness	x	x	x	x	x	x	x
VERTICAL_DAMPING	Tyre vertical damping	x	x	x	x	x	x	x
MC_CONTOUR_A	Motorcycle contour ellipse A	x						
MC_CONTOUR_B	Motorcycle contour ellipse B	x						

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
BREFF	Low load stiffness of effective rolling radius	x	x	x	x	x	x	x
DREFF	Peak value of effective rolling radius	x	x	x	x	x	x	x
FREFF	High load stiffness of effective rolling radius	x	x	x	x	x	x	x
Q_RE0	Ratio of free tyre radius with nominal tyre radius	x	x	x	x		x	
Q_V1	Tyre radius increase with speed	x	x	x	x		x	
Q_V2	Vertical stiffness increase with speed	x	x	x	x		x	
Q_FZ2	Quadratic term in load vs. deflection	x	x	x	x		x	
Q_FCX	Longitudinal force influence on vertical stiffness	x	x	x	x		x	
Q_FCY	Lateral force influence on vertical stiffness	x	x	x	x		x	
Q_CAM	Stiffness reduction due to camber	x						
PFZ1	Pressure effect on vertical stiffness	x	x					
BOTTOM_OFFST	Distance to rim when bottoming starts to occur	x	x	x	x		x	
BOTTOM_STIFF	Vertical stiffness of bottomed tyre	x	x	x	x		x	
[STRUCTURAL]								
LONGITUDINAL_STIFFNESS	Tyre overall longitudinal stiffness	x	x	x	x			
LATERAL_STIFFNESS	Tyre overall lateral stiffness	x	x	x	x			
YAW_STIFFNESS	Tyre overall yaw stiffness	x	x	x	x			
FREQ_LONG	Undamped frequency fore/aft and vertical mode		x		x			
FREQ_LAT	Undamped frequency lateral mode		x		x			
FREQ_YAW	Undamped frequency yaw and camber mode		x		x			
FREQ_WINDUP	Undamped frequency wind-up mode		x		x			
DAMP_LONG	Dimensionless damping fore/aft and vertical mode		x		x			
DAMP_LAT	Dimensionless damping lateral mode		x		x			
DAMP_YAW	Dimensionless damping yaw and camber mode		x		x			
DAMP_WINDUP	Dimensionless damping wind-up mode		x		x			
DAMP_RESIDUAL	Residual damping (proportional to stiffness)	x	x	x	x			
DAMP_VLOW	Additional low speed damping (proportional to stiffness)	x	x	x	x			
Q_BVX	Load and speed influence on in-plane translation stiffness		x		x		x	
Q_BVT	Load and speed influence on in-plane rotation stiffness		x		x		x	
PCFX1	Tyre overall longitudinal stiffness vertical deflection dependency linear term	x	x					
PCFX2	Tyre overall longitudinal stiffness vertical deflection dependency quadratic term	x	x					
PCFX3	Tyre overall longitudinal stiffness pressure dependency	x	x					
PCFY1	Tyre overall lateral stiffness vertical deflection dependency linear term	x	x					
PCFY2	Tyre overall lateral stiffness vertical deflection dependency quadratic term	x	x					

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
PCFY3	Tyre overall lateral stiffness pressure dependency	x	x					
PCMZ1	Tyre overall yaw stiffness pressure dependency	x	x					
C_BX0	In-plane belt translation stiffness						x	
C_RX	Longitudinal residual stiffness						x	
C_BT0	In-plane belt rotation stiffness						x	
C_BY	Out-of-plane belt translation stiffness						x	
C_RY	Lateral residual stiffness						x	
C_BGAM	Out-of-plane belt rotation stiffness						x	
C_RP	Yaw residual stiffness						x	
K_BX	In-plane belt translation damping						x	
K_BT	In-plane belt rotation damping						x	
K_BY	Out-of-plane belt translation damping						x	
K_BGAM	Out-of-plane belt rotation damping						x	
[CONTACT_PATCH]								
Q_RA1	Square root term in contact length equation		x					
Q_RA2	Linear term in contact length equation		x					
Q_RB1	Root term in contact width equation		x					
Q_RB2	Linear term in contact width equation		x					
ELLIPS_SHIFT	Scaling of distance between front and rear ellipsoid		x		x		x	
ELLIPS_LENGTH	Semimajor axis of ellipsoid		x		x		x	
ELLIPS_HEIGHT	Semiminor axis of ellipsoid		x		x		x	
ELLIPS_ORDER	Order of ellipsoid		x		x		x	
ELLIPS_MAX_STEP	Maximum height of road step		x		x		x	
ELLIPS_NWIDTH	Number of parallel ellipsoids		x		x		x	
ELLIPS_NLENGTH	Number of ellipsoids at sides of contact patch		x		x		x	
Q_A2	Linear load term in contact length				x		x	
Q_A1	Square root load term in contact length				x		x	
ELLIPS_INC	Discretisation increment of ellipsoid contour				x		x	
Q_LBF	Length of basic function				x		x	
Q_LOS1	Basic function offset threshold				x		x	
Q_LOS2	Basic function offset scaling factor with basic function length				x		x	
Q_LIMP1	Linear contact length term in basic function shift				x		x	
Q_LIMP3	Scaling factor for quasi-static longitudinal enveloping force				x			
Q_LIMP4	Scaling factor for dynamic longitudinal enveloping force				x			
Q_LIMP2	Quadratic contact length term in basic function shift						x	

Delft-Tyre

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
[INFLATION_PRESSURE_RANGE]								
PRESMIN	Minimum allowed inflation pressure	x	x					
PRESMAX	Maximum allowed inflation pressure	x	x					
[VERTICAL_FORCE_RANGE]								
FZMIN	Minimum allowed wheel load	x	x	x	x	x	x	x
FZMAX	Maximum allowed wheel load	x	x	x	x	x	x	x
[LONG_SLIP_RANGE]								
KPUMIN	Minimum valid wheel slip	x	x	x	x	x	x	x
KPUMAX	Maximum valid wheel slip	x	x	x	x	x	x	x
[SLIP_ANGLE_RANGE]								
ALPMIN	Minimum valid slip angle	x	x	x	x	x	x	x
ALPMAX	Maximum valid slip angle	x	x	x	x	x	x	x
[INCLINATION_ANGLE_RANGE]								
CAMMIN	Minimum valid camber angle	x	x	x	x	x	x	x
CAMMAX	Maximum valid camber angle	x	x	x	x	x	x	x
[SCALING_COEFFICIENTS]								
LFZO	Scale factor of nominal (rated) load	x	x	x	x	x	x	x
LCX	Scale factor of Fx shape factor	x	x	x	x	x	x	x
LMUX	Scale factor of Fx peak friction coefficient	x	x	x	x	x	x	x
LEX	Scale factor of Fx curvature factor	x	x	x	x	x	x	x
LKX	Scale factor of slip stiffness	x	x	x	x	x	x	x
LHX	Scale factor of Fx horizontal shift	x	x	x	x	x	x	
LVX	Scale factor of Fx vertical shift	x	x	x	x	x	x	x
LCY	Scale factor of Fy shape factor	x	x	x	x	x	x	x
LMUY	Scale factor of Fy peak friction coefficient	x	x	x	x	x	x	x
LEY	Scale factor of Fy curvature factor	x	x	x	x	x	x	x
LKY	Scale factor of cornering stiffness	x	x	x	x	x	x	x
LKYC	Scale factor of camber stiffness	x	x	x	x			
LKZC	Scale factor of camber moment stiffness	x	x	x	x			

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
LHY	Scale factor of Fy horizontal shift	x	x	x	x	x	x	x
LVY	Scale factor of Fy vertical shift	x	x	x	x	x	x	
LTR	Scale factor of Peak of pneumatic trail	x	x	x	x	x	x	x
LRES	Scale factor for offset of residual torque	x	x	x	x	x	x	x
LXAL	Scale factor of alpha influence on Fx	x	x	x	x	x	x	x
LYKA	Scale factor of alpha influence on Fx	x	x	x	x	x	x	x
LVYKA	Scale factor of kappa induced Fy	x	x	x	x	x	x	x
LS	Scale factor of Moment arm of Fx	x	x	x	x	x	x	x
LMX	Scale factor of overturning moment	x	x	x	x	x	x	x
LVMX	Scale factor of Mx vertical shift	x	x	x	x	x	x	x
LMY	Scale factor of rolling resistance torque	x	x	x	x	x	x	x
LMP	Scale factor of parking moment	x	x	x	x			
LKC	Scale factor of camber stiffness							x
LCC	Scale factor of camber shape factor							x
LEC	Scale factor of camber curvature factor							x
LSGKP	Scale factor of Relaxation length of Fx					x	x	x
LSGAL	Scale factor of Relaxation length of Fy					x	x	x
LGYR	Scale factor gyroscopic moment					x	x	x
[LONGITUDINAL_COEFFICIENTS]								
PCX1	Shape factor Cfx for longitudinal force	x	x	x	x	x	x	x
PDX1	Longitudinal friction Mux at Fznom	x	x	x	x	x	x	x
PDX2	Variation of friction Mux with load	x	x	x	x	x	x	x
PDX3	Variation of friction Mux with camber	x	x	x	x	x	x	x
PEX1	Longitudinal curvature Efx at Fznom	x	x	x	x	x	x	x
PEX2	Variation of curvature Efx with load	x	x	x	x	x	x	x
PEX3	Variation of curvature Efx with load squared	x	x	x	x	x	x	x
PEX4	Factor in curvature Efx while driving	x	x	x	x	x	x	x
PKX1	Longitudinal slip stiffness Kfx/Fz at Fznom	x	x	x	x	x	x	x
PKX2	Variation of slip stiffness Kfx/Fz with load	x	x	x	x	x	x	x
PKX3	Exponent in slip stiffness Kfx/Fz with load	x	x	x	x	x	x	x
PHX1	Horizontal shift Shx at Fznom	x	x	x	x	x	x	
PHX2	Variation of shift Shx with load	x	x	x	x	x	x	
PVX1	Vertical shift Svx/Fz at Fznom	x	x	x	x	x	x	x
PVX2	Variation of shift Svx/Fz with load	x	x	x	x	x	x	x
RBX1	Slope factor for combined slip Fx reduction	x	x	x	x	x	x	x

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
RBX2	Variation of slope Fx reduction with kappa	x	x	x	x	x	x	x
RBX3	Influence of camber on stiffness for Fx combined	x	x	x	x			x
RCX1	Shape factor for combined slip Fx reduction	x	x	x	x	x	x	x
REX1	Curvature factor of combined Fx	x	x	x	x	x	x	x
REX2	Curvature factor of combined Fx with load	x	x	x	x	x	x	x
RHX1	Shift factor for combined slip Fx reduction	x	x	x	x	x	x	x
PPX1	Linear pressure effect on slip stiffness	x	x					
PPX2	Quadratic pressure effect on slip stiffness	x	x					
PPX3	Linear pressure effect on longitudinal friction	x	x					
PPX4	Quadratic pressure effect on longitudinal friction	x	x					
PTX1	Relaxation length SigKap0/Fz at Fznom					x	x	x
PTX2	Variation of SigKap0/Fz with load					x	x	x
PTX3	Variation of SigKap0/Fz with exponent of load					x	x	x
[OVERTURNING_COEFFICIENTS]								
QSX1	Overturning moment offset	x	x	x	x	x	x	x
QSX2	Camber induced overturning couple	x	x	x	x	x	x	x
QSX3	Fy induced overturning couple	x	x	x	x	x	x	x
QSX4	Mixed load, lateral force and camber on Mx	x	x	x	x			
QSX5	Load effect on Mx with lateral force and camber	x	x	x	x			
QSX6	B-factor of load with Mx	x	x	x	x			
QSX7	Camber with load on Mx	x	x	x	x			
QSX8	Lateral force with load on Mx	x	x	x	x			
QSX9	B-factor of lateral force with load on Mx	x	x	x	x			
QSX10	Vertical force with camber on Mx	x	x	x	x			
QSX11	B-factor of vertical force with camber on Mx	x	x	x	x			
QSX12	Camber squared induced overturning moment	x	x					
QSX13	Lateral force induced overturning moment	x	x					
QSX14	Lateral force induced overturning moment with camber	x	x					
PPMX1	Influence of inflation pressure on overturning moment	x	x					
[LATERAL_COEFFICIENTS]								
PCY1	Shape factor Cfy for lateral forces	x	x	x	x	x	x	x
PDY1	Lateral friction Muy	x	x	x	x	x	x	x
PDY2	Variation of friction Muy with load	x	x	x	x	x	x	x

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
PDY3	Variation of friction Muy with squared camber	x	x	x	x	x	x	x
PEY1	Lateral curvature Efy at Fznom	x	x	x	x	x	x	x
PEY2	Variation of curvature Efy with load	x	x	x	x	x	x	x
PEY3	Zero order camber dependency of curvature Efy	x	x	x	x	x	x	x
PEY4	Variation of curvature Efy with camber	x	x	x	x	x	x	x
PEY5	Camber curvature Efc	x	x	x	x			x
PKY1	Maximum value of stiffness Kfy/Fznom	x	x	x	x	x	x	x
PKY2	Load at which Kfy reaches maximum value	x	x	x	x	x	x	x
PKY3	Variation of Kfy/Fznom with camber	x	x	x	x	x	x	x
PKY4	Curvature of stiffness Kfy	x	x	x	x			x
PKY5	Peak stiffness variation with camber squared	x	x	x	x			x
PKY6	Camber stiffness factor	x	x	x	x			x
PKY7	Load dependency of camber stiffness factor	x	x	x	x			x
PHY1	Horizontal shift Shy at Fznom	x	x	x	x	x	x	x
PHY2	Variation of shift Shy with load	x	x	x	x	x	x	
PVY1	Vertical shift in Svy/Fz at Fznom	x	x	x	x	x	x	
PVY2	Variation of shift Svy/Fz with load	x	x	x	x	x	x	
PVY3	Variation of shift Svy/Fz with camber	x	x	x	x	x	x	
PVY4	Variation of shift Svy/Fz with camber and load	x	x	x	x	x	x	
RBY1	Slope factor for combined Fy reduction	x	x	x	x	x	x	x
RBY2	Variation of slope Fy reduction with alpha	x	x	x	x	x	x	x
RBY3	Shift term for alpha in slope Fy reduction	x	x	x	x	x	x	x
RBY4	Influence of camber on stiffness of Fy combined	x	x	x	x			x
RCY1	Shape factor for combined Fy reduction	x	x	x	x	x	x	x
REY1	Curvature factor of combined Fy	x	x	x	x	x	x	x
REY2	Curvature factor of combined Fy with load	x	x	x	x	x	x	x
RHY1	Shift factor for combined Fy reduction	x	x	x	x	x	x	x
RHY2	Shift factor for combined Fy reduction with load	x	x	x	x	x	x	x
RVY1	Kappa induced side force Svyk/Muy*Fz at Fznom	x	x	x	x	x	x	x
RVY2	Variation of Svyk/Muy*Fz with load	x	x	x	x	x	x	x
RVY3	Variation of Svyk/Muy*Fz with camber	x	x	x	x	x	x	x
RVY4	Variation of Svyk/Muy*Fz with alpha	x	x	x	x	x	x	x
RVY5	Variation of Svyk/Muy*Fz with kappa	x	x	x	x	x	x	x
RVY6	Variation of Svyk/Muy*Fz with atan(kappa)	x	x	x	x	x	x	x
PPY1	Pressure effect on cornering stiffness magnitude	x	x					
PPY2	Pressure effect on location of cornering stiffness peak	x	x					

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MCTyre 1.1
PPY3	Linear pressure effect on lateral friction	x	x					
PPY4	Quadratic pressure effect on lateral friction	x	x					
PPY5	Influence of inflation pressure on camber stiffness	x	x					
PCY2	Shape factor Cfc for camber forces							x
PHY3	Variation of shift Shy with camber					x	x	
PTY1	Peak value of relaxation length SigAlp0/R0					x	x	x
PTY2	Value of Fz/Fznom where SigAlp0 is extreme					x	x	x
PTY3	Value of Fz/Fznom where Sig_alpha is maximum							x
[ROLLING_COEFFICIENTS]								
QSY1	Rolling resistance torque coefficient	x	x	x	x	x	x	x
QSY2	Rolling resistance torque depending on Fx	x	x	x	x	x	x	x
QSY3	Rolling resistance torque depending on speed	x	x	x	x	x	x	x
QSY4	Rolling resistance torque depending on speed ^4	x	x	x	x	x	x	x
QSY5	Rolling resistance torque depending on camber squared	x	x					
QSY6	Rolling resistance torque depending on load and camber squared	x	x					
QSY7	Rolling resistance torque coefficient load dependency	x	x					
QSY8	Rolling resistance torque coefficient pressure dependency	x	x					
[ALIGNING_COEFFICIENTS]								
QBZ1	Trail slope factor for trail Bpt at Fznom	x	x	x	x	x	x	x
QBZ2	Variation of slope Bpt with load	x	x	x	x	x	x	x
QBZ3	Variation of slope Bpt with load squared	x	x	x	x	x	x	x
QBZ4	Variation of slope Bpt with camber	x	x	x	x	x	x	x
QBZ5	Variation of slope Bpt with absolute camber	x	x	x	x	x	x	x
QBZ9	Slope factor Br of residual torque Mzr	x	x	x	x	x	x	x
QBZ10	Slope factor Br of residual torque Mzr	x	x	x	x	x	x	x
QCZ1	Shape factor Cpt for pneumatic trail	x	x	x	x	x	x	x
QDZ1	Peak trail Dpt" = Dpt*(Fz/Fznom*R0)	x	x	x	x	x	x	x
QDZ2	Variation of peak Dpt with load	x	x	x	x	x	x	x
QDZ3	Variation of peak Dpt with camber	x	x	x	x	x	x	x
QDZ4	Variation of peak Dpt with camber squared	x	x	x	x	x	x	x
QDZ6	Peak residual torque Dmr = Dmr/(Fz*R0)	x	x	x	x	x	x	x
QDZ7	Variation of peak factor Dmr with load	x	x	x	x	x	x	x
QDZ8	Variation of peak factor Dmr with camber	x	x	x	x	x	x	x

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
QDZ9	Variation of peak factor Dmr with camber and load	x	x	x	x	x	x	x
QDZ10	Variation of peak factor Dmr with camber squared	x	x	x	x			x
QDZ11	Variation of Dmr with camber squared and load	x	x	x	x			x
QEZ1	Trail curvature Ept at Fznom	x	x	x	x	x	x	x
QEZ2	Variation of curvature Ept with load	x	x	x	x	x	x	x
QEZ3	Variation of curvature Ept with load squared	x	x	x	x	x	x	x
QEZ4	Variation of curvature Ept with sign of Alpha-t	x	x	x	x	x	x	x
QEZ5	Variation of Ept with camber and sign Alpha-t	x	x	x	x	x	x	x
QHZ1	Trail horizontal shift Sht at Fznom	x	x	x	x	x	x	x
QHZ2	Variation of shift Sht with load	x	x	x	x	x	x	x
QHZ3	Variation of shift Sht with camber	x	x	x	x	x	x	x
QHZ4	Variation of shift Sht with camber and load	x	x	x	x	x	x	x
SSZ1	Nominal value of s/R0: effect of Fx on Mz	x	x	x	x	x	x	x
SSZ2	Variation of distance s/R0 with Fy/Fznom	x	x	x	x	x	x	x
SSZ3	Variation of distance s/R0 with camber	x	x	x	x	x	x	x
SSZ4	Variation of distance s/R0 with load and camber	x	x	x	x	x	x	x
PPZ1	Linear pressure effect on pneumatic trail	x	x					
PPZ2	Influence of inflation pressure on residual aligning torque	x	x					
QTZ1	Gyroscopic torque constant					x	x	x
MBELT	Belt mass of the wheel					x	x	x
[TURNSLIP_COEFFICIENTS]								
PDXP1	Peak Fx reduction due to spin parameter	x	x	x	x			
PDXP2	Peak Fx reduction due to spin with varying load parameter	x	x	x	x			
PDXP3	Peak Fx reduction due to spin with kappa parameter	x	x	x	x			
PKYP1	Cornering stiffness reduction due to spin	x	x	x	x			
PDYP1	Peak Fy reduction due to spin parameter	x	x	x	x			
PDYP2	Peak Fy reduction due to spin with varying load parameter	x	x	x	x			
PDYP3	Peak Fy reduction due to spin with alpha parameter	x	x	x	x			
PDYP4	Peak Fy reduction due to square root of spin parameter	x	x	x	x			
PHYP1	Fy-alpha curve lateral shift limitation	x	x	x	x			
PHYP2	Fy-alpha curve maximum lateral shift parameter	x	x	x	x			
PHYP3	Fy-alpha curve maximum lateral shift varying with load parameter	x	x	x	x			
PHYP4	Fy-alpha curve maximum lateral shift parameter	x	x	x	x			
PECP1	Camber w.r.t. spin reduction factor parameter in camber stiffness	x	x	x	x			

Tyre Property File		MF-Tyre 6.1	MF-Swift 6.1	MF-Tyre 6.0	MF-Swift 6.0	MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
PECP2	Camber w.r.t. spin reduction factor varying with load parameter in camber stiffness	x	x	x	x			
QDTP1	Pneumatic trail reduction factor due to turn slip parameter	x	x	x	x			
QCRP1	Turning moment at constant turning and zero forward speed parameter	x	x	x	x			
QCRP2	Turn slip moment (at alpha=90deg) parameter for increase with spin	x	x	x	x			
QBRP1	Residual (spin) torque reduction factor parameter due to side slip	x	x	x	x			
QDRP1	Turn slip moment peak magnitude parameter	x	x	x	x			

Obsolete parameters which may be in a Tyre Property File, but are ignored by MF-Tyre/MF-Swift 6.x

	description		MF-Tyre 5.2	SWIFT 1.2	MF-MC Tyre 1.1
[MODEL]					
TYPE		*1*	x	x	x
MFSAFE1		*1*	x	x	x
MFSAFE2		*1*	x	x	x
MFSAFE3		*1*	x	x	x
[SHAPE]	The complete shape section is obsolete	*2*	x		x
[INERTIA]					
M_A	Portion of tyre mass of tyre part fixed to rim	*3*		x	
I_AY	Normalized moment of inertia about Y of tyre part fixed to rim	*3*		x	
I_AXZ	Normalized moment of inertia about XZ of tyre part fixed to rim	*3*		x	
M_R	Normalized residual mass	*4*		x	
I_R	Normalized moment of inertia about Z of residual mass	*4*		x	
[STRUCTURAL]					
K_RX	Longitudinal residual damping	*5*		x	
K_RY	Lateral residual damping	*5*		x	
K_RP	Yaw residual damping	*5*		x	
[VERTICAL]					

BOTTOM_TRNSF	Transition range of bottoming	*6*		x	
[CONTACT_PATCH]					
FLT_A	Filter constant contact length	*7*		x	
Q_KC1	Low speed tread element damping coefficient	*8*		x	
Q_KC2	Low speed tread element damping coefficient	*8*		x	
[SCALING_COEFFICIENTS]					
LGAX	Scale factor of camber for Fx	*9*	x	x	x
LGAY	Scale factor of camber for Fy	*10*	x	x	x
LGAZ	Scale factor of camber for Mz	*11*	x	x	x

1	*parameter was not used*
2	*used in combination with ADAMS durability contact;*
	replaced by motorcycle contact and basic functions/ellipsoid contact
3	*replaced by new mass/inertia defintions*
4	*in MF-Swift 6.0 and 6.1 a new formulation is used without residual mass*
5	*replaced by parameter DAMP_RESIDUAL*
6	*parameter deleted*
7	*parameter set internally in the software*
8	*replaced by parameter DAMP_VLOW*
9	*parameter deleted, adjust PDX3 directly*
10	*camber force stiffness is controlled by parameter LKYC*
11	*camber moment stiffness is controlled by parameter LKZC*

4.4 Road Data File

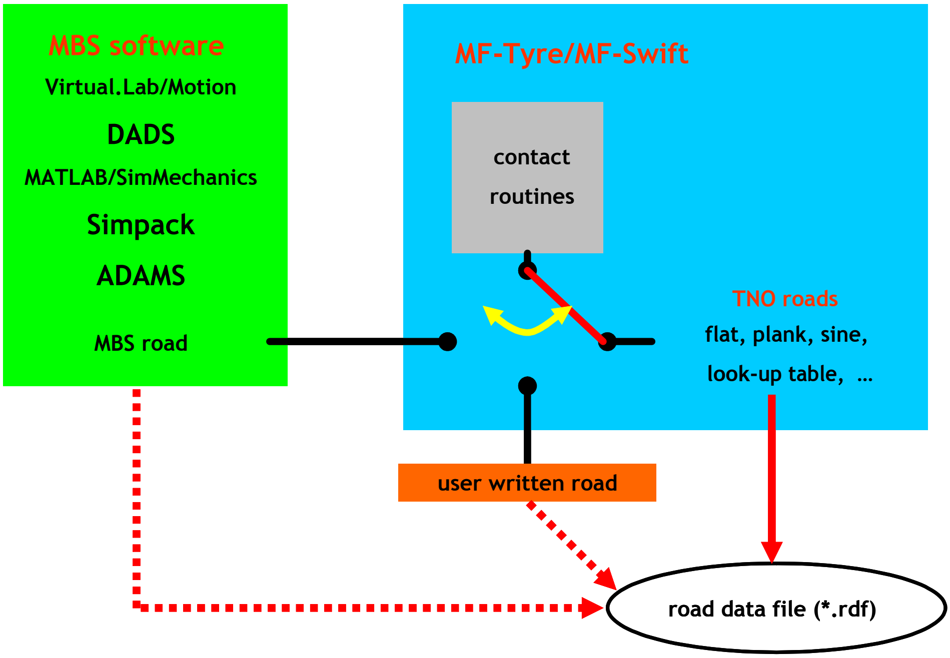
In general three ways exist to define a **road surface**:

- **Multi-body road** (default)
- TNO road 76 (including TNO OpenCRG Road 82)
- User-written subroutine

The default is using the road surfaces from the multi-body package MF-Tyre/MF-Swift is used in. Besides, TNO offers several relatively simple road surface types that can be used with the tyre model, e.g. with MATLAB/SimMechanics.

Finally, for experienced MF-Tyre/MF-Swift users that have Fortran programming experience an option exists to write their own road routines. Please contact 34 TNO for more information about the the user-written subroutine.

The various options are illustrated in the figure below:



Switching between these options is done based on the contents of the road data file. When the user specifies a road data file, MF-Tyre/MF-Swift analyses the file to see if the format is a TNO Road Type 76. If this is the case, the TNO road subroutine is used. If not the multi-body road will be called (default).

Overruling the switching mechanism

This switching mechanism can be overruled by the keyword `ROAD_SOURCE` in the `[MODEL]` section of the tyre property file. Only use this possibility if you are an experienced user. Three options exist:

- `ROAD_SOURCE = 'TNO'` use MF-Tyre/MF-Swift internal road definition
- `ROAD_SOURCE = 'MBS'` use road definition of the MBS package
- `ROAD_SOURCE = 'USER'` use the user written road

4.4.1 TNO Road Types

TNO offers the following road types:

- Flat Road 77
- Plank Road 77
- Polyline Road 78
- Sine Road 79
- Drum_Road 80
- OpenCRG Road 82

Note: The TNO Road Types are not available in all multi-body simulation packages.

TNO Road Definition

These road surfaces are defined in road data files (*.rdf). Like the Tyre Property File, the road data file consists of various sections indicated with square brackets:

```
! Comments section
$-----------------------------------------------------------UNITS
[UNITS]
LENGTH          = 'meter'
FORCE           = 'newton'
ANGLE           = 'degree'
MASS            = 'kg'
TIME            = 'sec'
$-----------------------------------------------------------MODEL
[MODEL]
ROAD_TYPE       = '...'
$------------------------------------------------------PARAMETERS
[PARAMETERS]
...
```

In the [UNITS] section, the units that are used in the road data file are set. The [MODEL] and [PARAMETERS] section are described below.

[Model] section

The [MODEL] section is used to specify the road type:

Road Type	Coding
Flat Road 77	ROAD_TYPE = 'flat'
Plank Road 77	ROAD_TYPE = 'plank'
Polyline Road 78	ROAD_TYPE = 'poly_line'
Sine Road 79	ROAD_TYPE = 'sine'

Drum Road [80]	`ROAD_TYPE = 'drum'`
OpenCRG Road [82]	`ROAD_TYPE = 'crg'`

[Parameters] section

The [PARAMETERS] section contains general and type specific parameters for the road surface.

General

The general parameters are valid for all Road types (except OpenCRG Road [82] for which only MU is valid) and are listed below:

MU	Road friction correction factor (not the friction value itself), to be multiplied with the LMU scaling factors of the tyre model. Default setting: MU = 1.0.
OFFSET	Vertical offset of the ground with respect to inertial frame.
ROTATION_ANGLE_XY_PLANE	Rotation angle of the XY-plane about the road Z-axis, i.e. definition of the positive X-axis of the road with respect to the inertial frame.
DRUM_RADIUS	Radius of the drum.

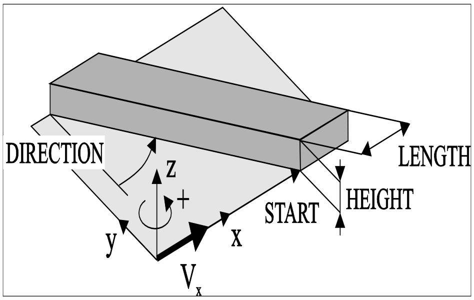
The road surface type specific parameters are explained in the next sections:

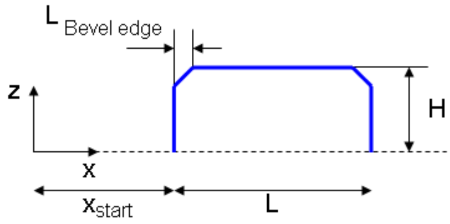
Flat Road

As the name already indicates this is a flat road surface. It has no parameters, except the General [77] parameters.

Plank Road

This is a single cleat or plank that is oriented perpendicular, or in oblique direction relative to the X-axis with or without bevel edges.





HEIGHT	Height of the cleat.
START	Distance along the X-axis of the road from the origin to the start of the cleat (i.e. travelling from the origin the tyre will hit the cleat at START).
LENGTH	Length of the cleat (including bevel) along X-axis of the road.
BEVEL_EDGE_LENGTH	Length of the 45 deg. bevel edge of the cleat.
DIRECTION	Rotation of the cleat about the Z-axis with respect to the Y-axis of the road. If the cleat is placed crosswise, DIRECTION = 0. If the cleat is along the X-axis, DIRECTION = 90.

Polyline

Road height as a function of travelled distance.

The [PARAMETERS] block must have a (XZ_DATA) sub-block. The sub-block consists of three columns of numerical data:

- Column one is a set of X-values in ascending order;
- Columns two and three are sets of respective Z-values for left and right track.

Example:

```
[PARAMETERS]
MU                              =  1.0 $ peak friction scaling
                                        $ coefficient
OFFSET                          =    0 $ vertical offset of the
                                        $ ground wrt inertial frame
ROTATION_ANGLE_XY_PLANE         =    0 $ definition of the positive
```

```
                                                  $ X-axis of the road wrt
                                                   $ inertial frame
$
$    X_road      Z_left    Z_right
(XZ_DATA)
-1.0e04           0         0
0                 0         0
0.0500            0         0
...               ...       ...
```

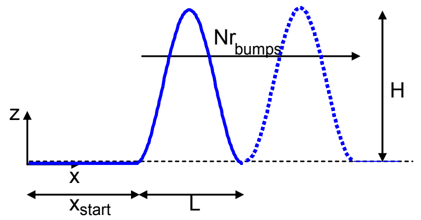
Sine Road

Road surface consisting of one or more sine waves with constant wavelength.

The TNO sine road is implemented as:

$$z(x) = \frac{H}{2}\left(1 - \cos\left(\frac{2\pi \cdot (x - x_{start})}{L}\right)\right)$$

With z: z-coordinate road; H; Height; x: current position; x_{start}: start of sine wave; L: Length



HEIGHT	Height of the sine wave.
START	Distance along the X-axis of the road to the start of the sine wave.
LENGTH	Wavelength of the sine wave along X-axis of the road.
DIRECTION	Rotation of the bump about the Z-axis with respect to the X-axis of the road. If the bump is placed crosswise, DIRECTION = 0. If the bump is along the X-axis, DIRECTION = 90.
N_BUMPS	Number of consecutive sine bumps.

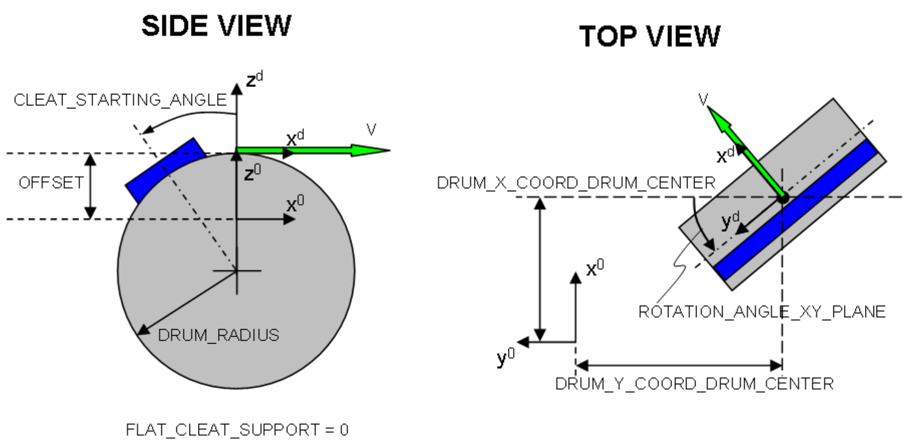
Note, the sine road in ADAMS is defined as follows:

$$z(x) = \frac{H}{2}\sin\left(\frac{2\pi \cdot (x - x_{start})}{L}\right)$$

Drum Road

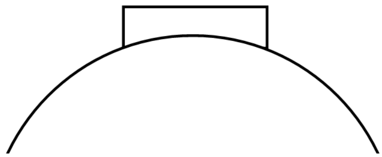
Note: For fixed spindle height cleat simulations on curved surfaces it is recommended to use the default road surfaces like e.g. Polyline Road or Plank Road and specify the DRUM_RADIUS as parameter. This is computational more efficient. For non-fixed spindle simulations, for example suspension or full vehicle on a drum, it is advised to use the Drum Road.

As the name already indicates this is a drum road surface. A single cleat that is oriented perpendicular to the local drum X,Z-plane can be mounted, which will be passed every revolution of the drum.

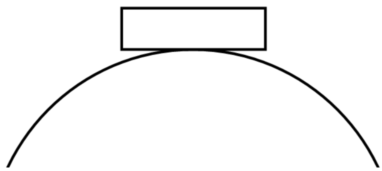


V	Tangential velocity of the drum in the local right handed drum coordinate system
CLEAT_STARTING_ANGLE	Drum angle coordinate of the centre of the first cleat (positive is a clockwise rotation). When undefined starting angle is 0.
DRUM_X_COORD_DRUM_CENTER	X-coordinate of drum centre in global coordinates, used for all wheels. When undefined x-coordinate drum centre is 0.

DRUM_Y_COORD_DRUM_CENTER	Y-coordinate of drum centre in global coordinates, used for all wheels. When undefined y-coordinate drum centre is 0.
DRUM_FLAT_CLEAT_SUPPORT	0: cleat is mounted on the drum such that its support on the drum surface is curved according to overall drum curvature.



1: cleat is mounted on the drum such that its support on the drum surface is flat.



When undefined cleat support is 0.

Note that the [PARAMETERS] block must have a (XZ_DATA) sub-block when an obstacle is mounted. The sub-block consists of two columns of numerical data:

- Column one is a set of X-values in ascending order;
- Column two is a set of Z-values.

These XZ_DATA are solely used to define the dimensions and shape of the obstacle.

Example:

```
! Comments section
$------------------------------------------------------------UNITS
[UNITS]
LENGTH          = 'mm'
FORCE          = 'newton'
ANGLE          = 'degree'
MASS           = 'kg'
TIME           = 'sec'
$------------------------------------------------------------MODEL
[MODEL]
ROAD_TYPE      = 'drum'
$-------------------------------------------------------PARAMETERS
```

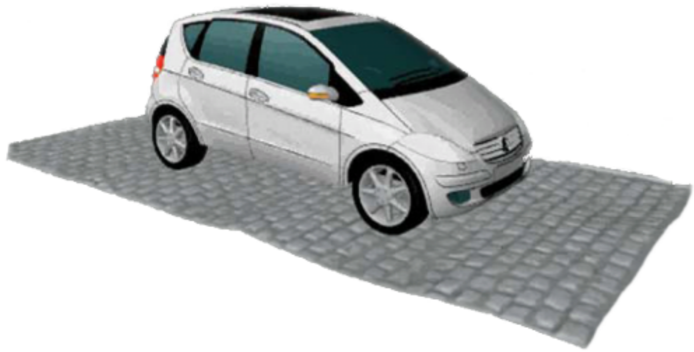
```
[PARAMETERS]
MU                          =  1.0 $ peak friction scaling
                                   $ coefficient
OFFSET                      =    0 $ vertical offset of the
                                   $ ground wrt inertial frame
ROTATION_ANGLE_XY_PLANE      =    0 $ definition of the positive
                                   $ X-axis of the road wrt
                                   $ inertial frame
$
DRUM_RADIUS                  =   2500 $ radius of the drum
V                            = -10000 $ velocity of the drum
CLEAT_SARTING_ANGLE          =      0 $ start angle of the centre
                                      $ of the cleat at the drum
DRUM_X_COORD_DRUM_CENTER     =      0 $ global displacement of
                                      $ the drum in x-direction
DRUM_Y_COORD_DRUM_CENTER     =      0 $ global displacement of
                                      $ the drum in y-direction
DRUM_FLAT_CLEAT_SUPPORT      =      0 $ the support of the cleat
$
(XZ_DATA)
-25       0
-15      10
 15      10
 25       0
```

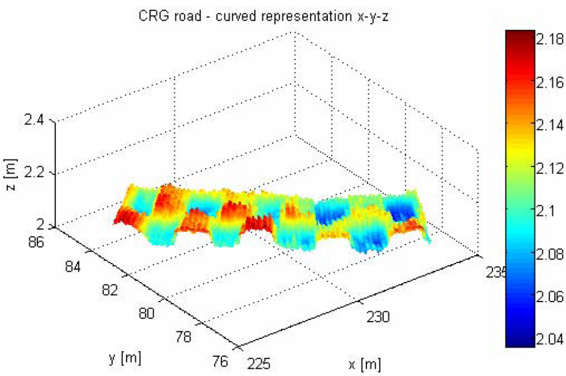
4.4.2 TNO OpenCRG Road

The TNO **OpenCRG** Road is the implementation of the interface between MF-Tyre/ MF-Swift and OpenCRG, maintained by VIRES Simulationstechnologie GmbH, Germany.

OpenCRG

OpenCRG is an initiative to provide a unified approach to represent 3D road data in vehicle simulations. The motivation is that simulation applications of vehicle handling, ride comfort, and durability load profiles ask for a reliable and efficient road representations. OpenCRG is based on CRG 84, Curved Regular Grid, developed by Daimler, which is made available to everybody.





The provided free material includes an efficient C-API implementation to evaluate the recorded 3D surface information and some Matlab® functions to handle the CRG road data files.

Documentation

The material for OpenCRG, including **documentation**, **source code** and **tools**, can be found on the OpenCRG website www.opencrg.org, in the section Download, using the links:

- User Manual
- OpenCRG tools (C-API and MATLAB)

License

OpenCRG is **licensed** under the Apache License, version 2.0. The License Conditions for OpenCRG are listed in the License Manual 11.

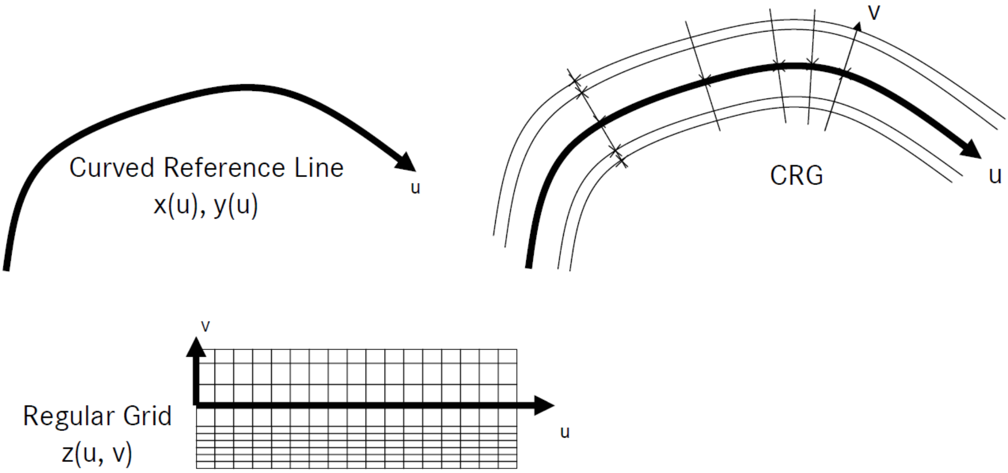
OpenCRG is a registered trademark of Daimler AG.

Invitation

The founders invite the community to share experiences and would be pleased to have further contributions to complement and extend their initial work.

CRG

CRG, or Curved Regular Grid, represents road elevation data close to an arbitrary road center line. The road is represented as a (curved) reference line, and a regular elevation grid, see figure below.



This approach results in improved storage efficiency (smaller road data files), and faster elevation evaluation, with respect to other methods.

> **Note**: The start of the CRG track is, by default, translated to the origin. This can be overruled by including an (empty) "$ROAD_CRG_MODS" block.

Curved Reference Line

The Curved Reference Line is defined in the base plane (usually x,y) by setting the direction (=heading / yaw angle). Optionally, a pitch and bank angle can be defined to represent the hilliness and cross slope.

Regular Elevation Grid

The Regular Elevation Grid, which is locally orthogonal, is a special form of Regular Grid, or Curvilinear or Structured Grid. It defines the elevation in the proximity of the reference line. The columns are longitudinal cuts that are parallel to the reference line. The rows are lateral cuts orthogonal to the reference line.

Friction

Friction, or variation of friction over the road surface, is a major issue in handling simulations. In the TNO CRG Road implementation the friction can be modelled in the same way as the road elevation. The data in the CRG file now does not represent elevation anymore, but friction value, or better described the Road Friction Correction Factor, which is multiplied with the other friction (scaling) factors defined elsewhere. This way e.g. mu-split situations can be modelled.

As friction and elevation data are stored in separate files, see below, both files do not need to have the same grid size. Typically, one needs a much higher grid accuracy for the elevation data, than for the friction data. Although not required, mind that the friction and elevation data represent the same road surface.

> **Note**: The start of the friction file is, by default, translated to the origin. This can be overruled by including an (empty) "$ROAD_CRG_MODS" block. In most cases this is required for friction to simulate correctly.

Implementation

There are 2 ways to reference the CRG file(s):

- Reference from RDF file 85
- Direct reference to CRG file 86

Note that only the first method enables friction variation.

Reference from RDF file

The reference to CRG file(s) can be set in the Road Data File, RDF. This way **friction** of the road surface can be defined as well. When no friction file is set the value of MU will be used as Road Friction Correction Factor.

The references to the CRG file(s) need to be set in the [PARAMETERS] section 77 of an RDF. Standard, the specified CRG file(s) are taken from the same directory as the RDF. For referencing to CRG file(s) in a different directory, absolute and relative paths can be used. The following parameters can be set:

MU	Road Friction Correction Factor (not the friction value itself), to be multiplied with the LMU scaling factors of the tyre model. Default setting: MU = 1.0. Note, MU will be ignored if keyword CRG_FRICTIONFILE is specified, but must be specified.
CRG_ROADFILE	Reference to the crg file containing the **height** data of the road

CRG_FRICTIONFILE	(Optional) Reference to the crg file containing the **friction** data of the road. Note, if this keyword is used the value of MU will be ignored. However, MU must be specified.

Note: The `ROAD_TYPE` needs to be set to `'crg'` in the [Model] section 76 when using the reference from RDF.

Direct reference to CRG file

A CRG file can also be directly referenced by an MBS Package 87 (except for ADAMS). MF-Tyre/MF-Swift will use this file to calculate the height of the road. Friction variation over the road surface is not possible with this method (because of limitations of the Standard Tyre Interface). The Road Friction Correction Factor (MU) will in this case be treated as being one.

4.5 Multi-Body Simulation Packages

In this chapter the use of the MF-Tyre/MF-Swift model with the supported 10 Multi-body simulation (MBS) packages is described.

Implementations by TNO

ADAMS (MSC) 87
MATLAB/Simulink (MathWorks) 92
DADS (LMS) 102

Implementations using a shared library

Recurdyn (FunctionBay) 104
CarSim/TruckSim/BikeSim (MSc) 107

Native implementations

Virtual.Lab (LMS) 108
Simpack (INTEC) 111
Madymo (TASS) 114

Note: In this manual, it is assumed that the user is familiar with the described MBS packages

4.5.1 ADAMS

MF-Tyre/MF-Swift is offered as a user programmed tyre model in ADAMS.

Note: To use the TNO tyre model you need a customised ADAMS solver. These are included in the delivery, see the ADAMS 22 section in the Installation Guide 9.

ADAMS/View

selecting the tyre model

To introduce MF-Tyre/MF-Swift in an ADAMS model using ADAMS/View commands, there are two options:

- Select Special Force: Tyre of the menu Force elements in the Main Toolbox
- Use of vpg_tyre and vpg_road in the Command Navigator

When using vpg_tyre and vpg_road, the road has to be specified first before the tyre can be modelled:

create a road:

Tools -> Command navigator -> vpg_road -> instance -> create
right click on instance name and select "vpg_road" -> "create", fill in the fields

Note: According to the Standard Tyre Interface (STI) the position and rotation of the wheel carrier centre are defined with respect to the earth fixed coordinate system.

In ADAMS the position and rotation of the wheel carrier centre are defined with respect to the **road reference marker** instead.

For correct calculations the position and orientation of this **road reference marker** must be set **equal** to the **global coordinate system**.

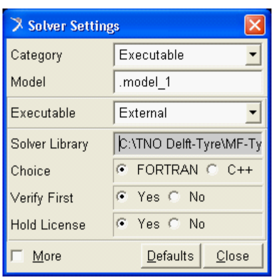
create a tyre:

Tools -> Command navigator -> vpg_tire -> instance -> create
right click on instance name and select "vpg_tire" -> "create", fill in the fields
You get a graphical representation of the tyre after closing the dialog box.

For both options, a wheel body including tyre force element is created. You will have to add a revolute joint between the wheel body and vehicle chassis component.

setting the solver

The MF-Tyre/MF-Swift solver can be selected in Solver Settings (Settings -> Solver -> Executable).



Set 'Executable' to *External*. Browse the 'Solver Library' and select 'acar_solverXX.dll' or 'private_solverXX.dll' (where XX stands for the ADAMS version) to use the acar or private solver, respectively.

For other Solver Settings, like saving output files, displaying solver messages and setting HMAX, please refer to the ADAMS manual.

ADAMS/Car

In ADAMS/Car it is sufficient to select a MF-Tyre/MF-Swift Tyre Property File 53, as low as the acar_solver.dll is located in the correct folder, see Using the private solver in ADAMS/Car 24.

4.5.1.1 Tyre Property File Format

To use the tyre model in ADAMS the following settings should be specified:

- Select MF-Tyre/MF-Swift model
- Set Operating mode
- Set Integrator

selecting the MF-Tyre/MF-Swift model

To use the MF-Tyre/MF-Swift model in ADAMS the following statements should be included in the [MODEL] section of the Tyre Property File:

```
PROPERTY_FILE_FORMAT      ='USER'
USER_SUB_ID               = 815
N_TYRE_STATES             = 4
```

This ensures that the TNO MF-Tyre/MF-Swift tyre model is called. This can also be checked in the ADAMS message file (*.msg), the following statement should appear:

```
TYR815 -> DELFT-TYRE MF-Tyre/MF-Swift 6.1 xxxxxxxx-x
```

selecting an operating mode

A four digit number (ABCD) would be required to define the operating mode (ISWITCH 48). In ADAMS the ISWITCH is split in selecting the tyre side in the graphical user interface and the other three digits in the Tyre Property File, called USE_MODE.

When defining a tyre in ADAMS via the graphical user interface the user has to identify a tyre as being "left" or "right". MF-Tyre/MF-Swift will honour this ADAMS sideflag and adjust the value for "A" accordingly. The user may overrule this by specifying the value "A" in the Tyre Property File (so USE_MODE is then a four digit number).

In ADAMS the other three digits of the operating mode are selected by setting the value of USE_MODE in the [MODEL] section of the Tyre Property File. Example:

```
USE_MODE                    = 434
```

In any case the user will get a clear feedback on the operating mode of the tyre model in the ADAMS message file (*.msg). A typical message would look as follows:

```
TYR815: tyre number    1, USE_MODE= 1434
 *tyre side    : left
 *contact      : 2D short wave length (basic functions)
 *dynamics     : rigid ring
 *slip forces  : combined
```

Tip:

If you have any unexpected results, first have a look at this message.

4.5.1.2 Miscellaneous

using a local integration scheme

Local integration can significantly speed up the simulation time and increase robustness when using rigid ring dynamics on an uneven road surface. For calculations on a level road surface without rigid ring dynamics a global integration will be faster and more accurate. Therefore in ADAMS it is advised use a:

- **global integration** for MF-Tyre: dynamics mode 47 0:2: the tyre differential equations are solved in the ADAMS solver together with the multibody equations.
- **local integration** for MF-Swift: dynamics mode 3:4: the tyre differential equations are solved locally inside the tyre model independent of the multi-body model

The parameters for the integrator are set in [MODEL] section of the Tyre Property File 53, for example:

```
HMAX_LOCAL                  = 0.00025
TIME_SWITCH_INTEG           = 0.1
N_TIRE_STATES               = 4
```

HMAX_LOCAL defines the step size of the local integrator. Too large values may result in instability and generally 0.25 ms (default) is a safe value (do not choose a value greater than 0.001 s).

TIME_SWITCH_INTEG defines the time when the switch is made from global to local integration. It is possible to have ADAMS calculate static equilibrium for the tyre model and at a later stage during the simulation switch to local integration to speed it up.

N_TYRE_STATES defines the number of states available to ADAMS to do a global integration. The default for using MF-Tyre/MF-Swift is 4, providing enough states available for MF-Tyre (dynamics mode 0 to 2) to be calculated globally, and MF-Swift (dynamics mode 3 and 4) to be calculated locally.

The ADAMS message file will provide information on the states. Some examples:

- GLOBAL integration of tyre dynamics (0/ 4): 0 states required, 4 available
- GLOBAL integration of tyre dynamics (6/30): 6 states required, 30 available
- LOCAL integration of tyre dynamics (30/ 4): 30 states required, 4 available

Note 1: When using local integration the maximum step size HMAX of the ADAMS integrator (see ADAMS manual) has to be set to 1 ms or smaller, otherwise the simulation results may become inaccurate or unstable.

Note 2: to use a global integration for MF-Swift, set N_TYRE_STATES to 30 and comment out the line defining HMAX_LOCAL from the Tyre Property File by using a $ or ! character.

4.5.1.3 Backward compatibility

To ensure backward compatibility, USE_MODE=24 is automatically replaced by USE_MODE=434, when ADAMS encounters an old SWIFT 1.2 Tyre Property File. So existing models using MF-Tyre 5.2 or SWIFT 1.2 will run without modifying the Tyre Property File.

4.5.2 MATLAB/Simulink/SimMechanics

MF-Tyre/MF-Swift is offered in MATLAB/Simulink as:

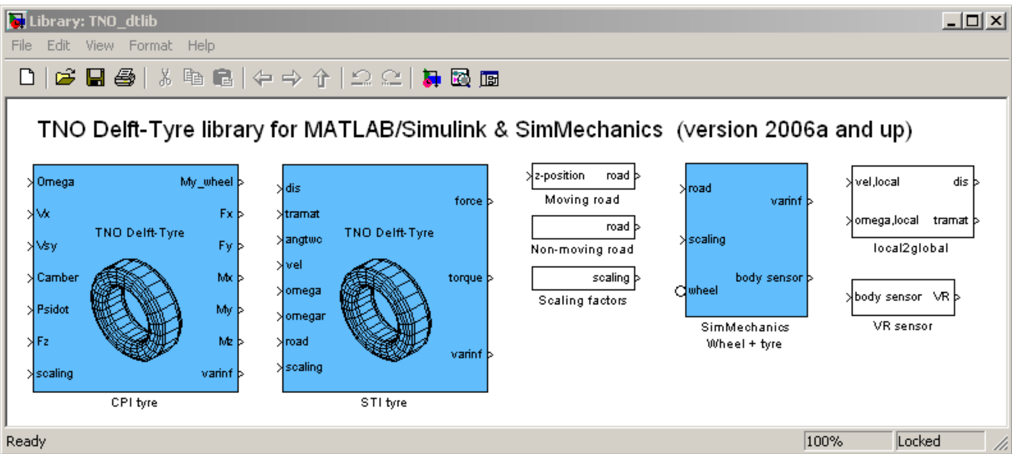
- Tyre blocks in Simulink 92
- (MATLAB) Command line function 99

MF-Tyre/MF-Swift is offered for MATLAB/Simulink 6.5 and up. A SimMechanics connection is offered for MATLAB 2006a and up.

4.5.2.1 Simulink

MF-Tyre/MF-Swift blocks are available from the library "TNO_dtlib.mdl". Three main blocks are available to the user in Simulink (see also figure below):

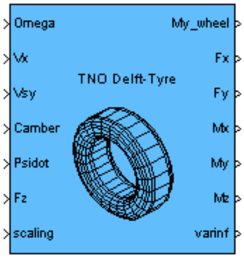
- CPI tyre 92
- STI tyre 94
- SimMechanics Wheel + Tyre 95



Furthermore, some blocks are provided to easily model moving and non-moving road surfaces, coordinate system transformation and animation of the wheel using the Virtual Reality Toolbox. See the help function of the blocks and the Simulink and SimMechanics demos for more information.

CPI Tyre

The **Contact Point Interface** evaluates the Magic Formula in the contact point. The outputs are forces (& moments) in the **contact point** (MF-Tyre 37 only)



CPI tyre

Inputs

#	Input	Description	Unit
1	*Omega (1)*	wheel angular velocity about wheel spin axis	[rad/s]
2	*Vx (1)*	forward velocity of wheel center in the wheel plane, parallel to the road surface	[m/s]
3	*Vsy (1)*	lateral velocity at road level, parallel to the road surface and perpendicular to the wheel plane	[m/s]
4	*Camber (1)*	inclination angle of the wheel with respect to the road	[rad]
5	*Psidot (1)*	tyre yaw velocity with respect to the road	[rad/s]
6	*Fz (1)*	tyre normal load	[N]
7	*scaling (7)*	scaling input 98 factors	[-]

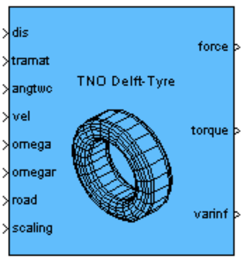
Outputs

#	Output	Description	
1	*My Wheel (1)*	resultant moment about the wheel spin axis	[Nm]
2	*Fx (1)*	longitudinal force	[N]
3	*Fy (1)*	lateral force	[N]
4	*Mx (1)*	overturning moment	[Nm]
5	*My (1)*	rolling resistance moment	[Nm]

6	*Mz (1)*	self aligning moment	[Nm]
7	*varinf (30)*	First 30 parameters of tyre model output 50: use the "bus selector" block to select the appropriate signals	[-]

STI Tyre

The **Standard Tyre Interface** is the **standard** way to couple the tyre model to the vehicle model. Based on vehicle (axle) input, forces and moments on the **axle** are calculated.



STI tyre

Inputs

#	Input	Description	Unit
1	*dis (3)*	x, y and z-coordinate of the wheel carrier in the global axis system	[m]
2	*tramat (9)*	transformation from wheel carrier axis system to the global axis system. Tramat(1..3) is the representation of the unit x-vector of the carrier axis system in the global frame. The same applies to tramat(4..6) being the unit y-vector and tramat(7..9) being the unit z-vector of the carrier axis system expressed in the global frame	[-]
3	*angtwc (1)*	rotation angle of the wheel with respect to the wheel carrier about the wheel spin axis	[rad]
4	*vel (3)*	global velocity of the wheel centre expressed in the wheel carrier local frame (x, y and z component)	[m/s]

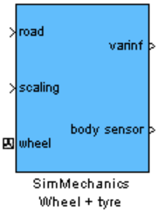
5	*omega (3)*	global angular velocity of the wheel centre expressed in the wheel carrier local frame (x, y and z component)	[-/s]
6	*omegar (1)*	relative angular velocity of the wheel with respect to the wheel carrier about the wheel spin axis	[N]
7	*road (18)*	a vector with 18 components for the specification of the motion of the road. Component (1..3) defines the position, (4..12) defines the rotation matrix, (13..15) defines the linear velocity, (16..18) defines the angular velocity. All componts are specified in the inertial reference frame	[-]
8	*scaling (7)*	scaling input 98 factors	[-]

Outputs

#	Output	Description	
1	*force (3)*	forces applied by the tyre onto the rim at the centre of the wheel	[N]
2	*torque (3)*	moments applied by the tyre onto the rim at the centre of the wheel, expressed in the carrier axis system	[Nm]
3	*varinf (40)*	tyre model output 50: use the "bus selector" block to select the appropriate signals	[-]

SimMechanics Wheel + Tyre

Tyre model block that can be attached to a SimMechanics model directly as a body. The block has extra output port for the tyre model output 50 (varinf).



Inputs

#	Input	Description	Unit
1	*road (18)*	a vector with 18 components for the specification of the motion of the road. Component (1..3) defines the position, (4..12) defines the rotation matrix, (13..15) defines the linear velocity, (16..18) defines the angular velocity. All componts are specified in the inertial reference frame	[-]
2	*scaling (7)*	scaling input 98 factors	[-]
3	*wheel*	wheel centre connection: wheel centre connection to a joint (typically a revolute joint). Note: the wheel body uses the coordinates, orientation and velocities from the joint (adjoining axis system). The assumption is made that the wheel rotates about the y-axis, so the accompanying revolute joint also needs to have its rotation axis about the y-axis	[-]

Outputs

#	Output	Description	
1	*varinf (40)*	tyre model output 50: use the "bus selector" block to select the appropriate signals	[-]
2	*body sensor*	position and rotation matrix information, typically for connection with Mathworks's Virtual Reality toolbox	

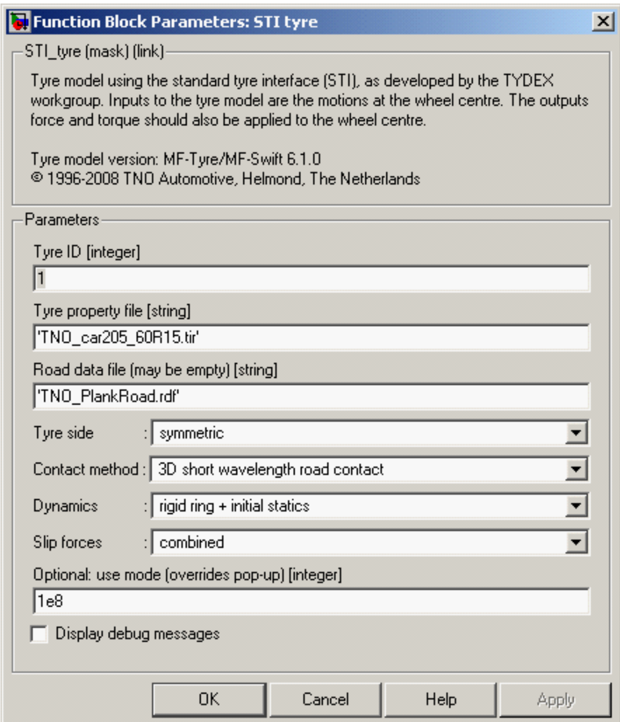
Interface

The interface of MF-Tyre/MF-Swift of all three mentioned blocks is presented below:

- Tyre ID: an integer value identifying the tyre block. This number will be used when messages related to the tyre block are printed. Please take care that the numbers within one model are unique.
- Rim inertia: (SimMechanics only): sets the mass and inertia of the rim of the

wheel. The tyre mass as specified in the tyre property file will be added automatically to the wheel mass; this can be checked by switching *Debug messages* on

- Tyre Property File 53 and Road Definition File 74.
- Main operating modes of the Tyre model: Tyre side 45; Contact Method 45; Dynamics 47; Slip forces 47.
- Optional: use mode: sets the ISWITCH 48 directly.
- Display debug messages: get additional feedback from the tyre model about mass and inertia of the wheel.



Example operating mode selection: Simulink interface for STI

Mass specification in the SimMechanics block

In the "SimMechanics Wheel + tyre" block, the complete wheel (consisting of rim and tyre) is modelled. The "wheel centre connection" port should be connected via a revolute joint to an axle body. In the mask of the "Wheel and tyre" block you specify the mass and inertia of the **rim** only, the mass and inertia of the **tyre** is obtained from the Tyre Property File. A detailed breakdown of the mass will be shown if "Display debug messages" is switched on (see figure above). For example:

```
Delft-Tyre 1 -> use_mode=1114
 wheel mass = 19.3 kg
 wheel Ixx  = 1.391 kgm2
 wheel Iyy  = 2.736 kgm2
 tyre mass  = 9.3 kg (belt mass = 7.1 kg)
 tyre Ixx   = 0.391 kgm2 (belt Ixx  = 0.326 kgm2)
 tyre Iyy   = 0.736 kgm2 (belt Iyy  = 0.636 kgm2)
```

Note: When switching on rigid ring dynamics the mass/inertia distribution is adjusted in such a way that the mass and inertia properties of the complete wheel (rim+tyre) remain unchanged.

Road input

The Road input port of the **STI Tyre** and **SimMechanics Wheel + Tyre** blocks to the **Non-moving road** block by default, or by the Moving road block when the **Moving Road** Dynamics Mode 47 is used. The input for the Moving Road block is the z-position of one of the posters.

Scaling input

In addition to the normal functionality, the Simulink and SimMechanics blocks allow a user to change tyre scaling factors 55 as a function of time or any other signal available in the model.

Vector index	Scaling factor	Description
1	LMUX	peak longitudinal friction coefficient
2	LKX	longitudinal slip stiffness
3	LMUY	peak lateral friction coefficient
4	LKY	cornering stiffness
5	LKYC	camber stiffness lateral force
6	LTR	pneumatic trail
7	LKZC	camber stiffness aligning moment

Note: The values in the **scaling vector** do not replace the scaling factors 55 from the Tyre Property File 53. The scaling factor that will be used in simulation is the **product** of the scaling factor in the tyre property file and the corresponding coefficient in the scaling vector. So, if no additional scaling - other than defined in the tyre property file - needs to be applied, a vector of [1 1 1 1 1 1 1] should be used.

Initialisation

When using "rigid ring + initial statics" the tyre model will give the following messages:

```
Delft-Tyre   1: rigid ring balancing...
 vertical tyre force     :  4721.4 N
 effective rolling radius:  0.3038 m
 angular velocity        :  32.886 rad/s (slip: -0.080 %)
```

Note: You could use this information to set the correct angular velocity of the wheel when specifying the initial conditions in your model.

Note on SimMechanics

The wheel body uses the coordinates, orientation and velocities from the joint (adjoining axis system). The assumption is made that the wheel rotates about the y-axis, so the accompanying revolute joint also needs to have its rotation axis about the y-axis.

4.5.2.2 Command line function

The Magic Formula model can be evaluated for series of input variables by typing the following command at the MATLAB command line:

```
dteval
```

For more information on dteval, please type on the MATLAB command line:

```
help dteval
```

4.5.2.3 Backward compatibility

For older versions of MATLAB (6.5 and up) the library "TNO_dtlib_v65.mdl" in Simulink can be used. The only difference with respect to the latest library " TNO_dtlib.mdl" is that SimMechanics is not supported.

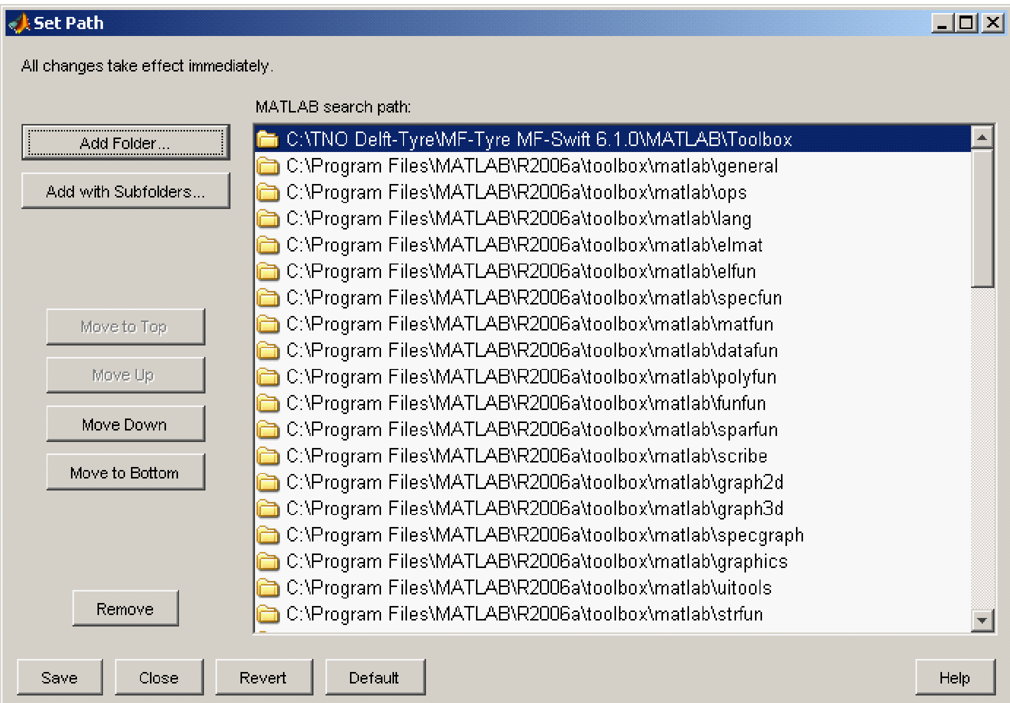
The MATLAB command line functions “mfread” and “mfeval” have been replaced by the new function “dteval”.

The sequence of the signals in the output vector (varinf) in the Simulink tyre block has changed. Please use the help function of this block to learn more about the new definition. In addition a “Bus Selector” block may be used to select the appropriate output signals based on their names.

4.5.2.4 OpenCRG

If you want to use the OpenCRG routines delivered with the DelftTyre products add the library to your MATLAB search path
Typically this directory has the name:

```
C:\TNO Delft-Tyre\MF-Tyre MF-Swift 6.1.2\OpenCRG\Matlab\Lib
```



The following steps should be taken:

- From the MATLAB menu select:
  File > Set Path…
- A new window showing the MATLAB search path will appear (see figure above).

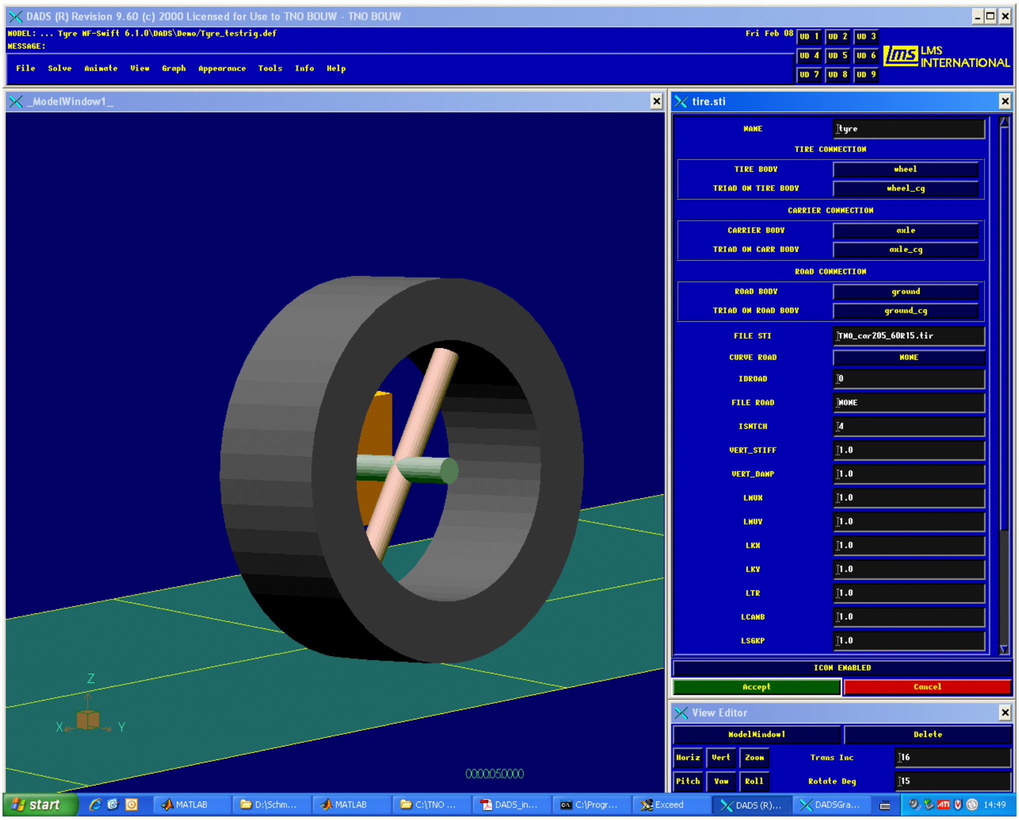
In this window click on "Add Folder…"

- Browse for the Library directory, followed by "ok" in the "Browse For Folder" window.
- Finally in the "Set Path" window click on "Save" and "Close". From now on the OpenCRG routines are available in every MATLAB session.

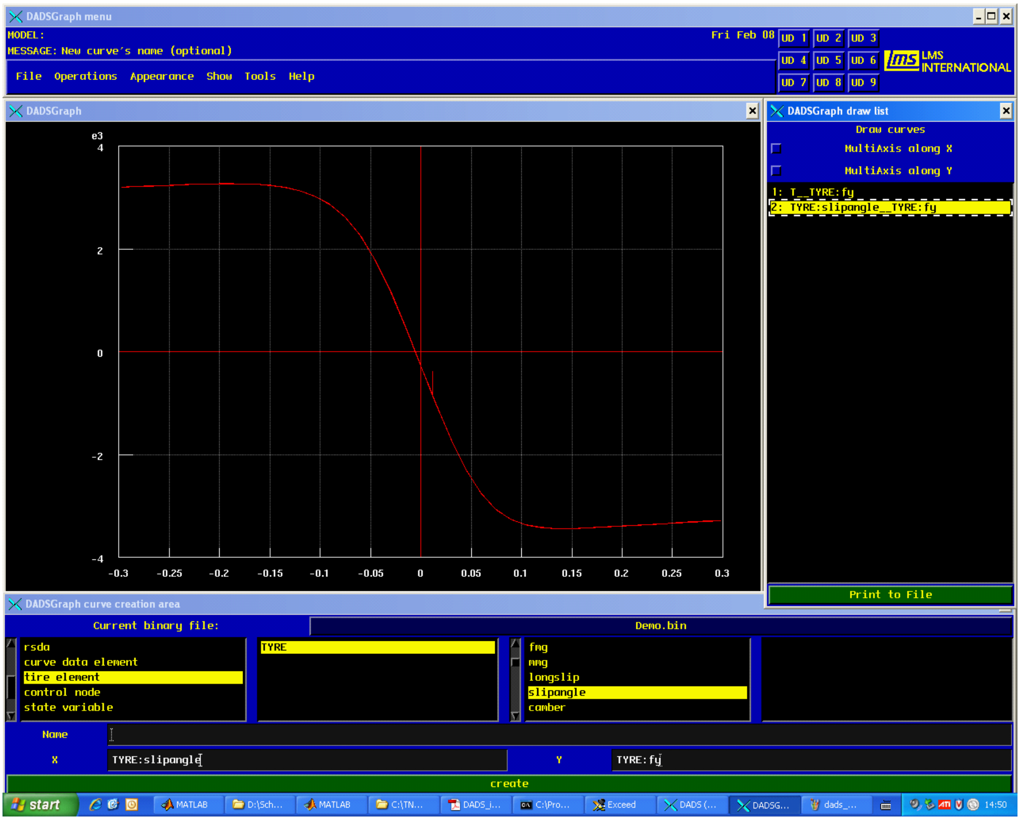
Some examples of using the tools are given in the Start Menu: All Programs > TNO Delft-Tyre > MF-Tyre & MF-Swift 6.1.2 > OpenCRG > Matlab demo's.

4.5.3 LMS DADS

MF-Tyre/MF-Swift is offered for DADS 9.6. To introduce the tyre model and to change the tyre model settings (Tyre Property File, scale factors, etc.) in the DADS GUI, select Force, Tire, STI in the DADS modelling panel:

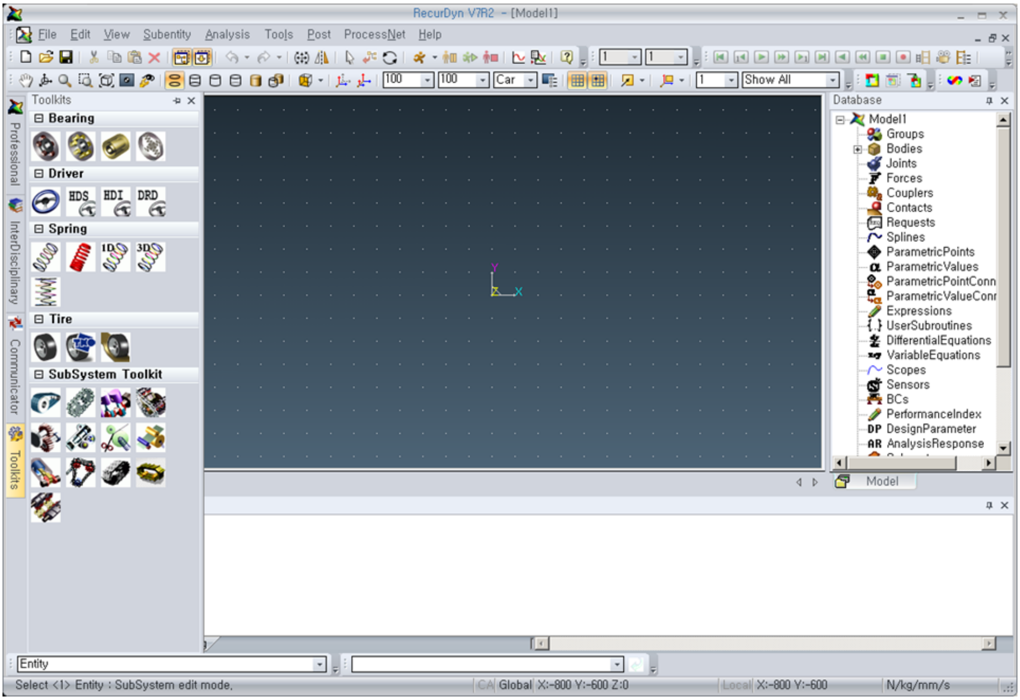


To plot the tyre model outputs after having performed a simulation, open the DADSGraph menu and select "tire element" and the signal you want to plot:

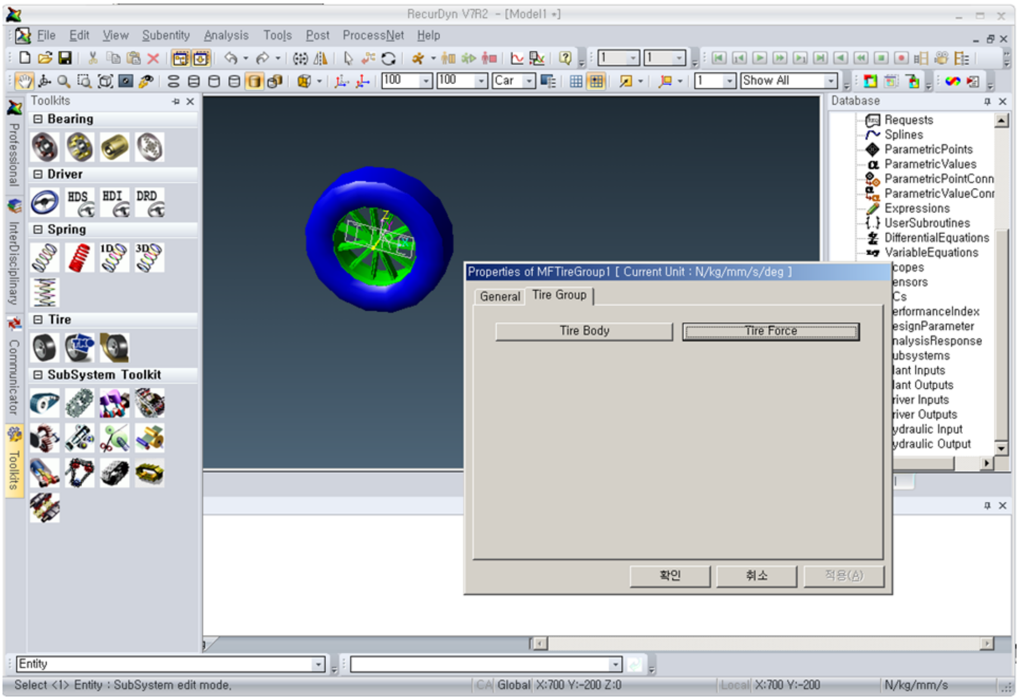

DADSGraph menu
MODEL:
MESSAGE: New curve's name (optional)
Fri Feb 08
UD 1 UD 2 UD 3
UD 4 UD 5 UD 6
UD 7 UD 8 UD 9
LMS INTERNATIONAL
File Operations Appearance Show Tools Help
DADSGraph
e3
4
2
0
-2
-4
-0.3 -0.25 -0.2 -0.15 -0.1 -0.05 0 0.05 0.1 0.15 0.2 0.25 0.3
DADSGraph draw list
Draw curves
MultiAxis along X
MultiAxis along Y
1: T__TYRE:fy
2: TYRE:slipangle__TYRE:fy
Print to File
DADSGraph curve creation area
Current binary file:
Demo.bin
rsda
curve data element
tire element
control node
state variable
TYRE
fmg
mmg
longslip
slipangle
camber
Name
X
TYRE:slipangle
Y
TYRE:fy
create
start
MATLAB
D:\Sch...
MATLAB
C:\TN...
DADS_i...
C:\Pro...
Exceed
DADS (...
DADSG...
dads_...
14:50

4.5.4 Recurdyn

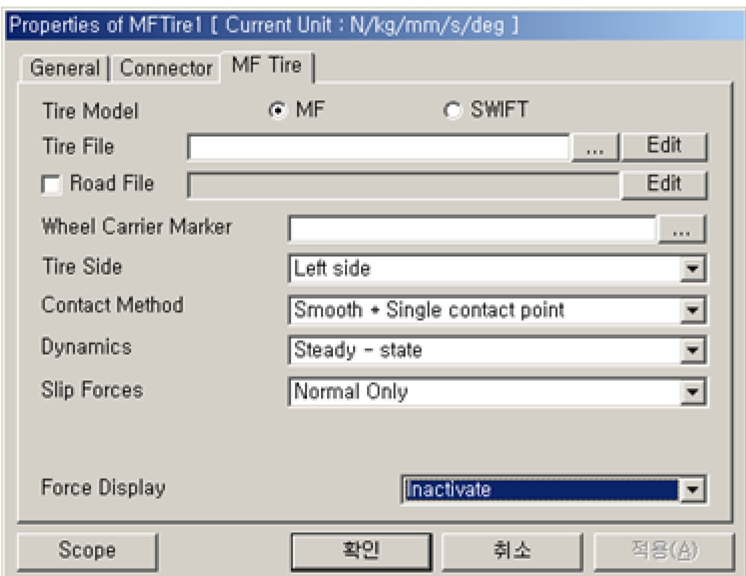
In **Recurdyn**, the MF-Tyre/MF-Swift model can be selected in the menu **Toolkits** under **Tire**. Select the model with the TNO logo.



In the **Properties** window of the tyre select **Tire Force** ...



... to show the MF-Tyre/MF-Swift properties.

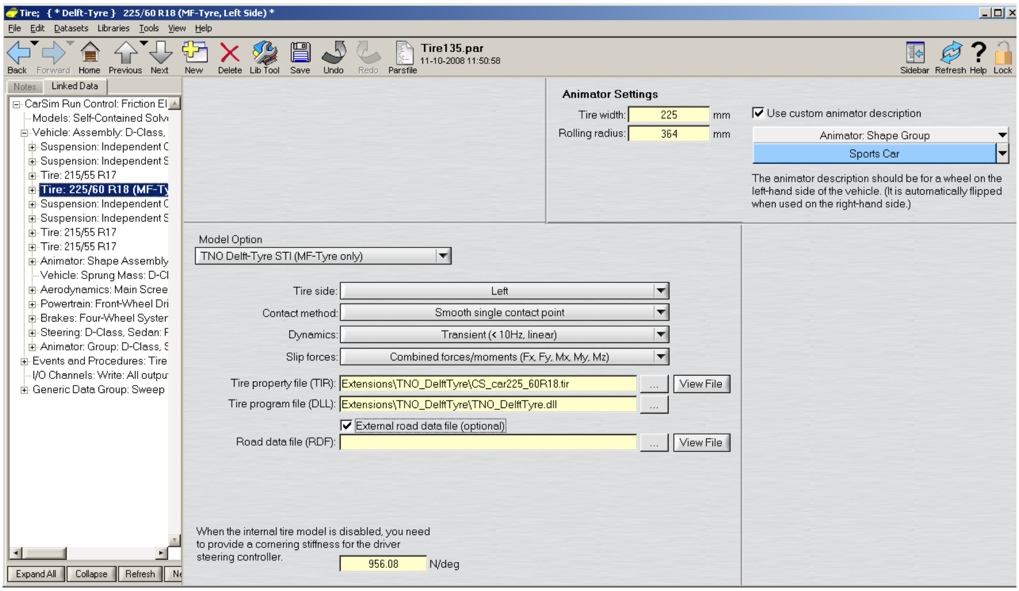


The following MF-Tyre/MF-Swift parameters can be set.

Parameter	Description
Tire Model	Select MF-Tyre operating modes 43 or MF-Swift operation modes 43 for *Dynamics*
Tire File	Select Tyre Property File 53.
Road File	Select Road Data File and if it should be used
Tire Side	Select Tyre Side 45
Contact Method	Select Contact Method 45
Dynamics	Select Dynamics 47
Slip Forces	Select Slip Forces 47

4.5.5 CarSim / TruckSim / BikeSim

In **CarSim**, **TruckSim** and **BikeSim**, the MF-Tyre/MF-Swift model can be selected in the window **Tire** under **Vehicle Assembly**.



The following MF-Tyre/MF-Swift parameters can be set.

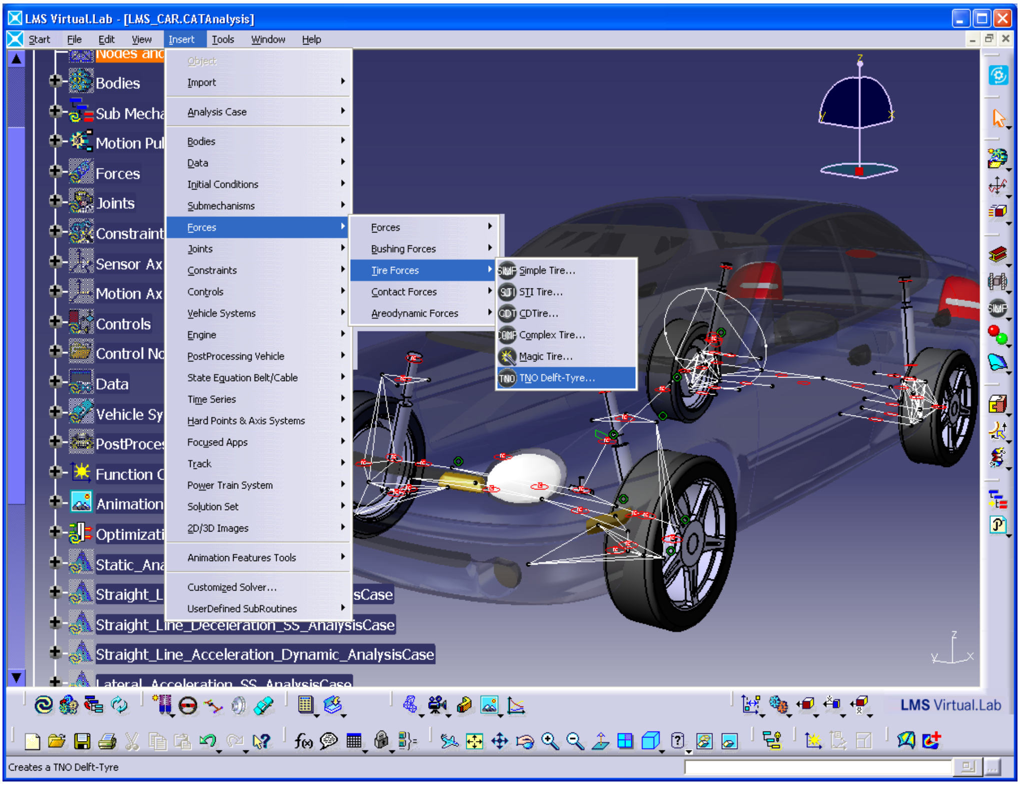
Parameter	Description
Model Option	Select *TNO Delft-Tyre (MF-Tyre only)* for MF-Tyre operating modes 43 or *TNO Delft-Tyre (MF-Swift)* MF-Swift operation modes 43 for *Dynamics*
Tire Side	Select Tyre Side 45
Contact Method	Select Contact Method 45
Dynamics	Select Dynamics 47
Slip Forces	Select Slip Forces 47
Tire Property File	Select Tyre Property File 53.
Tire Program File (DLL)	Select `TNO_DelftTyre.dll` in directory `Extensions\TNO_DelftTyre`
(External) Road Data File	Select Road Data File 74 and if it should be used

Note: The CarSim **Tire Tester** supports only **steady-state** operating mode 45.

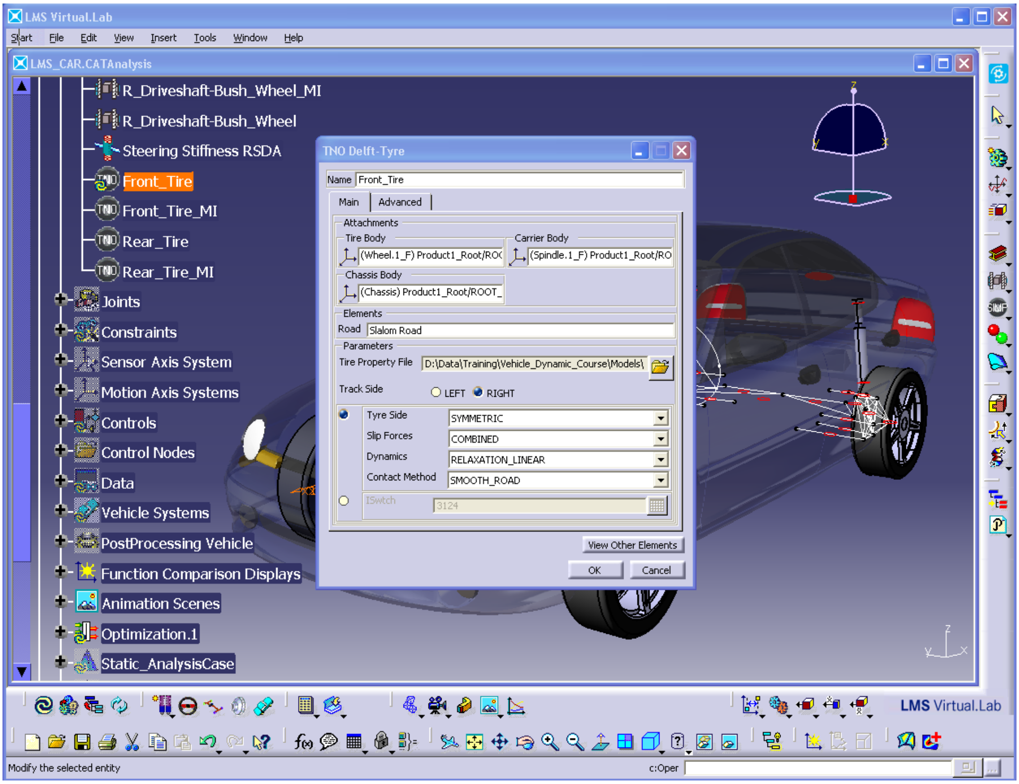
Known issue: The time-step of the simulation should be chosen small enough (see Simulation Guidelines 41) for the simulation to produce correct results.

4.5.6 Virtual.Lab

In **Virtual.Lab**, the MF-Tyre/MF-Swift model can be selected in the **Insert** menu through **Forces** > **Tire Forces** > **TNO Delft-Tyre...**



The following MF-Tyre/MF-Swift parameters can be set, as indicated in the figure below.



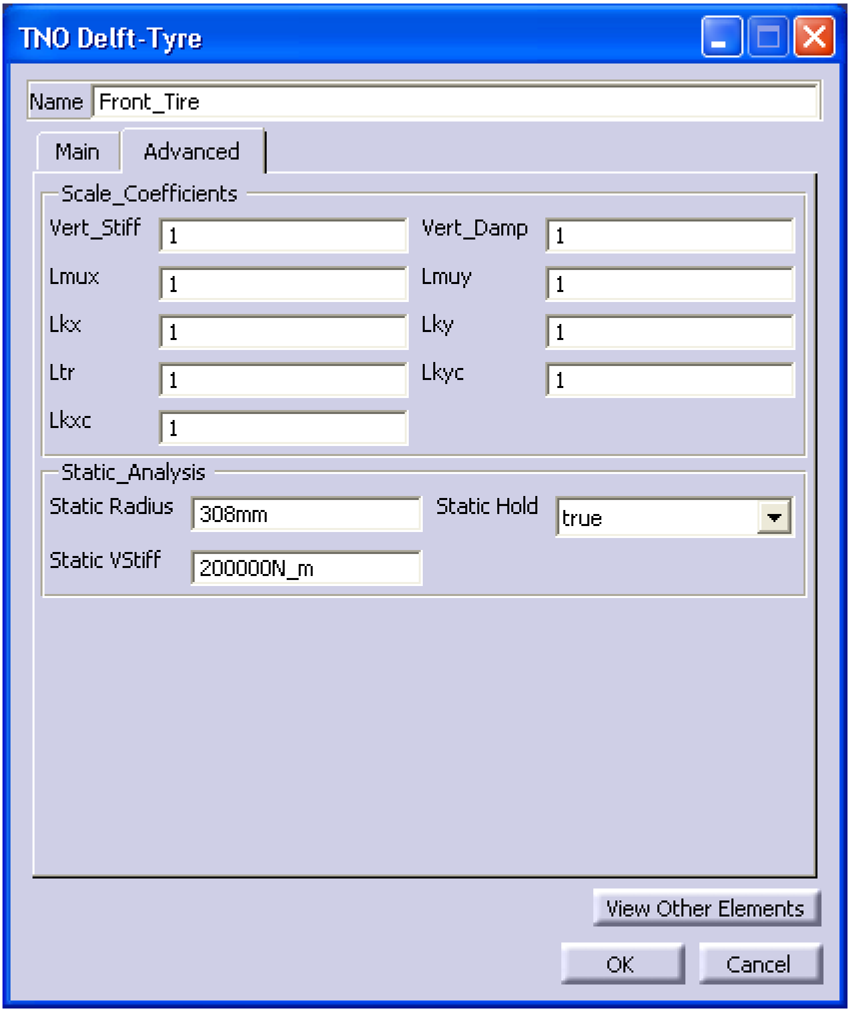
Main tab

Parameter	Description
Road	Select Road Data File 74
Tire Property File	Select Tyre Property File 53.
Slip Forces	Select Slip Forces 47
Tyre Side	Select Tyre Side 45
Dynamics	Select Dynamics 47
Contact Method	Select Contact Method 45

Note: One may set the model properties also directly through the ISwitch 48 variable.

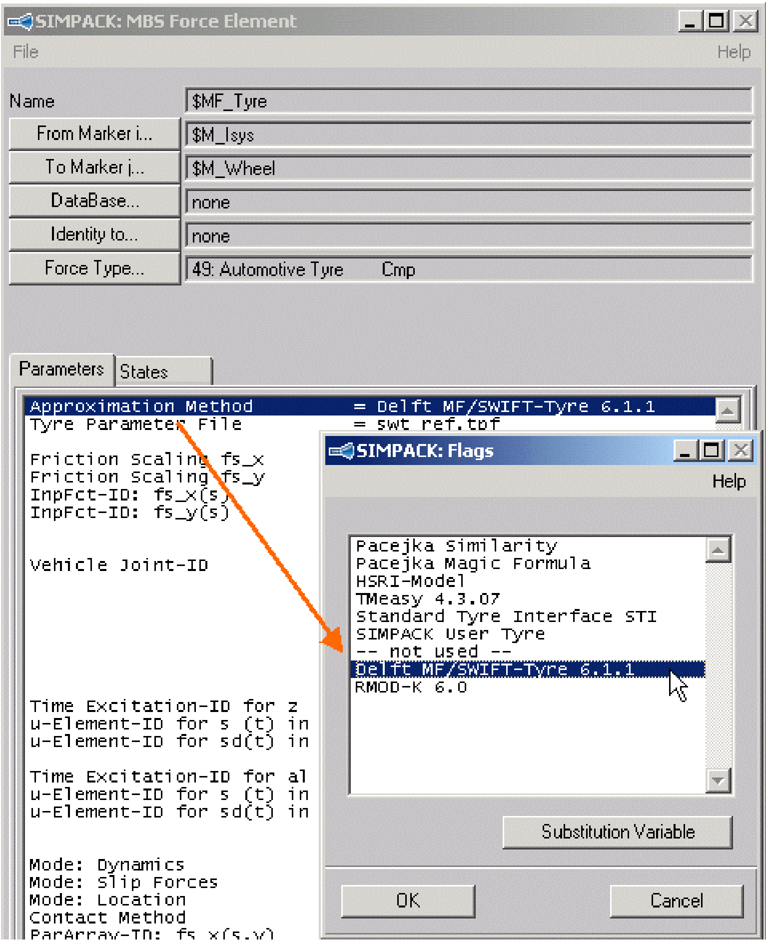
Advanced tab

In the advanced tab, Scale Factors 55 may be set.

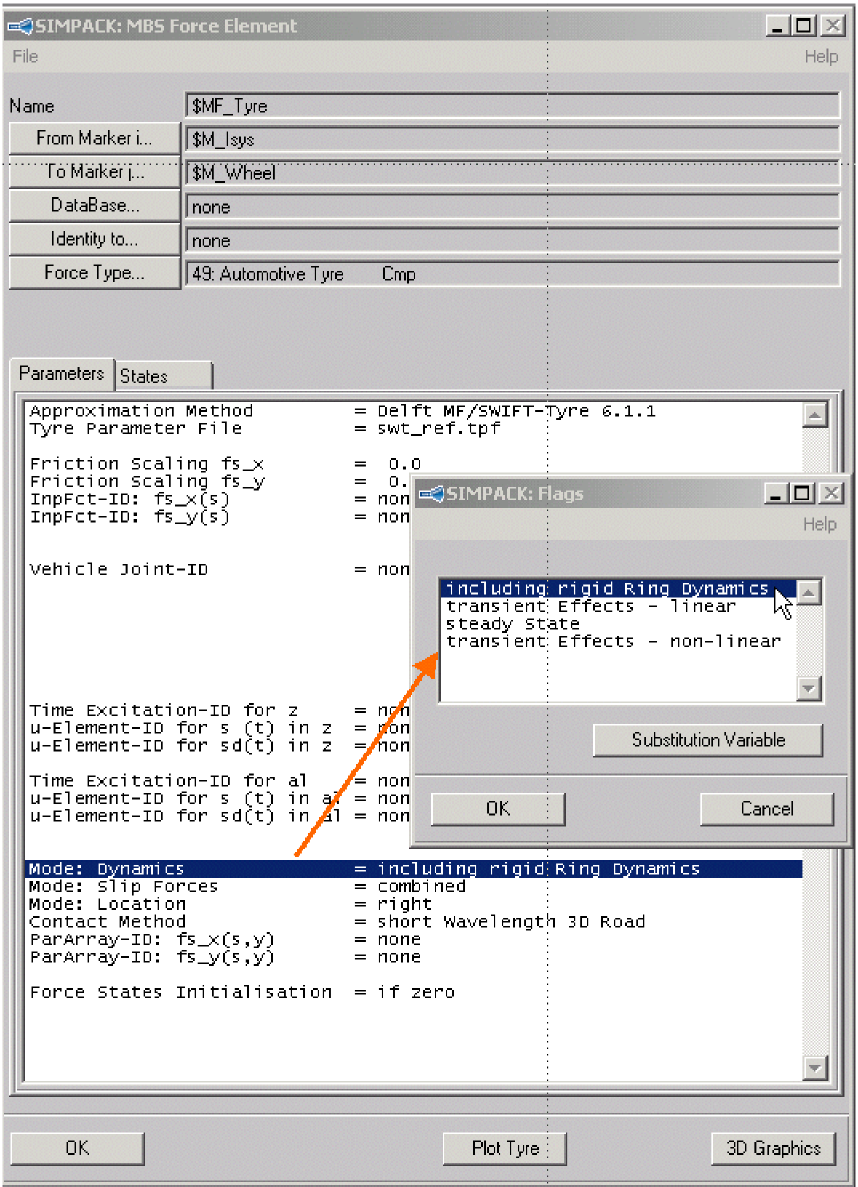

TNO Delft-Tyre
Name
Front_Tire
Main
Advanced
Scale_Coefficients
Vert_Stiff
1
Vert_Damp
1
Lmux
1
Lmuy
1
Lkx
1
Lky
1
Ltr
1
Lkyc
1
Lkxc
1
Static_Analysis
Static Radius
308mm
Static Hold
true
Static VStiff
200000N_m
View Other Elements
OK
Cancel

4.5.7 Simpack

In **SIMPACK** the MF-Tyre/MF-Swift model can be selected in the property window of the **MBS Force Element**. Double-click on the line containing **Approximation Method** and select *MF-Tyre/MF-Swift 6.1.2*.



The following MF-Tyre/MF-Swift parameters can be set.



Parameter	Description
Tyre Parameter File	Select Tyre Property File 53.
Mode: Location	Select Tyre Side 45
Mode: Contact Method	Select Contact Method 45

Mode: Dynamics	Select Dynamics 47
Mode: Slip Forces	Select Slip Forces 47

4.5.8 MADYMO

In **MADYMO**, the MF-Tyre/MF-Swift model needs to be defined using the following keywords:

- `CONTACT.TYRE_ROAD`: definition of contact and operating modes 45;
- `TYRE`: definition of tyre representation and selection of Tyre Property File 53;
- `ROAD`: definition of road profile

CONTACT.TYRE_ROAD

In the **CONTACT.TYRE_ROAD**, the MADYMO **TYRE** SYSTEM.MODEL and **ROAD** SYSTEM.REF_SPACE are selected and the operating modes 45 of the MF-Tyre/MF-Swift are specified, see table below.

Parameter	MADYMO options	Description
TYRE_LOCATION	Tyre location:	Select Tyre Side 45
	• LEFT	• tyre is mounted on the left side of the car
	• RIGHT	• tyre is mounted on the right side of the car
	• SYMMETRIC (default)	• symmetric tyre characteristics
	• MIRRORED	
ANALYSIS_TYPE	Contact method:	Select Contact Method 45
	• SINGLE_POINT (default)	• smooth road contact, single contact point
	• CIRCULAR	• smooth road contact, circular cross section (motorcycle tyres)
	• 2D_CONTACT	• road contact for 2D roads (using traveled distance)
	• 3D_CONTACT	• road contact for 3D roads
TYRE_TYPE	Tyre type:	Select Dynamics 47
	• STEADY_STATE (default)	• Steady-state evaluation (< 1 Hz)
	• TRANSIENT	• Transient effects included, tyre relaxation behaviour (< 10 Hz, linear)
	• DYNAMIC	• Transient effects included, tyre relaxation behaviour (< 10 Hz, nonlinear)

LOAD_TYPE	Applied load type:	Select Slip Forces 47
	• NORMAL (default)	• no Magic Formula evaluation (Fz only)
	• LONGITUDINAL	• longitudinal forces/moments only (Fx,My)
	• LATERAL	• lateral forces/moment only (Fy,Mx,Mz)
	• UNCOMBINED	• uncombined forces/moment (Fx,Fy,Mx,My,Mz)
	• COMBINED	• combined forces/moment (Fx,Fy,Mx,My,Mz)
	• COMBINED_TURN_SLIP	• combined forces/moment (Fx,Fy,Mx,My,Mz) + turnslip

TYRE

The **TYRE** contains the reference to the **TYRE.DATA** and **SURFACE.CYLINDER**.

TYRE.DATA

In **TYRE.DATA**, the Tyre Property File 53 is specified via the keyword **FILE**.

SURFACE.CYLINDER

The **SURFACE.CYLINDER** can be specified through the keywords **SEMI_AXIS** and **DEGREE**. CRDSYS_OBJECT and CHAR do not need to be specified.

ROAD

ROADs may be specified in three ways, see [MADYMO Reference Manual] for details:

- **ROAD.ANALYTIC**
  Four predefined road types: Bump, Ramp, Flat, Wave.

- **ROAD.MESH**
  Road profile described by a mesh of an FE model.

- **ROAD.USER**
  Road profile described by a user-written subroutine 74.

4.6 References

[1] Pacejka, H.B.: "Tyre and Vehicle Dynamics", Second Edition, Butterworth-Heinemann, Oxford, 2006.

[2] Pacejka, H.B., I.J.M. Besselink: “Magic Formula Tyre model with Transient Properties”, Supplement to Vehicle System Dynamics, Vol. 27, pp. 234-249, 1997.
[3] Zegelaar, P.W.A., “The Dynamic Response of Tyres to Brake Torque Variations and Road Unevenesses”, dissertation, Delft University of Technology, The Netherlands, 1998.
[4] Maurice, J.P., “Short Wavelength and Dynamic Tyre Behaviour under Lateral and Combined Slip Conditions”, dissertation, Delft University of Technology, The Netherlands, 1999.
[5] Schmeitz, A.J.C., “A Semi-Empirical Three-Dimensional Model of the Pneumatic Tyre Rolling over Arbitrarily Uneven Road Surfaces”, dissertation, Delft University of Technology, Delft, The Netherlands, 2004.
[6] Besselink, I.J.M., H.B. Pacejka, A.J.C. Schmeitz, S.T.H. Jansen: “The SWIFT tyre model: overview and applications”, Presented at the AVEC 2004: 7th International Symposium on Advanced Vehicle Control, 23-27 August 2004.
[7] A. Riedel, J.J.M. van Oosten: “Standard Tyre Interface, Release 1.4”. Presented at 2nd International Colloquium on Tyre Models for Vehicle Dynamics Analysis, February 20-21 1997. Issued by the TYDEX - Working group.
[8] TNO Automotive: “MF-Tool 6.1 Users Manual”, TNO Automotive, The Netherlands, 2008.

Index

- M -

- O -

- P -

- R -

- S -

- T -

- V -

- W -