cosin scientific software

cosin scientific software AG

Hans-Stützle-Str. 20
81249 München
GERMANY

info@cosin.eu
www.cosin.eu

FTire/fit

Measurement-Based FTire Parameterization

Documentation and User's Guide

Contents

Document Revision: 2022-1-r26108

Preface

This document describes the Flexible Structure Tire Model (**FTire**) parameterization tool **FTire/fit**. For more information about **FTire**, and other **cosin** tire simulation tools, please visit cosin.eu.

1 Legal Notices

2 FTire/fit Software Installation Procedure

FTire/fit is part of the **cosin** software installation package, to be downloaded from cosin.eu. Execute the downloaded installer and follow the installation prompts on the screen. After installation, provided a license key is available, **FTire/fit** can be launched by left-clicking the appropriate button in the **cosin** software main menu.

3 Introduction to FTire Parameterization using FTire/fit

FTire/fit is the parameter identification and validation toolbox for **FTire**. **FTire/fit** uses the following kinds of experimental and/or FE results:

- **footprint shape bitmaps**,
- **static tire properties** like radial, longitudinal, lateral, and torsional stiffness on different surface geometries,
- **deformation of the tire carcass** due to different contact and loading conditions,

- tire properties in **steady-state rolling conditions**, and
- time- and/or frequency-domain measurements of **cleat tests** that excite large-amplitude tire vibrations,

to fit some, or all, **FTire** model data for a given tire brand, tire size, and one or more inflation pressure(s). This is done such that the measurements are approximated by **FTire** simulations in the sense of minimized mean squared deviations.

The data to be fitted can be defined in a very flexible way. Thus, the actual availability and properties of these measurements can easily be taken into account. At the same time the model parameters, to be fitted, can be chosen with a similar flexibility, starting from one or only few parameters up to a nearly complete set. For most types of measurements, **FTire/fit** can automatically adjust single parameters in such a way that the least squares difference between the measurement and simulation is minimized. The 'expert knowledge' about the relationship between type of measurement and parameters, that can be determined on basis of this measurement, is 'hard-coded' into **FTire/fit**.

During the development process of **FTire**, an important experience was made about the **role of modal data**. They seem to contain less relevant information than static measurements (in contrast to what has been assumed in the early days of **FTire**), and are also rather laborious to obtain. There is one obvious cause for this lack of relevant information: during modal measurements on an unloaded tire, only small deformation amplitudes will be reached. However, these measurements are then used to parameterize the respective tire model for load cases with large to extreme deflection values, that is, for completely different operating conditions.

Another important experience was made about the large amount of valuable information that is contained in **footprint bitmaps**. The same holds for several kinds of static deflection curves, without and with camber angle, on a flat surface or on certain well-defined obstacles. Moreover, handling properties, like cornering stiffness and pneumatic trail, show a high correlation with certain out-of-plane stiffness data. In many cases, after a thorough analysis of static and steady-state behavior, there remain only few dynamically relevant parameters to be adjusted in order to get also a good correlation for the cleat tests.

The general parameterization procedure, as described in the following section and summarized in figure 1, has proven to be successful in many cases, and is well supported by **FTire/fit**:

1. Prepare the identification process:

   a) create a new data file with a special variant of **FTire/estim**, by specifying tire and rim size, load index, speed range, mass, and inflation pressure(s) as well as manufacturer and brand information. Select a reference tire which is as close as possible to the new tire. **FTire/estim** is launched with a single mouse-click from within the **FTire/fit** GUI;

   b) specify all drum diameters and cleat geometries used during the measurements. Alternatively the cleat geometry and drum dimensions can also be defined in the TYDEX data files. **FTire/fit** will automatically update the cleat definition file if the test surface does not match any previously defined test surface. **FTire/fit** provides example road data files and functions to manage such obstacle-defining files;

   c) specify ('check-in') all available footprint bitmap files at different loads, camber angles and road surfaces. **FTire/fit** will automatically calibrate these files and save the relevant information for later identification and/or validation.

   d) Import or digitize the tread and carcass contour geometry data. **FTire/fit** provides a respective digitizing tool. If a cross-section image is found, the user will have to prepare it in a way explained below, using an image processing software, like MS-Paint, Paintbrush, Gimp, etc.

e) specify ('check-in') all static, steady-state, and dynamic measurement files used in the sequel. If these files are given in the TYDEX file format , a single mouse-click to check them in is sufficient in many cases. **FTire/fit** will automatically recognize what kind of measurement they contain, will determine constant operating conditions, like inflation pressure, wheel load, camber angle, etc., and will save the information on how the validation and/or identification is to be performed. Moreover, depending on the kind of measurement, **FTire/fit** will occasionally extract relevant information like radial, longitudinal, lateral, and torsional stiffness, cornering stiffness, slip stiffness, pneumatic trail, camber thrust, sliding friction, etc., and insert this information into the tire data file. In some cases these parameters are considered merely as optional 'nominal' values. In such a case, it is left up to the user whether or not the value is actually to be used. The user can make this decision later by using **FTire**'s data file editor *cosin/tools*, which is directly accessible from **FTire/fit**'s GUI. If the files are given in any other ASCII file format, **FTire/fit** will assist in importing the files and then create an **FTire/fit**- and Matlab-readable file (an mtl-file) out of it. If measurements are only given in terms of scanned images, **FTire/fit** provides a digitizing tool to convert the data to mtl-files;

2. Identification and/or validation of FTire's mass, stiffness, damping, friction, and belt kinematics data that can be seen as 'basic' data in an FTire data file :

   a) Identify dynamic rolling radius on basis of the measurement of the angular velocity of a freely rolling tire at different drum speeds and wheel loads (or roughly estimate the rolling radius by subtracting tread gauge from maximum unloaded radius).

   b) vertical stiffness on a flat surface (which is merely a validation of the two deflection values for half and full LI load that have been automatically inserted in the data file in step (1e)). The respective simulation will be prepared by **FTire/fit**; a single mouse-click is sufficient to launch a validation simulation and save all results for later report generation. If the actual stiffness deviates from the predicted one, adjust the respective deflection values accordingly. This might happen if there is a discrepancy between the static and the steady-state simulations, caused by different treatment of hysteresis and friction properties;

   c) longitudinal and lateral stiffness on a flat surface (same remark as in 2b);

   d) torsional stiffness (turning the standing tire about the vertical axis). If the simulation results deviate from measurements, adjust the belt torsional stiffness about radial axis accordingly;

   e) vertical stiffness on a longitudinal and transversal cleat. Adjust lateral belt bending stiffness and belt in-plane bending stiffness if the simulation deviates from measurement;

   f) vertical stiffness at large camber angle on a flat surface and on a transversal cleat. Adjust belt torsion and twist stiffness about circumferential axis if the simulation deviates from measurement;

   g) footprint size and shape at different wheel loads and camber angles. **FTire/fit** provides an automatic simulation preparation and a post-processing tool to superimpose the simulated footprint boundary over the measured contact patch bitmap. Again, all this is done by a single mouse-click. If there is a mismatch in size or shape, adjust the in-plane and lateral bending stiffnesses. Run (2e) again and find a compromise.

   h) longitudinal slip stiffness. Either activate the measured nominal value directly, by replacing tread rubber stiffness, or identify tread rubber stiffness manually;

   i) cornering stiffness and pneumatic trail. Either activate the measured nominal values directly, by replacing lateral stiffness and out-of-plane bending stiffness, or (re-)identify these two values manually. If there is a discrepancy to the value of lateral stiffness determined in (2c), find a compromise.

j) Identify sliding friction coefficients. During check-in of the measurement files, **FTire/fit** has automatically collected all available and relevant test cases. Ideally, this identification is performed by one mouse-click only. After the identification, validate the relevant measurement cases. If the identified values of stiction and sliding friction differ considerably, stick-slip-phenomena might occur in lateral and longitudinal stiffness simulation. In this case, find a compromise by relaxing the differences in friction coefficients. **FTire/fit** automatically detects and collects the relevant information from the following tests:

   i. pulling the tire in longitudinal and

   ii. in lateral direction,

   iii. turning the tire about vertical axis, and

   iv. running at large longitudinal slip or

   v. at large slip angles.

k) in-plane cleat-test identification (or just a validation) cycle, determining few more parameters like the percentage of free mass, the structural damping (expressed in terms of the modal damping), longitudinal coupling of tread shear stiffness, tread rubber damping, etc.,

l) out-of-plane cleat-test identification (or just a validation) cycle, determining the few remaining parameters like conicity, modal out-of-plane damping, the coupling between belt torsion and lateral displacement, etc.

3. finishing tasks, comprising of

   a) optimization of certain numerical data to increase convergence speed during pre-processing;

   b) packing of project files for distribution;

   c) and generation of the report, by collecting all images and other information, and combining them with other information and explanatory text.

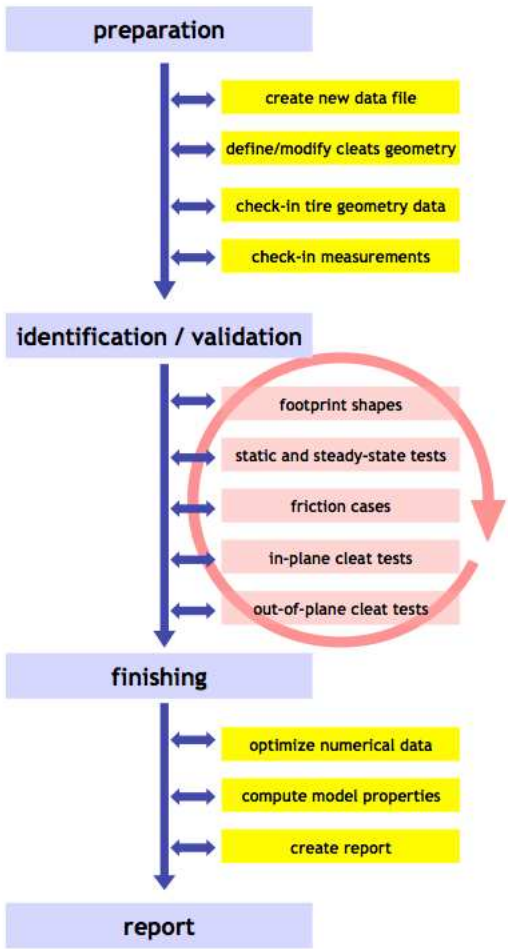

Figure 1: **FTire/fit** workflow

In the following chapters these steps and the respective program invocations are explained in greater detail.

4 Working with FTire/fit

Figure 2 shows the **FTire/fit** workbench GUI (Graphical User Interface). The **FTire/fit** workbench contains all buttons to control the parameterization process, and to invoke all other assisting tools. In each stage of the parameterization process the layout of the workbench will automatically update, so that only the relevant fields and buttons are visible.

In-program support is provided by **'tool-tips'** which appear when the cursor is hovered over a button or selection area. These **'tool-tips'** will display important information about the underlying field or will describe the command that will be executed when a button is pressed. You may deactivate tool-tips display in case of too much interference with your work.

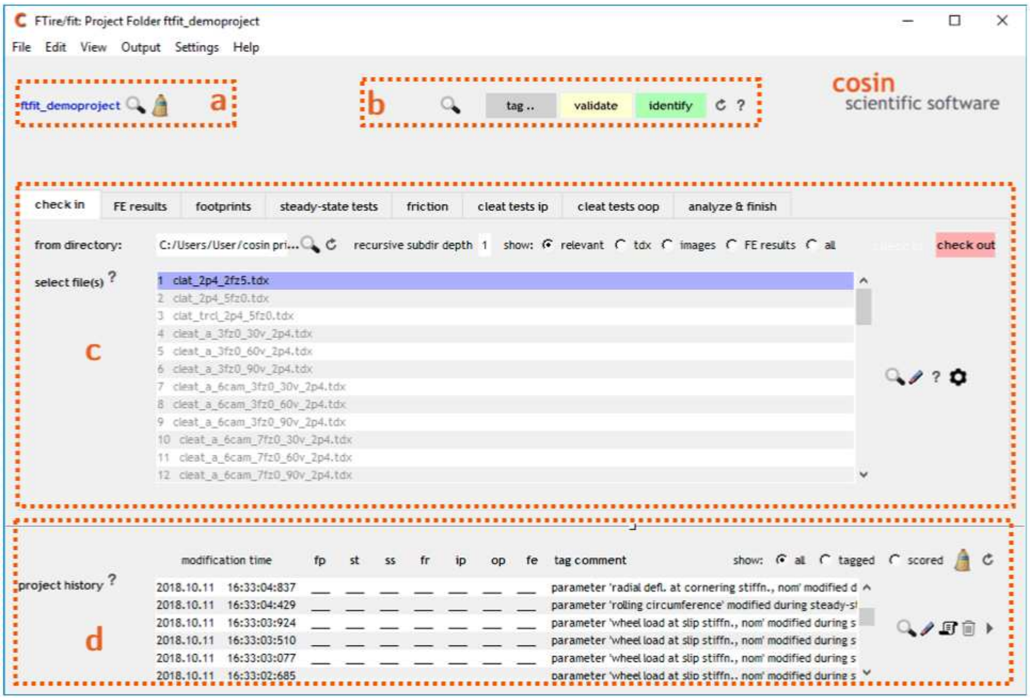

Figure 2: **FTire/fit** workbench

The top right corner of the workbench displays the current project name (**section a** of figure 2) The (**loupe** button), next to the project name, will open the current project folder in a new window. The (**clean** button) can be used to clean the current project folder. If selected, all temporary files will be deleted from the folder.

Section b (of figure 2) consists of a set of buttons that can be used to access the current data file and to perform various project tasks. Selecting the (**loupe** button) will open the current data file (_best_so_far.tir) in **cosin/tools**. The tag .. (**tag** button) will save the current state of the project and add a new tagline to the project history. The validate (**validate** button) can be used to run a validation of all 'checked-in' measurements, that are currently selected, and to compile the results into a **FTire Modeling and Validation Report**. The identify (**identify** button) can be used to perform an completely automatic FTire parameter identification. This process might take several hours. In principle, the whole process could be run fully automatically without any, or with only a minimal amount of, user interaction. Such an interaction, for example, would be inevitable if tire or measurement information is incomplete or cannot be extracted automatically from the given measurement files. In most cases however, it is suggested to work in an interactive mode. This gives the user the opportunity to occasionally influence the compromises that need to be found. Such compromises are necessary if different measurement information is not fully consistent, say. The (**refresh** button) can be used to refresh the menu after a validation or identification process. The ? (**check** button) can be used to determine which tasks can be preformed in the tabs below.

The launch-pad (**section c** of figure 2) is organized into various categories, which can be accessed through the tabs in the top area of the launch-pad. The first tab is the '**check-in**' tab which holds a selection of tools to import measurements into the project. The measurement tabs, in the middle, are dynamically created/removed depending on the test data that is currently 'checked-in'. The last tab is the '**analyze & finish**' tab which holds a selection of tools for final processing as well as a report generation tool.

The project history (**section d** of figure 2) will list all progress and changes that were made to the current project. Additional tools can be accessed by 'right-clicking' on a selected entry in the project history list. The project history also shows the validation group scores for various test categories. A score of a 1000 indicates that the model is capable to perfectly replicate the response of the measurement category. The measurements are divided into the following categories:

- **'fp'**: footprints;
- **'st'**: static tests;
- **'ss'**: steady-state tests;
- **'fr'**: friction tests cases;
- **'ip'**: dynamic in-plane cleat tests;
- **'op'**: dynamic out-of-plane cleat tests;
- **'fe'**: finite element tests.

4.1 Preparation Tasks

4.1.1 Creating a New Project

Select *File* → *New project ..* to create a new parameterization project. You will be guided through the following steps:

- specifying a directory containing the measurement files to be used,
- creating a initial data file using **FTire/estimate**. This data file is called **_best_so_far.tir** and will be continuously improved during the course of parameterization,
- specifying a directory to contain all project-relevant files (the **project folder**),
- specifying the so-called **cleat definition file**, which describes geometry and stiffness properties of the testing equipment,
- checking in measurement files from this or other directory(ies), and
- validate initial data file.

4.1.2 Creating a First Tire Data File

In step 2 of the project creation, first a file selection window will pop up (figure 3), asking to select a **reference data file**. **FTire/estimate** will use this data file to roughly estimate the basic mass, stiffness, damping, and other parameters of the new tire file, based upon changes in size, weight, and inflation pressure between reference file and new file.

To create a good initial guess for the subsequent identification process, the size of the tire in the reference file should be as close as possible to the one of the new file.

Size, load index, speed symbol, rim width, manufacturer, brand name, mass, and inflation pressure are to be entered in the window which opens after specification of the reference data file (figure 3). The blue numbers show the respective values of the reference data file.

On the right-hand side of the window there are some useful tools , like unit converters and tire specification tables for load and speed indices.

Experienced users might wish to **customize the estimation formulae**, using the **f(x)** button in the bottom left corner, to make them better match own experiences. These estimation formulae are used in **FTire/estimate**.

Document Revision: 2022-1-r26108

A menu to modify the formulae pops up after clicking the respective button. However, please note that such adaptations are unnecessary if enough experimental data for the new tire are available. Most parameters will be corrected later anyway, during the fitting process.

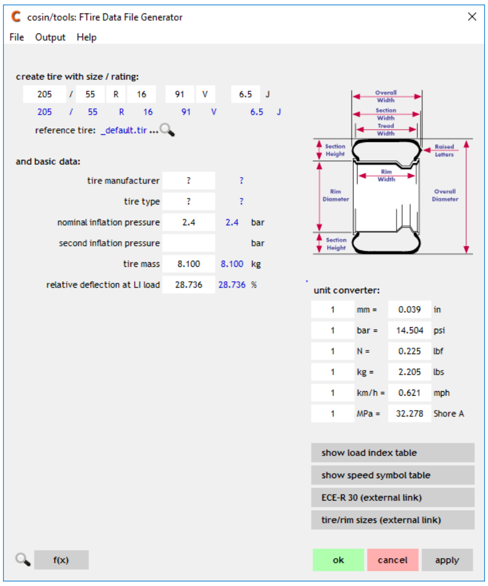

Figure 3: **Ftire/estimate** window to create a first guess of the new data file

Having entered all values, click 'OK' to close the window and return to the **FTire/fit** GUI. In the subsequent identification, the newly created **_best_so_far.tir** will always contain the latest ('best') version of the data file. Whenever **_best_so_far.tir** is updated, the previous data status will be saved in the working directories subfolder **'history'.** The name of these backup files contains the generation time, which helps you to easily reset to an older parameterization process state if necessary.

4.1.3 Checking-in the Cleat Definition File

After having created the data file, you need to specify a second file describing all obstacle geometries used in your identification. This file is called **the 'cleat definition file'**. The **FTire/fit** installation contains a very general example for such a cleat definition file ('demo_cleat.clt'), which can be used as template for modifications; it is proposed to use a copy of this file as starting point for your own one.

The file format of cleat definition files is the versatile and user-friendly **cosin/io format**. For a detailed explanation of this format, see the respective cosin/io documentation. **FTire** data files should be written in **cosin/io format,** rather than TeimOrbit format. Moreover, the simulation scripts, needed later in the identification and validation phase, use this same format.

The most important advantage of **cosin/io** format, as compared to TeimOrbit format, is that it can be **parameterized**- using formula expressions rather than numerical values for all data items. And, in contrast to xml format or other widely used file formats, they are 'easy-to-read', and do not require a special editor.

The **cosin/io** format subdivides a file into so-called **data blocks** (similar like int the TeimOrbit format). In the case of the cleat definition file, these data blocks are named ..$obstacle_x (x = A, B, C, D, etc. - not case sensitive). Note the leading dollar-sign in the name, which denotes the beginning of a new data block. Each data block describes a single road surface geometry; let it be a cleat, a rotating drum, several cleats on a rotating drum, and so on. In the subsequent identification stage, reference to these surface geometries is established only by the suffix letter x = A, B, C, D, etc.

Below is an example of such a cleat definition data block. The exact meaning of the data items is documented in the cosin/road manual. However, most of the modifications to be made are obvious and are easy to adapt, such as:

- **cleat height**,
- **cleat length**,
- **cleat bevel edge length**,
- **cleat direction** (mounting direction angle relative to the drum axis),
- **drum diameter**,
- **type of drum** (tire running outside or inside drum),

etc. Note that **length unit is 'mm'**. Type of the bevel edge can be selected between 'straight' or 'rounded'. There are several more options available to define special cleat properties, see cosin/road.

```
$obstacle_A ! 20x25 mm, 0 deg
************************************************************
type drum; v vdrum/3.6; mu_factor 1.00; mu_factor_cleat 0.7

 cleat_height                 20   ! [mm]
 cleat_length                 25   ! [mm]
 cleat_bevel_edge_length       5   ! [mm]
 cleat_angle                   0   ! [deg]
 diameter                      2.0 ! [m] negative if outer drum
```

Specify as many data blocks as required to define all different cleats and road surfaces that had been used during the measurements. The assignment of these blocks to the different measurement files will be done later, during the 'check-in' process of the measurements.

Remember to select a **different** letter x ('obstacle_x') for each such data block. The letter must match the one in the TYDEX header of the measurement file in case it is specified. Alternatively the cleat geometry and drum dimensions can also be defined in the TYDEX data files. **FTire/fit** will automatically update the cleat definition file if the test surface does not match any previously defined obstacles in the cleat definition file.

It is possible to define a wide variety of other types of road surface geometries in a similar way, like twin or multiple cleats, pot holes, sine waves, etc.

4.1.4 Checking-in Measurement Files

Once the cleat definition file is specified, **FTire/fit** is ready to recognize ('check-in') experimental data.

To initiate the 'check-in' process :

- **selecting the directory** where the experimental data resides, then **marking the respective files in the measurement file list**, by either:
    1. left-clicking a single file, or
    2. dragging a **rectangle** over the files to be used, or
    3. selecting multiple files using the 'Shift' or 'Ctrl' buttons, and then

- **hitting the 'check-in'** button, in the top right corner of the launch-pad, with the left mouse button.

If you have specified one or more file(s), they will be marked blue in the file list, and **FTire/fit** will loop over all selected files. You may select **any arbitrary file**. **FTire/fit** will decide whether or not a file is relevant, and how it is to be interpreted. If this decision is not unambiguous, **FTire/fit** it will ask you whether and how to use it.

To skip the automatic measurement type recognition, the user may select a single file and hit the **'check-in'** button with the right mouse button. A new window will open, prompting the user to specify the type of the measurement. If possible, **FTire/fit** will use the specified measurement type and proceed with the check-in procedure.

Depending on the file type, left-clicking the (**loupe** button),located in the right centre of the launch-pad, will open an **appropriate application program** to view the contents of the file, whereas clicking the (**edit** button) will show the file contents in an ASCII editor, if applicable. Clicking the **?** (**check file** button), or alternatively 'right clicking' on a file, will run a measurement 'check-in' check (called 'What If' check) to determine the nature of the test and if all required data is specified in the file. Selecting the (**change settings** button) will open the **FTire/fit:Preferences** window. Experimental data may be given both as **image files** (in **bmp**, **png**, **tif**, **gif**, or **jpg** format), or as **ASCII text** files.

Image files are used to describe either:

- the tire's **cross section geometry**, or
- the tire's **tread pattern geometry**, or
- experimental data in terms of **scanned diagram images**, or
- measured **footprint** shapes.

Text files are used to save sensor information during static, steady-state, or dynamic time- or frequency domain measurements, given either in

- the preferred **standardized TYDEX format**, or
- in **general ASCII format**, to be interpreted by an assisting import tool.

In case an **image file** is recognized, based upon the file extension, **FTire/fit** will open a window similar to one of those shown in figure 4. The image will be displayed in this window.

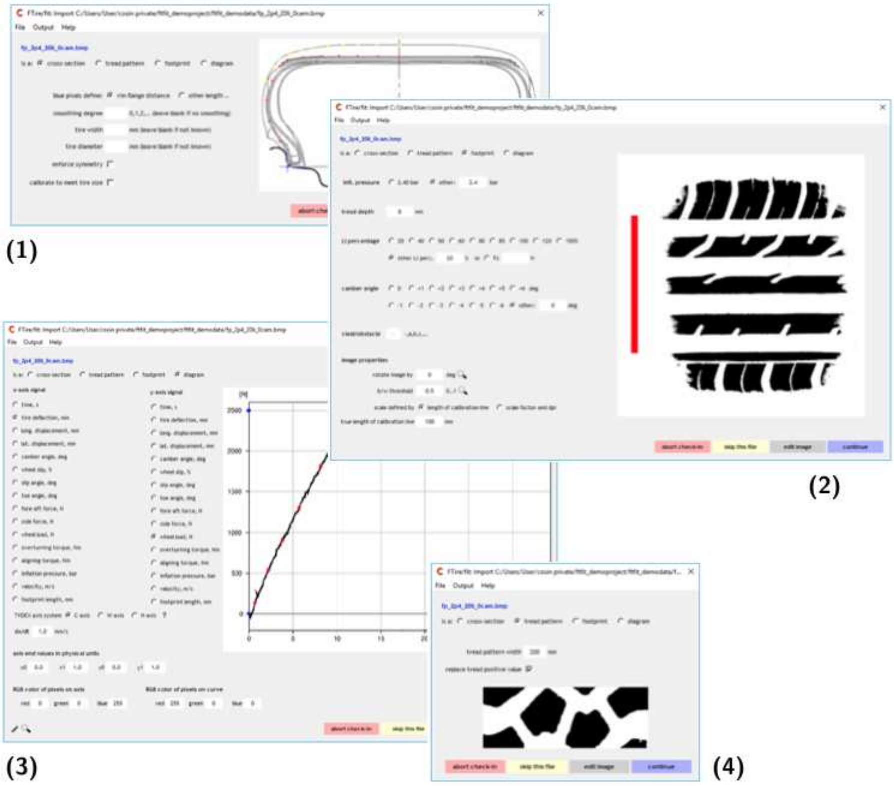

Figure 4: Types of images that can be checked in: (1) cross-section drawing, (2) footprint measurement, (3) scanned measured diagram, (4) tread pattern design

If **FTire/fit** cannot determine the image contents automatically, the user will be prompted to clarify the contents type in the top left corner of the window. A selection is to be made between:

1. a **cross-section** drawing,
2. a static **footprint** measurement,
3. a scanned measurement **diagram**, or
4. a **tread-pattern design** bitmap.

Depending on the user's selection, the windows appearance will change to either one of the types explained in chapter 4.1.5 to 4.1.9.

If, on the other hand, the file is an **ASCII file** in **TYDEX format**, **FTire/fit** tries to determine the kind of measurement contained in the file, by opening, reading, and analyzing the measurement channels.

Provided this recognition is unambiguous, and all necessary additional information is found in the file, **FTire/fit**:

- extracts the relevant measurement signals,
- based upon these signals, performs one among different least squares fits to approximate relevant tire properties like vertical/longitudinal/lateral stiffness, cornering /slip stiffness, sliding friction cases etc., and finally
- stores the resulting information in _best_so_far.tir and/or an auxiliary file, **FTire/fit**'s **'control file'**.

The information in the control file will later be used to parameterize simulation script templates, in order to identify and/or validate the measurement. Find more details about the checking-in of TYDEX files in chapter 4.1.10.

Finally, if the file contains a general, **non-TYDEX ASCII format**, an import-assistant window will open. This window queries the user for information about how to read, scale, or mathematically combine the measurement channels. By these operations, a TYDEX-equivalent file will be generated for further processing. Find more about this case in chapter 4.1.11.

The user can later add experimental data files the same way as explained above, at each sub-sequent stage of the identification process. Equally well, 'checked-in' test data may be discarded. The respective procedure will be explained later.

4.1.5 Preparing an Image File

In case of an image file, after having specified the contents, **FTire/fit** will change the window contents, showing how to **prepare** the image for final 'check-in' (figures 5, 6, 7, 10), and to query for any additional information that is needed.

In most cases, you will have to **'edit the image'** with MS-Paint or any other image-processing program of your choice. Alternatively, click **'skip this file'** if you decide not to make use of the image, or click **'abort check-in'** if you want to skip **all** remaining files that had been marked for check-in.

4.1.6 Checking-in a Tire Cross-Section Image

If an image file contains a **cross-section drawing**, it can be used to extract respective spline data **of the carcass line** and **tread/side-wall surface**. Figure 5 shows the preparation instructions for a **tire cross-section drawing**. These instructions stipulate where additional pixels, in **red**, **yellow**, or **blue** color, should be placed to define the **carcass**, **tread**, and **rim** geometry. When the **'edit image**' button is selected the image will be opened in the default image editing software. In addition, the image editing instructions will be displayed in the '**FTire/fit: Import**' window.

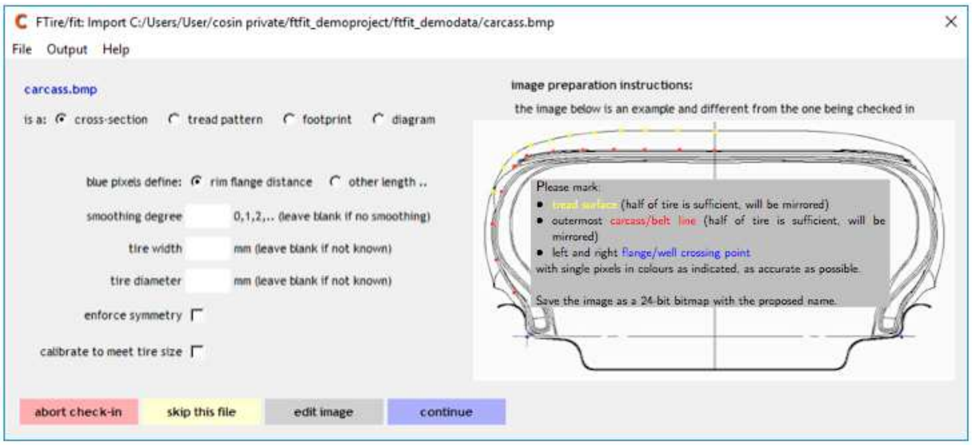

Figure 5: Instructions for digitizing a cross section bitmap

After all pixels are set, save the modified image as a 24-bit-bitmap file with the proposed name, **exit** the image processing program, and click **'continue'** in **FTire/fit**'s child window. Note that the proposed name for the modified image is different from the original name; the original file will **not** be overridden. **FTire/fit** will then:

- read the bitmap file,
- search for the pixels,

- scale the respective carcass/tread geometry data points according to the two pixels marking the rim size, and
- add/replace the resulting geometry spline data to/in the tire file _best_so_far.tir.

Checking-in of cross-sections does not require the specification of any additional data.

4.1.7 Checking-in a Tread Pattern Bitmap

A **tread pattern design bitmap** is expected to be cyclic repeatable in circumferential direction, and, for reasons of automatic scalability, there should not be any margin between bitmap and image border. Moreover, the circumferential axis of the tread pattern is expected to coincide with the vertical axis of the image.

The **tread pattern design image** preparation instructions are shown in **FTire/fit**'s child window (figure 6). The exact width of the tread pattern design should be entered in the the **tread pattern width** (in mm) field, and therefore doesn't need to coincide with the tread width as specified in the data file. When the **'edit image'** button is selected the image will be opened in the default image editing software. In addition, the image editing instructions will be displayed in the '**FTire/fit: Import**' window.

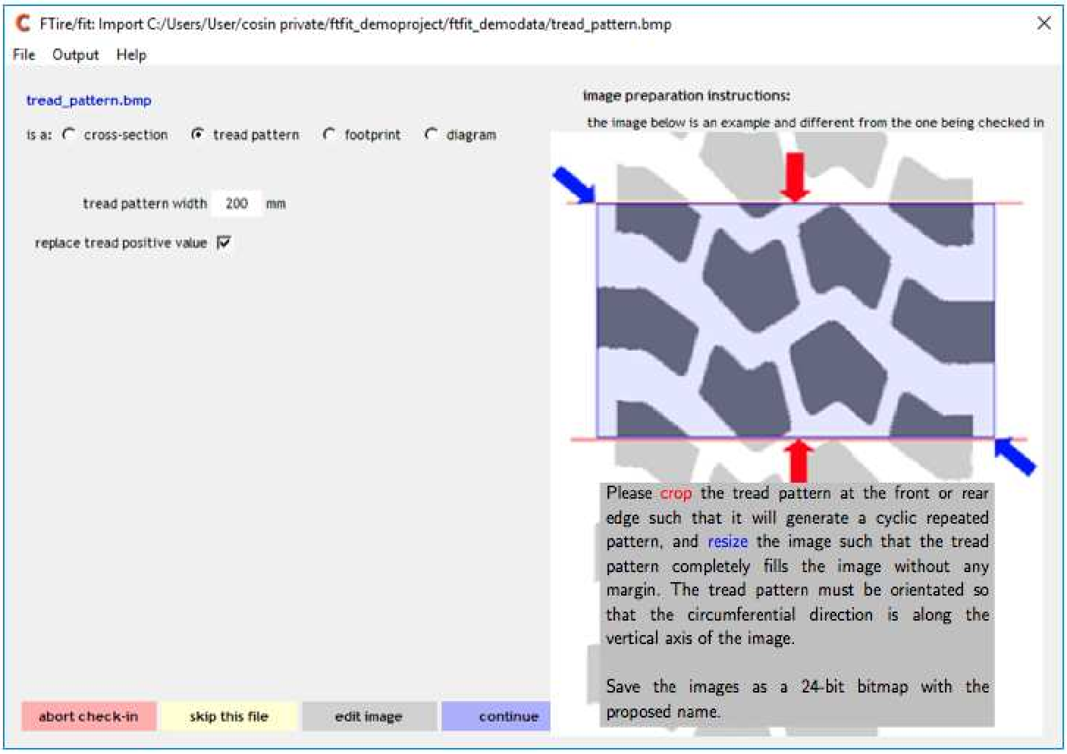

Figure 6: Instructions and data for preparing and checking-in a tread pattern bitmap

After the image is prepared, **'save'** the modification as a 24-bit-bitmap file with the proposed name, exit the image processing program, and click **'continue'** in **FTire/fit**'s child window. Again, note that the proposed name for the modified image is different from the original name; the original file will **not** be overridden. Next, **FTire/fit** will insert the bitmap file name and information about its extension and scaling in the tire file _best_so_far.tir.

4.1.8 Checking-in Measurements: Footprint Images

As opposed to the geometrical tire data discussed so far, **footprint images** are an important part of the **experimental data**, used later during the parameter identification. It is also recommended to use the same naming convention as discussed in the FTire Parameterization documentation to speed up the 'check-in' process. During 'check-in' information from the filename will be used to identify the measurement and to extract the required information about the operating conditions during the measurement.

FTire/fit needs to scale a footprint image exactly. This is achieved by scanning the image for a red line with a known length. Such a line is to be added to the bitmap (see figure 7). The length of the line is to be entered in the data entry field. In addition, the red line indicates the orientation of the bitmap. It is oriented perpendicular to the rolling direction.

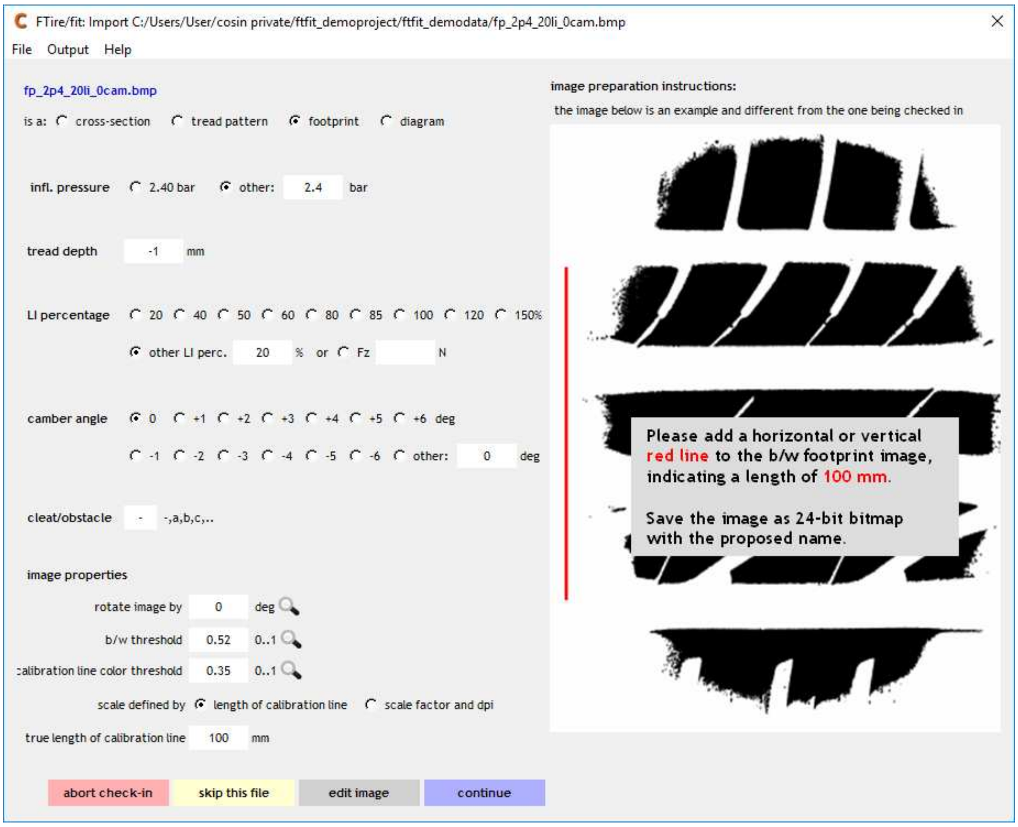

Figure 7: Instructions and data for preparing and checking-in a footprint bitmap

Moreover, the exact **operating conditions** of the footprint measurement have to be entered in the data entry field (see figure 7):

- **inflation pressure** can either be one of the pressure values p_1 or p_2, specified during creation of the data file, or a third independent value, specified in [bar]. Check the respective selection;
- **tread depth** of the tire, during the test in [mm]. Default value will be used if value is negative;
- **wheel load** can be defined either in percentage of the rated load ('LI load'), or directly in [N];
- **camber angle** in [deg];
- **cleat/obstacle** describing the road surface during the test. Leave blank if the road was flat;
- **image rotation angle** in [deg];
- **black/white threshold** value (between 0 and 1; dark images require a smaller value, light images a larger one. Default value is 0.52 and will work for most bitmaps),
- calibration line color threshold value (between 0 and 1; color threshold to recognize calibration line. Default value is 0.35 and will work for most bitmaps),

- the **length of the calibration line** ('scale length') in [mm]. Alternative, if no such calibration line was added to the bitmap and 'scale factor' is checked as scale definition type, the image **scaling factor** in dots per inch [dpi],
- min. and max. ground pressure scaling values in [MPa] (only requested when the image contains contact pressure distribution information).

When the **'edit image**' button is selected the image will be opened in the default image editing software. In addition, the image editing instructions will be displayed in the '**FTire/fit: Import**' window. After the image is prepared, **save** the modification as a 24-bit-bitmap file with the proposed name, **exit** the image processing software, **specify the additional data** in **FTire/fit**'s child window, and click **'continue'** (figure 7). Again, note that the proposed name for the modified image is different from the original name; the original file will **not** be overridden.

FTire/fit saves all information necessary to compare the measured footprint with a simulated one. The data is written to an internal **control file** (named **contr.ftf**). The actual content of this file, as far as footprint measurements are concerned, is displayed in the **'footprints'** test category (figure 8). In this window, for each new footprint being checked in, another row of check-boxes, buttons, and entry fields is created. These widgets provide direct access to the respective measurement.

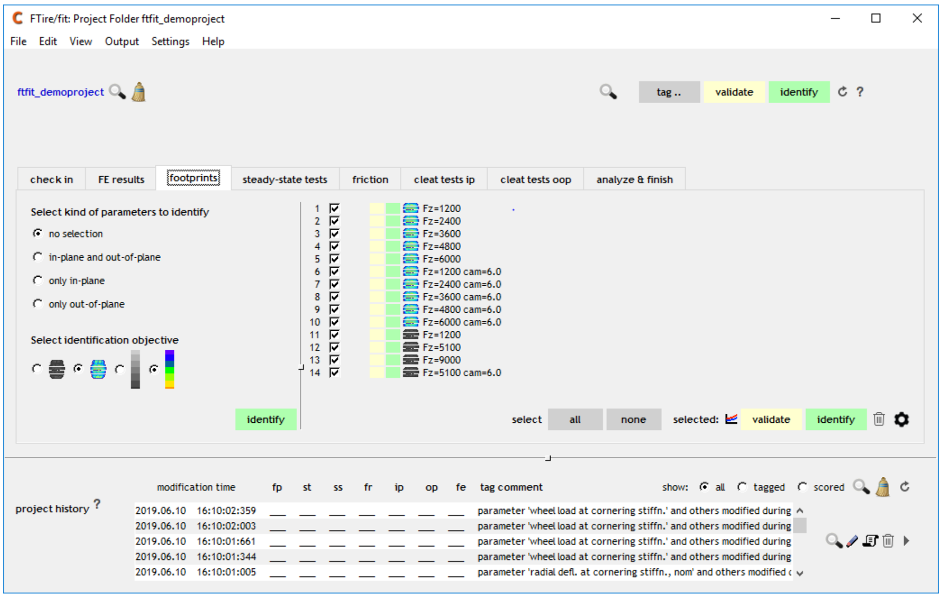

Figure 8: Measurement window for footprint-type measurements

The rows, and thus the measured footprints, are sorted with respect to the operating conditions, so the 'check-in sequence is irrelevant.

Selecting a measurement, by 'left-clicking' on the measurement entry (figure 8), will open a 'Measurement Details' child window (figure 9). This window allows the user to edit the 'check-in' details, as well as to view, edit and delete the 'checked-in' measurement from the project if necessary. The following buttons are available:

- **(loupe** button), to open the footprint, using the OS's standard image processing program;
- (**info** button), to display the **'check-in details**';
- (**interactive ID** button), to run an interactive comparison. In this session, the animation window is equipped with slider bars, allowing the user to modify certain predefined parameters during a running simulation, and to study the effect on better or worse coincidence between measurement and simulation.

Because these parameters are modified on-line, it would take too much time to refresh the pre-processing. This is why the slider bars are only available for certain internal parameters, the change of which would not affect the pre-processing results of other internal parameters too much;

- entry field to define the number of cross section modifiers to be available during a interactive comparison;
- (**trash can** button), to delete the measurement from the project (selected measurement will be 'checked-out'). Of course, the original measurement file will never be deleted.

Using the entry fields on the bottom of the 'Measurement Details' window you can manually modify, if required, several additional parameters, allowing to adjust certain operating conditions or identification settings that were set during the 'check-in' phase:

- **'cleat/obstacle'**: the obstacle or cleat type, denoted by a single letter (see chapter 4.1.11);
  - (**edit** button), to open obstacle in a editor. Left-click: open obstacle in cosin/io data-block editor, Right-click: open obstacle in cosin/tools for roads;
  - (**loupe** button), to visualize the obstacle,
- **'wheel load'**: the wheel load in [N];
- **'camber angle'**, the tire camber angle in [deg];
- **'inflation pressure'**, the actual inflation pressure during measurement in [bar]; the value '-1' denotes no change compared to the value specified in the data file;
- **'tread depth'**, the actual tread depth during measurement in [mm]; value '-1' denotes no change compared to the value specified in the data file for a new tire;
- **'image scaling factor'**, the scaling factor of a footprint bitmap file (real length / length in image);
- **'image rotation angle'**, the footprint rotation angle in a bitmap, in [deg];
- **'b/w threshold'**, the bitmap black/white threshold value;
- **'min. ground pressure'**, color palette minimum ground pressure scaling value, in [MPa];
- **'max. ground pressure'**, color palette maximum ground pressure scaling value, in [MPa]

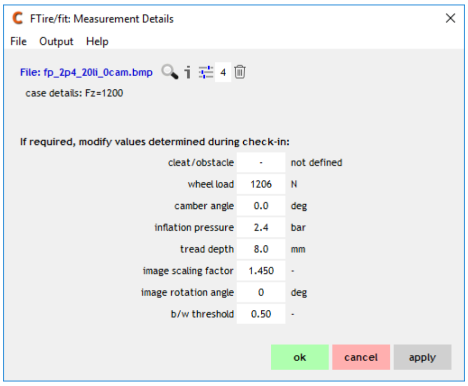

Figure 9: Measurement details window for footprint-type measurements

The remaining functions of the 'footprints' measurement window are explained in chapter 4.2.

4.1.9 Checking-in Measurements: Scanned Diagram Images

The fourth type of image files, scanned diagrams, represent another kind of **experimental data**, used for identification. **FTire/fit** provides a digitizing function to generate an intermediate ASCII file, which is equivalent to, and treated in exactly the same way as, the standardized TYDEX files which are discussed in 4.1.10. Only limited information about the testing conditions can be acquired from the image. It is thus important to follow the standard measurement procedure as discussed in FTire Parameterization documentation and to follow the same naming convention as discussed in the document. **FTire/fit** will try to extract the required information from the file name.

During digitizing, **FTire/fit** searches and reads **x/y data pairs** out of a diagram curve in the image. When preparing the image, with an image processing program, the respective points on the curve are to be marked by **colored pixels** (default color is red). In order to exactly calibrate and scale the image, **FTire/fit** needs to know origin position, as well as axis end points with their respective physical value. These points also need to be marked with **colored pixels** (default color is blue). The exact procedure is explained in the preparation instruction (figure 10), which appear when the '**edit image**' button is selected.

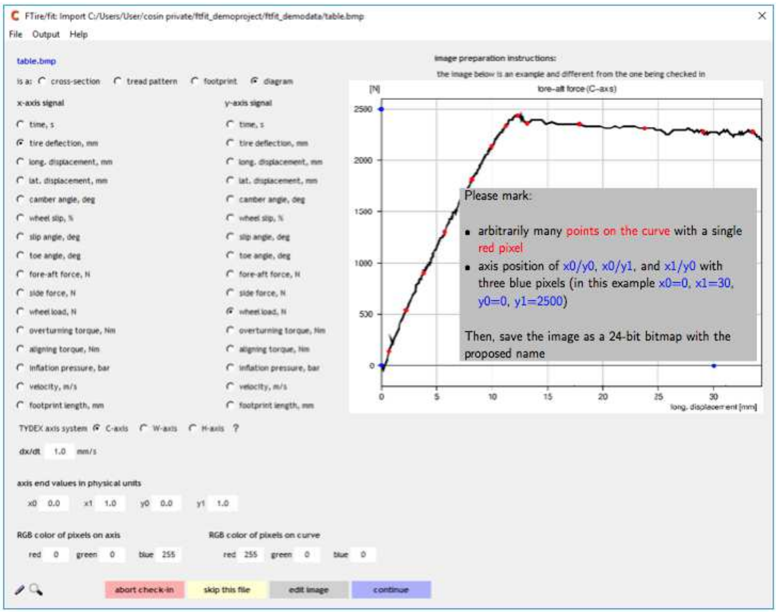

Figure 10: Data for checking-in a scanned diagram, and plot/edit buttons

The **values of the axis end points in physical units** (x_0 and x_1 on the x-axis, y_0 and y_1 on the y-axis) are to be entered on the bottom of the data entry field (figure 10). The units of these values depend on the selection of the **x- and y-axis variables**, which are to be checked in the two columns above. The rate of change of the **x-axis variable**, with respect to time, also needs to be defined.

If specifying a tire force or moment, the respective **TYDEX coordinate system** (C-, W-, or H-axis system) is to be selected in another row of 'radio-buttons' near the bottom of the data entry field. Please refer to the TYDEX documentation for an exact definition of these axis systems.

After you have prepared the image, save the modification as 24-bit-bitmap file with the proposed name, **exit** the image processing program, **specify the additional data** in **FTire/fit**'s child window, and click **'continue'** (figure 10). As said before, the proposed name for the modified image is different from the original name; the original file will **not** be overridden.

FTire/fit will now perform the final digitizing step and **create the intermediate mtl file**.The content of the intermediate file, and thus the digitizing result, will then be 'checked-in'. In this step of the 'check-in' procedure, the intermediate file will be **treated exactly the same way as TYDEX files** are treated. This procedure, together with the necessary user interaction, is explained in 4.1.10.

4.1.10 Checking-in Measurements: TYDEX-Files

If an **ASCII text file** in TYDEX format is checked in, or if the intermediate file created during check-in of scanned diagrams or general other ASCII files is available, **FTire/fit** will open this file and search for the existence and the statistical properties of certain measurement signals. This task is especially easy both with TYDEX files and with files in **FTire/fit**'s internally used 'mtl format', because all relevant signals have well-defined and well-known names.

In addition to the extraction of validation and identification test case information, and depending on the type of measurement, **FTire/fit** will occasionally **extract tire data** from the measurement and insert these data into the tire data file _best_so_far.tir. This can be done either in terms of **actual values**, like the data describing the radial stiffness characteristic on a flat surface, or in terms of merely **nominal data**, that can later be activated for use.

As a result of the analysis, **FTire/fit categorizes** all measurements into exactly one of the following classes:

1. carcass deformation characteristics from finite element analyses,
2. steady-state belt extensibility characteristics at different inflation pressures,
3. dynamic rolling circumference characteristics,
4. radial stiffness characteristic at zero rolling speed on flat surface,
5. radial stiffness characteristic at zero rolling speed on transversally oriented cleat,
6. radial stiffness characteristic at zero rolling speed on longitudinally oriented cleat,
7. radial stiffness on drum at non-zero rolling speed,
8. longitudinal stiffness / sliding characteristic of blocked wheel near zero speed on flat surface,
9. lateral stiffness /sliding characteristic of blocked wheel on flat surface,
10. lateral stiffness / sliding characteristic at zero rolling speed on transversally oriented cleat,
11. torsional stiffness/hysteresis characteristic of blocked wheel when turning about vertical axis,
12. longitudinal force vs. longitudinal slip characteristic at higher rolling speed,
13. side force vs. slip angle characteristic at higher rolling speed,
14. side force vs. camber angle characteristic at higher rolling speed,
15. cleat test under symmetric conditions ('in-plane cleat test'),
16. cleat test under un-symmetric conditions ('out-of-plane cleat test'),
17. test not used or type not recognized.

Moreover, from the data file, **FTire/fit** tries to **extract values** for:

1. inflation pressure,
2. steady-state wheel load (if kept constant),
3. steady-state longitudinal slip (if kept constant),

4. steady-state slip angle (if kept constant),
5. steady-state camber angle (if kept constant),
6. drum speed,
7. drum diameter, and
8. cleat type.

If at least one of these values is needed to unambiguously describe the measurement conditions, but is not found in the data file, **a small window** will be displayed (figure 11) requesting the missing information. This window shows the name of the respective measurement file, and has check-boxes and/or entry fields to **specify or enter all missing values** (the actual appearance of the window varies with the kind of data missing). After having completed this, click either:

- **'continue'** to complete checking-in of the file, or
- **'skip this file'** if you decide not to make use of it, or
- **'cancel check in'** if you want to quit the whole check-in process.

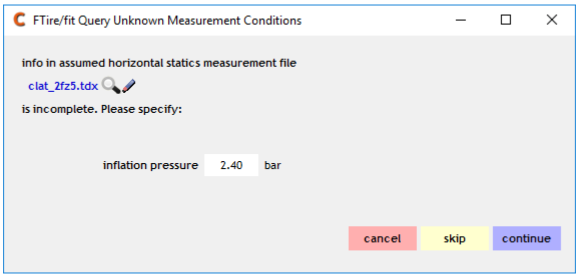

Figure 11: Child window prompting for missing measurement conditions

When the measurement conditions can unambiguously be determined, **FTire/fit** will add one or more new validation/identification cases to the control file.

These cases are grouped into **six categories**. Such a category can be selected for display from the respective measurement tab of the top section of the launch-pad (figure 2). The measurement categories are:

- **finite element** results of carcass deformation simulations (class (1) in the list above)
- **footprint shape** measurements (already discussed and shown in chapter 4.1.8 and figure 8)
- measurements in **static or steady-state** conditions (classes (2) to (14) in the list above)
- measurement conditions where **friction** plays an important role, to determine **friction coefficients**,
- dynamic cleat tests with purely symmetric conditions (**'in-plane cleat tests'**, class (15) above)
- dynamic cleat tests with un-symmetric conditions (**'out-of-plane cleat tests'**, class (16) above).

The second and third category will be filled simultaneously, on basis of the same measurement files. As mentioned, all relevant measurement information will be recognized and extracted automatically from the files during the 'check-in' process.

4.1.11 Checking-in Measurements: General ASCII-Files

The last type of files that can be 'checked-in' are **general ASCII files**, containing equidistant time domain measurements, arranged in several columns, in a format other than TYDEX. **FTire/fit** provides an import tool, helping to create TYDEX-equivalent files out of such files.

If during check-in a non-TYDEX ASCII file is recognized, an import-assisting window pops up (figure 12).

This window allows to specifying several computation formulae (figure 12). These formulae will be used to calibrate or combine one or more channels in the ASCII file, resulting in well-defined measurement channels, using units and coordinate systems as defined in the TYDEX standard.

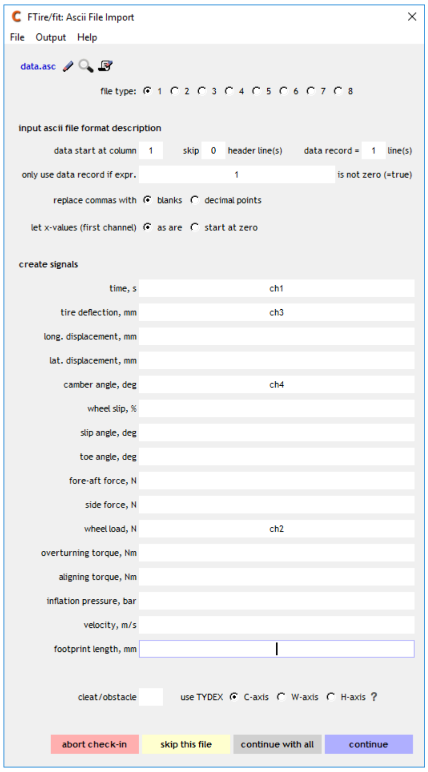

Figure 12: ASCII file import window

Nearly every meaningful arithmetic expression will be recognized, including all important mathematical functions, arbitrarily nested parentheses, all arithmetic, logical, and comparison operators, predefined unit conversion factors, like:

- **mph2ms** (mph to m/s),
- **lbf2N** (pound-force to N),
- **in2m** (inch to meter),
- **ft2m** (foot to meter),
- **r2d** (radians to degree),
- **psi2bar** (psi to bar),

and much more.

Measurement channels in the given ASCII file are made available as variables **chx** (x=1,2,3,..,99), see figure 12. The record number is available as variable **ch0**.

For a detailed documentation of arithmetic expressions, see the cosin/io documentation chapter.

Channels for which computation formulae are not specified (that is, for which the formula entry field is left blank) will not be created.

Forces and moments can be computed in either one of the three coordinate systems (**C-, W-, and H-axis**), defined in the TYDEX standard. Similarly as in the scanned diagram import window (see chapter 4.1.9), this coordinate system can be selected in a row of radio-buttons, located below the formulae entry fields (figure 12).

In another set of entry fields (figure 12), the ASCII file format can be specified in greater detail, for assisting **FTire/fit**'s reading and interpretation routine. This includes:

- the **specification of comma interpretation** either as field separators or as decimal points (**'replace commas with blanks'** vs. **'replace commas with decimal points'**);
- disregarding a certain number of leading columns (**'data start at column xx'**);
- skipping a certain number of comment lines in the header (**'skip xx header lines'**);
- merging a certain number of lines into one logical record (**'data record consists of xx lines'**);
- only taking into account a data record if its contents fulfills a certain condition, expressed by an arithmetic expression (**'only use data record if following expression is true:yyyy'**). As an ex-ample, this field can be used to extract a single measurement out of a file that contains more than one measurements in a sequence: **'only use data record if following expression is true:ch0>=3001& ch0<=4000'** would select lines 3001 to 4000;
- shifting the values of the independent variable (the first channel in the output file, typically time) such that it starts at value 0 (**'x-values start at zero'** checkbox).

In case of having to check-in **several different** general ASCII file types, up to 8 different sets of settings can be stored and retrieved with the radio-buttons on the top of window (figure 12).

This is a very simple way to make use of data given as part of one of these popular file types: mark the data columns you want to check in, copy them to the clip-board (hit Ctrl-C), and then continue in **FTire/fit** as if you were checking in an ASCII file.

After you have specified the relevant information, click **'continue'**. **FTire/fit** will now **create the intermediate file**. Similarly as with the scanned image files, the content of the intermediate file can be checked in two different ways: it can be **plotted** with the interactive plot program **cosin/ip**, or it can be **edited** with the ASCII editor. In addition, the **original** ASCII file can be edited after clicking the respective button on bottom of the window.

In the same way as ASCII files, the contents of the **clipboard** is treated in case one of the following file types is being checked in:

- a **pdf-file**
- a **Word document** (doc file),
- an **Excel spreadsheet** (xls file), or
- a **PowerPoint presentation** (ppt file).

In the next step, the intermediate file will **be treated exactly the same way as original TYDEX files** are treated. This procedure, together with the necessary user interaction, is explained in 4.1.10.

4.1.12 View/Modify Measurements after Check-in

Selecting a measurement in the measurement section, by 'left-clicking' on the measurement entry, will open a 'Measurement Details' child window (see figure 9). This window allows the user to edit the 'check-in' details, as well as to view, edit and delete the 'checked-in' measurement from the project if necessary. The following buttons might be available (depending on measurement):

- **(loupe** button), to open/view the measurement. Footprint are opened in the OS's standard image processing program, all other measurements will be opened in **cosin/ip**.
- (**info** button), to display the **'check-in details**';
- (**edit** button), to open the measurement in a ascii editor;
- (**script** button), to open a GUI to edit the simulation script;
- (**edit script** button), to open simulation script in a text editor;
- (**save as reference** button), to save the last simulation as a reference;
- (**clean** button), to remove the saved reference file;
- (**interactive ID** button), to run an interactive comparison. In this session, the animation window is equipped with slider bars, allowing the user to modify certain predefined parameters during a running simulation, and to study the effect on better or worse coincidence between measurement and simulation. Because these parameters are modified on-line, it would take too much time to refresh the pre-processing. This is why the slider bars are only available for certain internal parameters, the change of which would not affect the pre-processing results of other internal parameters too much;
- (**trash can** button), to delete the measurement from the project (selected measurement will be 'checked-out'). Of course, the original measurement file will never be deleted.

Several additional entry fields, allowing to adjust certain operating conditions or identification settings (if relevant), a selection is listed below:

- '**XSH**': the measurement shift along the x-axis;
- **'YSH**': the measurement shift along the y-axis;
- **'XSC**': the measurement scale factor of the the x-axis;
- **'YSC**': the measurement scale factor of the y-axis;
- **'type**': the type of the independent variable;
- '**IAXIS**': the `TYDEX` axis system;
- **'cleat/obstacle'**: the obstacle or cleat type, denoted by a single letter (see chapter 4.1.11);
  - (**edit** button), to open obstacle in a editor. Left-click: open obstacle in cosin/io data-block editor, Right-click: open obstacle in cosin/tools for roads;
  - **(loupe** button), to visualize the obstacle,
- **'V'**: the rolling speed in [km/h];
- **'FZ'**: the wheel load in [N];
- **'KA'**, the tire longitudinal slip in [%];
- **'AL'**, the tire slip angle in [deg];
- **'CAM'**, the tire camber angle in [deg];
- **'DD'**, the drum diameter (if not specified in the cleat definition file) in [m], positive if drum is an inner drum and negative if it is an outer drum;
- **'P'**, the actual inflation pressure during measurement in [bar]; the value '-1' denotes no change compared to the value specified in the data file;

- **‘TD’**, the actual tread depth during measurement in [mm]; value ‘-1’ denotes no change compared to the value specified in the data file for a new tire;
- ’MU’: the friction factor of the test surface;
- **‘search’**, the measurement time in [s], starting at which the first cleat contact is to be searched for in the measurement file;
- **‘trig’**, the measurement time of first cleat contact (if automatic detection fails), in [s]. If the ‘trig’ value is negative, the automatically detected value is to be used anyway;
- **‘tol’**, the percentile change in the steady-state wheel load which indicates a cleat contact (the exact time of first contact will be estimated by using this value, together with the rate of change of wheel load with respect to time);
- **‘w.time’**, a 5-digit integer number, defining the weights of time domain signals used to build the least squares expression for cleat test identification. The weights are defined in the sequence F_x, F_y, F_z, M_x, M_z. The value 90900, say, will set the same weight on F_x and F_z, but will neglect F_y, M_x, M_z;
- **‘w.freq’**, weights similar as ‘w.time’, but to be used in frequency domain;
- **‘fmax’**, the maximum frequency in [Hz] to be taken into account during frequency domain least squares fit. As a side effect, ‘fmax’ will determine the time span of the cleat test simulation runs. The default and recommended value is 200 Hz.

4.2 Identification / Validation Tasks

After having checked in some or all data files, the identification and/or validation phase can be entered. Clearly, these two steps **don’t need to be carried out in a strictly sequential manner**. As soon as measurements are displayed in one of the measurement categories, they can be used to improve the data file _best_so_far.tir.

Figure 13 shows the general layout of the launch-pad if a measurement category tab is selected. Depending on the category selected, the launch-pad section will have a specific layout; compare figures 8 and 14 to 18. **Section a,** of figure 13, is indicating the ’**parameter selection section**’. The category specific section layout will be discussed later.

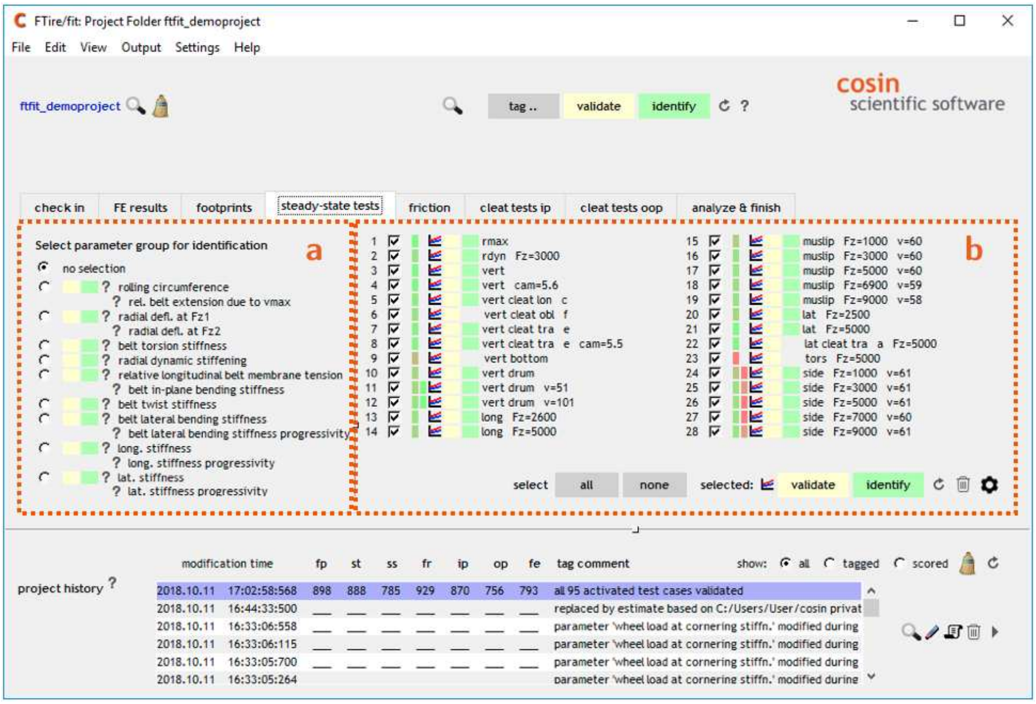

Figure 13: FTire/fit launch-pad

All measurements are organized in numbered rows in the '**measurement section**', on the right-hand side (**section b** of figure 13). If a large number of measurements is 'checked-in' for a specific test category, the measurements might be organized in multiple columns (see figure 13). The following widgets may appear in each measurement row:

- **'measurement number'**, to identify the measurement within the measurement category;
- a **'use'** check-box, to select/deselect the measurement case during scripted validation or identification;
- a single/multiple **'validation error'** field/s, the color of the field quantifies the error between the measurements and the corresponding simulation results. The number of visible fields depends on the number of signals that are being compared. The color ranges from red (poor agreement) to green (good agreement). Hovering the cursor over the field will display the percentage error. If no validation results are available the field/s will not be displayed;
- a (**'browse validation results'**) button, to quickly show the most recent comparison results between the measurement and its corresponding simulation. If no such image is available, the button will not be displayed;
- a yellow **'validate'** button, to compute and display a new comparison, using the most recent status of the data file _best_so_far.tir;
- a green **'identify'** button (if applicable), to automatically identify relevant tire parameter/s from the single measurement, and updating the status of the data file _best_so_far.tir;
- internal **'case details'**, describing the measurement for easy identification. FTire/fit will duplicate or create a new file for each 'checked-in' measurement, having a well-defined file name and format, no matter whether or not the original file name could have been used directly or not. These internal file names all have an underscore as prefix.

In the bottom right corner of the '**measurement section**' (**section b** of figure 13) are a number of buttons that control all measurements within the current category. These buttons in this section may consist of:

- a all (gray select **'all'**) button, to select all measurements within the measurement category;
- a none (gray select **'none'**) button, to deselect all measurements within the measurement category;
- a (**'browse validation results'**) button, to browse all available validation results;
- a validate (yellow **'validate'** selected) button, to compute and display all all selected measurements and their corresponding simulation results in form of a graph;
- a identify (green **'identify'** selected) button, to automatically identify relevant tire parameters from the selected measurements, and updating the status of the data file _best_so_far.tir;
- a (**'refresh'**) button, to reload the simulation scripts. User-specific changes will be overwritten.
- a (**'trash can'**) button, to delete all selected measurements, of the current category, from the current project (selected measurement swill be 'checked-out'). Only **FTire/fit** internally used files will be deleted, the original measurement files will never be deleted by **FTire/fit**;
- a (change **'settings'**) button, to open the corresponding **FTire/fit Preferences** window. The settings can us used to control the parameter identification process.Figure 8 and 14 to 18 show the respective window layout of all of the measurement categories.

4.2.1 Measurement Windows

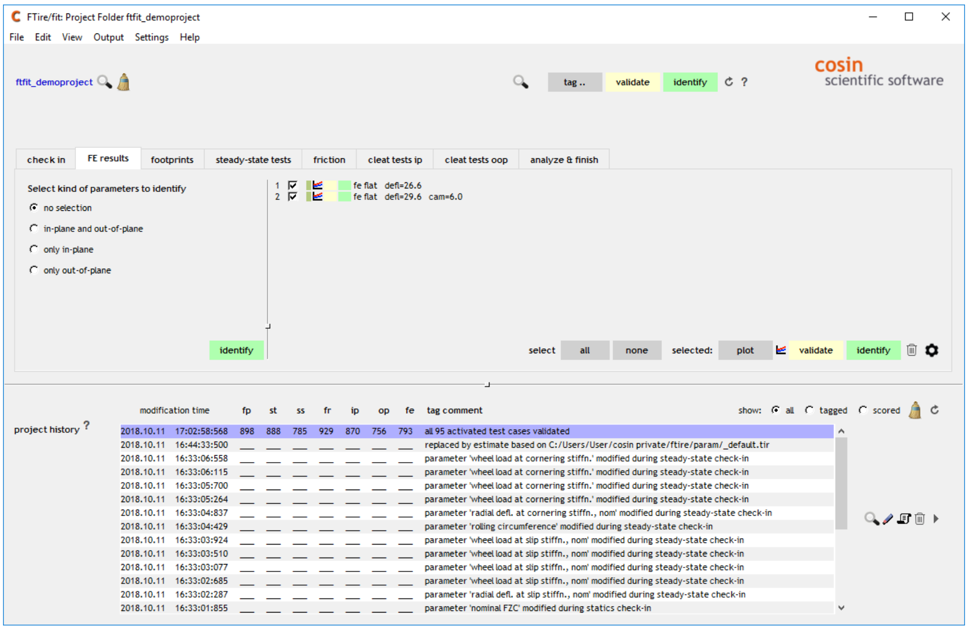

Figure 14: Measurement window for category 'FE results'

The layout of the '**footprints**' and '**FE results**' measurement categories are shown in figure 8 and figure 14. The 'parameter selection section', on the left hand side of the launch pad, allows the user to specify a kind of **parameter group** that should be identified. The corresponding measurements, that are predominantly influenced by the parameter selection, will be highlighted in the right-hand measurement section. The following widgets may appear in this section:

- a **'select'** radio-button, to select/deselect the parameter group/s for individual identification;
- **'parameter group description'**, describing the parameter group;

- a

(green **'identify'** button) - at the bottom right corner, to automatically identify relevant tire parameters, of the selected **parameters group**, from the selected measurements, and updating the status of the data file _best_so_far.tir.

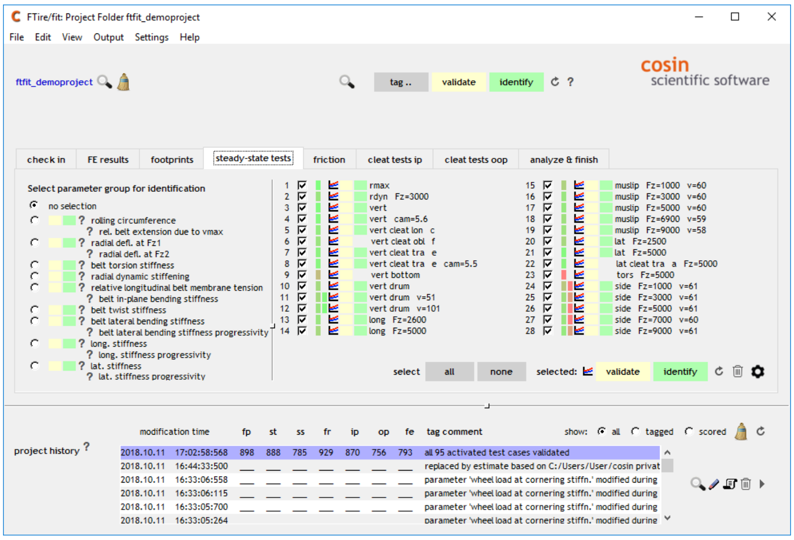

Figure 15: Measurement window for category 'static and steady-state cases'

For the steady-state tests (figure 15), the left-hand side (**section a** of figure 13) lists all the parameters, or groups of parameters, that can be identified from the currently available measurements. Again, all available measurements, associated to the current category, are listed in the measurement section (**section b** of figure 13) on the right-hand side . A parameter, or set of parameters, can be selected by activating the radio-buttons on the left. The corresponding measurements, that are predominantly influenced by the parameter selection, will be highlighted in the right-hand measurement section (**section b** of figure 13). The following widgets may appear in each parameter/parameter group row:

- **'select'** radio-button, to select the parameter or parameter group;
- a yellow **'validate'** button, to compute and display a new comparison of all tests that are sensitive to the selected parameter or parameter group, using the most recent status of the data file _best_so_far.tir;
- a green **'identify'** button, to automatically identify the respective selected single/multiple parameter/s from the relevant measurement/s so that the measurement/s is optimally replicated by the simulation. The identified parameter/s will the be saved in the data file _best_so_far.tir;
- a **?** (**'help'**) button, to get more information about a parameter. **FTire/fit** will open the corresponding help file..
- **'parameter details'**, describing the parameter or parameters.

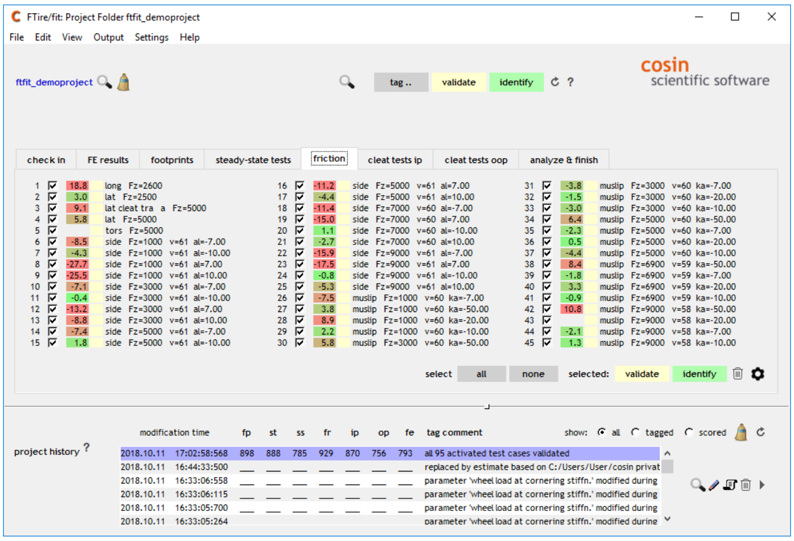

Figure 16: Measurement window for category ‘friction’

The **'friction'** window displays only the extracted friction cases. These cases were extracted during the 'check-in' process of some of the **'steady state tests**'. In contrast to all other measurement categories, the error, between the measured and simulated force and moment signals of the friction cases, is shown within the **'validation error'** field. The color of the field again corresponds to the level of correlation between the measurement and simulation.

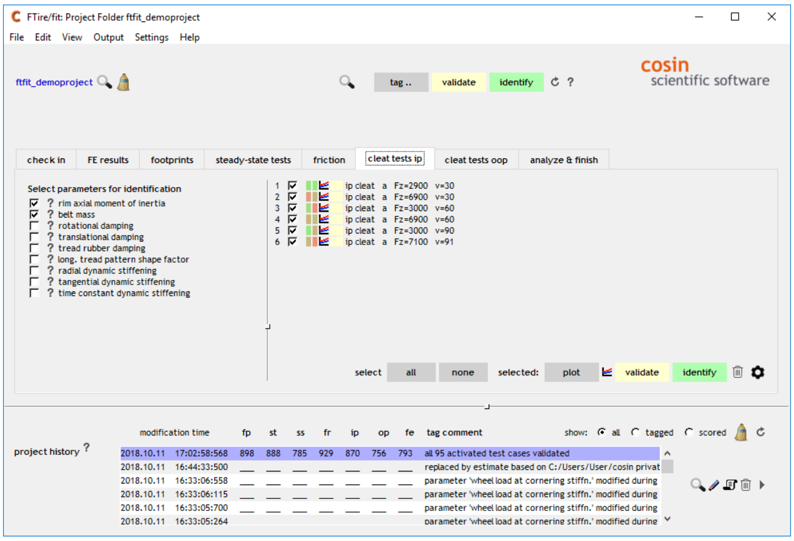

Figure 17: Measurement and parameter windows for category ‘in-plane cleat tests’

For the dynamic cleat tests categories (figure 17 and figure 18), the left-hand side of the launch-pad allows the user to specify a set of parameters, maximum of two, that should be identified individually. Parameters with high relevance for dynamic cleat tests, but small influence on other measurement types, can be activated / deactivated for optimization. Moreover, they can be equipped with minimum and maximum bounds(needs to be activated in the project **Preferences**). The following widgets may appear in each parameter row:

- a **‘use’** check-box, to select/deselect the parameter for individual identification;
- a **?** (**'help'** button), to get more information about a parameter. **FTire/fit** will open the corresponding help file that defines the parameter..
- **'parameter details'**, describing the parameter;

- **'identification bounds'**, to set the lower and upper variation range of the parameter during the identification process. Individual identification bounds need to be activated in the project **Preferences.**

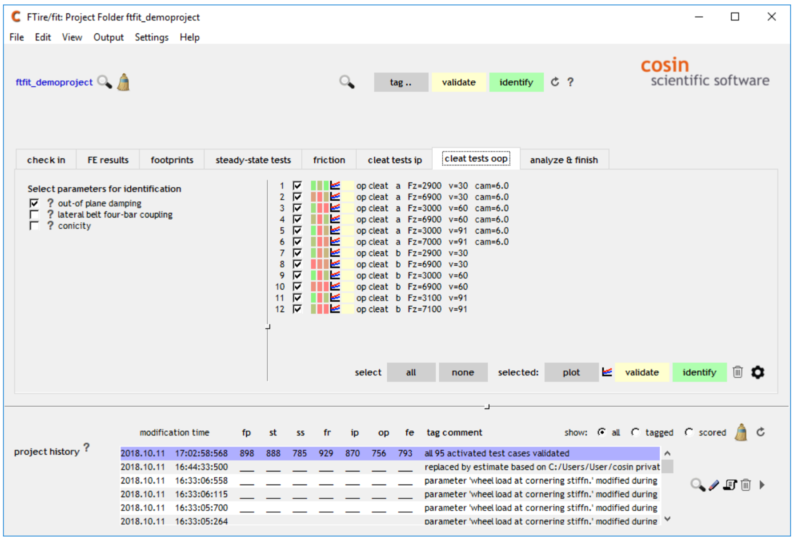

Figure 18: Measurement and parameter windows for category 'out-of-plane cleat tests'

4.2.2 Performing Validation/Identification Tasks

An identification or validation task can be launched in the following ways:

- **single identification**, using **single** measurements, either **'manually'** or **'semi-automatically'**, by using the **green identify** buttons in a measurement window . Note that not all measurement types are equipped with such a **green identify** button, but only those for which an isolated identification of one or at most two parameter(s) would be practical;
- **single validation** (comparison) of **single** measurements, with the **yellow validate** buttons in a measurement window;
- **group identification** of **all relevant parameters**, using **all activated** measurements of a **single category**. This is initiated by selecting the identify button in the bottom right corner of the '**measurement section**' (see figure 13);
- **group validation**, using **all activated** measurements of a **single category**. This is initiated by selecting the validate button in the bottom right corner of the '**measurement section'** (see figure 13);
- **parameter group identification** of **a parameter or group of parameters**, using **all activated and relevant** measurements of a **measurement category**. This is initiated by selecting the identify button in the bottom right corner of the '**parameter selection section'** (see figure 13) or by selecting the **green validation button** next to the parameter group;
- **parameter group validation** of **aa parameter or group parameters**, using **all activated and relevant** measurements of a **measurement category**. This is initiated by selecting the **yellow validation button** next to the parameter group;
- **fully automatic identification, using all activated measurements of all categories**. This process will run through a predefined sequence of all identification sub-tasks. It is initiated by **selecting** the identify button in the top section of the main menu (**section b** of figure 2)**.** Fully automatic identification includes the complete identification and validation process using all 'checked-in' test data, an automatic optimization

of numerical settings, the computation of additional tire model properties, finishing tasks, and the generation of the result report;

- **complete validation, using all activated measurements of all categories**. This process will perform or refresh all validation simulations, using the most recent data file. It is initiated by selecting the validate button in the top section of the main menu (**section b** of figure 2). A report will also be be generated at the end of this task.

Certainly, the **complete and fully automatic identification** will not always result in satisfactory results. As with most other complex computation tasks, the fitting procedure might work better if performed interactively, on basis of certain experiences with previous identification projects, together with a sound knowledge of **FTire**'s model structure and parameter sensitivities.

The **semi-automatic identification**, mentioned above, is invoked by selecting the **green 'identification' button** in the measurement row. Depending on the kind of measurement at hand, and if available, clicking such a button will fine-tune at most two predefined parameters at a time. The selection of these parameters is 'hard-coded' in **FTire/fit**, by choosing the ones with highest correlation to the respective kind of measurement.

The **green 'identification' buttons** will only be effective if the operating conditions during the measurement (inflation pressure and tread depth) are the same as the ones in the tire data file. The parameters that can be automatically adjusted this way are:

- **in-plane bending stiffness** in case of a measured footprint bitmap, to match the averaged footprint length, for first and second inflation pressure;
- **lateral bending stiffness** in case of a measured footprint bitmap, to match the averaged footprint width, for first and second inflation pressure;
- **wheel loads at first and second deflection value** in case of a vertical deflection characteristic on even surface, independently for first and second inflation pressure;
- **relative longitudinal belt membrane tension** in case of a vertical deflection characteristic on transversal cleat for first inflation pressure;
- **in-plane bending stiffness** in case of a vertical deflection characteristic on transversal cleat for second inflation pressure (note that the identification result will override the one using a footprint bitmap. The user will have to find a respective compromise between the two possibly contradicting results);
- **lateral bending stiffness** in case of a vertical deflection characteristic on longitudinal cleat, independently for first and second inflation pressure (note that the identification result will override the one using a footprint bitmap. The user will have to find a respective compromise between the two possibly contradicting results);
- **longitudinal tire stiffness** in case of a longitudinal deflection characteristic on flat surface, in-dependently for first and second inflation pressure;
- **lateral tire stiffness** in case of a lateral deflection characteristic on flat surface, independently for first and second inflation pressure;
- **torsional tire stiffness** in case of a torsional deflection characteristic on flat surface, independently for first and second inflation pressure;
- **friction characteristics**, in the case of specific steady-state measurements;
- **damping parameters** of the tire structure and tread elements, in the case of dynamic cleat tests;
- etc.

The **'manual'** identification requires certain knowledge concerning the sensitivity between important model data and model properties. In this identification approach, it is most convenient to make use of **FTire**'s data file editor (cosin/tools for tires). This editor, conveniently accessible with the (**edit/analyze current status data file**) button in the top line of the main window, does not only simplify the finding and changing of parameter values in the tire data file. It also indicates what parameters would actually be used with the current setting, and provides several graphical data visualizations and short-cuts to other **FTire** tools.

4.2.3 Browse Validation Results

When the (**'browse validation results'**) button is selected a **FTire/fit: Validation window** (figure 19) will open. The window shows the measurement and corresponding **FTire** simulation results. The following buttons are available in the bottom left/right corner of the validation window:

- a (gray '<<<') button, to view previous validation result of the current measurement category (if results are available);
- a (gray '>>>') button, to view next validation result of the current measurement category (if results are available);
- a (gray **'save ..'**) button, to save the validation results window, as a png image file, for further analysis;
- a (gray **'dup'**) button, to duplicate the validation results window for further analysis;
- a (gray **'zoom+'**) button, to zoom-in on validation plot;
- a (gray **'zoom-'**) button, to zoom-out on the validation plot;
- a (red **'quit'**) button, to close the validation results window;

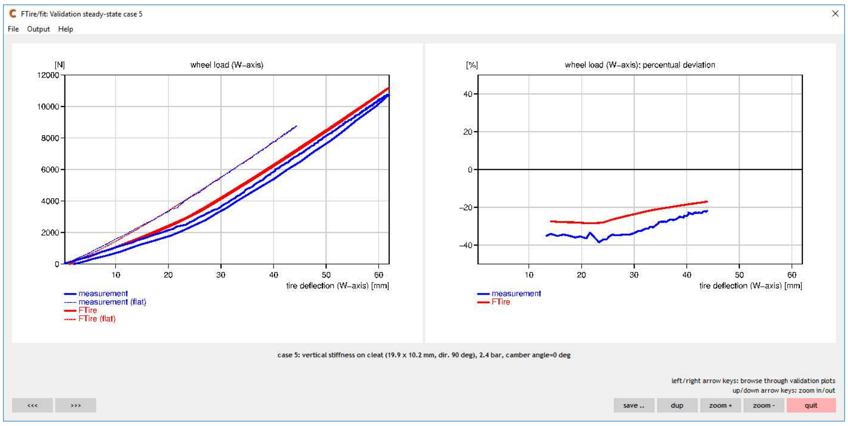

Figure 19: Validation window of a steady-state test case

4.3 Finishing Tasks

As mentioned above, the **fully automatic** and complete identification process will include all finishing steps, which include:

- the refreshment of all **validation** simulations, using the final state of the model data,
- an automatic **optimization of numerical settings** to increase convergence speed during tire data pre-processing,

- the computation of additional **tire model properties**, and
- the generation of the result **report**.

If identification has been performed **interactively**, however, these tasks have to be launched manually. As usual, **FTire/fit**'s workbench will only show those buttons, the selection of which is supported in the current project status.

Optimization of numerical data and **computation of additional FTire model properties** both can be selected at any time after initial creation of _best_so_far.tir, with the respective buttons in the '**analyse & finish**' tab of the launch pad. However, these steps are not required unless the final report is to be generated.

Once the additional tire model properties have been calculated, they can be **viewed** without re-calculation. Moreover, a report in pdf format can be generated (button **create report** in the main menu). Among others, this report will contain:

- report generation date,
- **FTire** and **FTire/fit** program versions that have been used,
- tire manufacturer,
- tire brand,
- tire size,
- inflation pressure(s),
- a listing of the latest model data file,
- a graphical representation of the cross-section geometry 'as seen' by **FTire**,
- a listing of additional tire properties,
- a listing of the cleat definition file, and
- comparison images of all validation cases that have been computed so far.

Please note that validation images **might not reflect the latest state of the data file**. This can happen if tire data has been changed, either by identification or manually, using the data file editor **cosin/tools**, without refreshing the validation. For this reason, it is recommended to launch a **compare all** validation is run **prior to the generation of the final report**, even if this step might take some computing time.

After creation the pdf report, together with copies of all important project and result files, will be saved in sub-folder **'report'** of the project folder.

5 FTire/fit*'s Menu-bar Functions*

The buttons in the menu bar of the **FTire/fit**window perform the following functions:

5.1 File

- **New project ..** – creates a new **FTire/fit** project;
- **Create/renew demo project** – creates a **FTire/fit** project from the supplied demo dataset. If a demo project already exists, then the project will be renewed;

- **Open project ..** – to open an existing **FTire/fit** project ;
- **Search project in** – to search for a an existing **FTire/fit** project in a list of standard and recently accessed locations;
- **Open last project** – to open the previously accessed **FTire/fit** project;
- **Open recent project** – to open an existing **FTire/fit** project from a list of recently accessed projects;
- **Open legacy prj-file ..** – to open a legacy **FTire/fit** project file;
- **Show project folder** – to view the project folder;
- **Clean project folder** – to delete all temporary files in the project folder, no undo available;
- **Clear identification history** – to clear all entries in the project history, no undo available. Refresh the project history list, by selecting the ('**refresh**') button in the project history section, to update the display;
- **Replace tire data file by new estimate ..** – to replace the current tire data file with a new estimate, created with the **cosin/tools: FTire Data File Generator**;
- **Replace tire data file by new other ..** – to replace the current tire data file with a new tire data file;
- **Replace cleat definition file ..** – to replace the current cleat definition file with a new cleat definition file;
- **check in test facility info file ..** – to check-in a test facility html info file, the information of the file will be used in the **FTire/fit** validation report;
- **check in test facility logo ..** – to check-in a test facility logo, the image will be used in the **FTire/fit** validation report;
- **Save current Ftire data as ..** – to export the current tire data file (.tir);
- **Run shell in private data folder ..** – to open a command window in the private data folder;
- **Show most recent program call ..** – to open the latest program call (run.bat) in a text editor;
- **Show most recent auxiliary program call ..** – to open the latest auxiliary program call (runaux.bat) in a text editor;
- **Quit** – close **Ftire/fit**;

5.2 Edit

- **Edit/analyze current data file** – to open the current tire data file in **cosin/tools for tires**;
- **Open identified data file in txt editor** – to open the current tire data file in a text editor;
- **Edit/analyze other data file** – to open a tire data file in **cosin/tools for tires**;
- **Open cleat definition file** – to open the project cleat definition file (contr.clt) with **cosin/tools for roads** for further analysis;
- **Edit FTire/fit control file** – to open the **FTire/fit** control file (contr.ftf) in a text editor (experts only). The file contains all the relevant information of the **FTire/fit** project. **FTire/fit** needs to be closed before the file is edited;
- **Edit Ftire/sim control file** – to open the **FTire/fit** simulation control file (contr.sim) with the **cosin Simulation Script Editor**. The file contains all the **FTire** simulation data that controls the interactive cleat test identification simulation. ;

- **Edit optimizer control file** – to open the **FTire/fit** optimizer control file (contr.opt) in a text editor (experts only). The file contains all the relevant information of the **FTire/fit** optimization routines that are used during the parameter identification;
- **Edit cleat definition file** – to open the project cleat definition file (contr.clt) with a text editor;
- **Notes and remarks** – to open the project remarks file (remarks.txt) with a text editor. This file can be used by the user to save project specific notes and remarks;

5.3 View

- **Show identification history** – to show/hide the project history in the **FTire/fit** window;
- **Show tooltips** – to show/hide the interactive help messages in the **FTire/fit** window;

5.4 Output

- **Hide message window** – hide **cosin** message window;
- **Show message window** – show **cosin** message window;
- **Show verbose msg window** – show the comprehensive log output in the **cosin** message window;
- **Animation off** – Switch the animations off;
- **Animation on** – Switch the animations on;
- **Browse log file** – open the **cosin** Messages of the most recent application call;
- **Browse auxiliary log file** – open the **cosin** Messages of the most recent auxiliary routine call;
- **Save log file as..** – save the **cosin** Messages of the most recent application call;
- **Print log file** – print the **cosin** Messages of the most recent application call;
- **Clear log file** – clear the **cosin** Messages of the most recent application call;
- **List files of last run** – list all files that were used and/or created during the most recent application call;

5.5 Settings

- **Preferences ..**, to view and edit the current **FTire/fit** project preferences. The **FTire/fit** project preference window (shown in figure 20) groups al available project preferences into the following tabs:

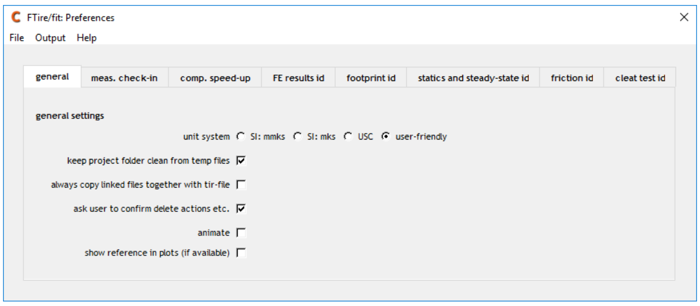

Figure 20: **FTire/fit** project preference window

- **general**, to define general project settings:
  - **unit system**, radio-buttons to define the unit system that is used in the **FTire/fit** project. The following options are available:
    * **SI: mmks**, millimeter/kilogram/second;
    * **SI: mks**, meter/kilogram/second;
    * **USC**, inch/pound/second;
    * **user-friendly** (default), SI user-friendly;
  - **keep project folder clean from temp files**, check-box to automatically delete all temporary files in the project folder;
  - always copy linked files together with tir-file, check-box to automatically save all linked files in the project history folder;
  - **ask user to confirm delete actions etc.** , check-box to set wether or not the user is promoted to confirm all delete, and similar, actions;
  - **animate** , check-box to show the **FTire** animation during all validation simulations (if useful);
  - **show reference in plots (if available)**, check-box to show all available references in the comparison result plots;
- **meas. check-in**, to define measurement check-in settings:
  - **try to determine meas. type from file name** (enabled by default), check-box to enable/disable the check-in feature that tries to identify the measurement type from the measurement file name;
  - **cautious autofill** (disabled by default), check-box to only use parameter values coded in the file name of the measurement file for autofill if the file name indicates the measurement type;
  - **in case of multiple meas. files of same kind**, radio-buttons to define the procedure that is followed when multiple measurements of the same kind are checked in. The following options are available:
    * i**gnore file**, to abort check-in of the latests measurement file;
    * **replace file**, to continue with the check-in, and replace the checked-in measurement with the latests measurement;
    * **add file** (default), to continue with the check-in, and to add the latests measurement file to the project;
    * **ask what to do**, to continue with the check-in, and to prompt user input if duplicate files are detected;
  - **in case of footprint measurement**, radio-buttons to define if the tread rib geometry should be determined from the footprint measurements. The following options are available:
    * **ignore rib geometry** (default);
    * **identify rib geometry**;
  - **in case of time-domain test, assume**, radio-buttons to define the measurement type, that is assumed, when a time-domain test is checked-in. The following option are available:
    * **cleat test**;
    * **steady-state test**;

    * **determine autom., using tolerances below** (default), if this option is defined, **FTire/fit** tries to automatically determine the type of measurement using a set of tolerances;
    * **in case of static measurement, assume**, radio-buttons to define the measurement type, that is assumed, when a static test is checked-in. The following option are available:
        · **vert. displ,** assume vertical displacement;
        · **horiz. displ,** assume horizontal displacement;
        · **determine autom** (default), if this option is defined, **FTire/fit** tries to automatically determine the type of measurement;
- **in case of cleat test**, radio-buttons to define if the FZ measurement offset should be taken into account. The following option are available:
    * **ignore Fz offset (for compat. with older versions)**;
    * **take into account Fz offset** (default);
- **use vertical stiffness from other tire data files** (disabled by default), check-box to define the ;
- **use tire design data file (tdd/fet file)**, radio-buttons to define use of the tire design data file. The following options are available:
    * **for rel. stiffness replacement**;
    * **for abs. stiffn. replacement**;
    * **for FE results id** (default);
- **adjust obviously wrong sings in F&M data files** (disabled by default), check-box to define if **FTire/fit** should automatically try to correct the sign convention used in the force and moment measurement files;
- **force stiffn. characteristics to pass through origin** (enabled by default), check-box to define if **FTire/fit** should automatically try to correct the offset in the force measurement files;
- **if applicable, include thermal properties in validation and identification** (enabled by default), check-box to define if the thermal data of the force and moment (F&M) measurements should be used during the check-in process. If this option is enabled, and the corresponding thermal data is available in the measurement files, then during check-in **FTire/fit** apply the necessary settings so that the thermal model is activated for the validation and identification simulations. Note, this might significantly increase the simulation time of these tests and is only required if the thermal model needs to be validated or identified;
- **TYDEX signal TRDTEMP is** , drop-down menu to define the FTire temperature channel that should be used. The following options are available:
    * tire structure temperature
    * tread surface temperature (center)
    * mean tread surface temperature
    * mean contact patch temperature
    * probe segment mean tread temperature
    * user defined sensor location;

- **ignore unkown OBSTTYPE entry in TYDEX files** (disabled by default), check-box to define the ;
- **inflation pressure recognition tolerance**, entry field to define the inflation pressure tolerance that is used to recognize the standard inflation pressure test cases;
- **modification factor variation tolerance**, entry field to define the modification factor of the minimum variation tolerance to recognize the respective horizontal static measurement;
- **threshold factor bottoming**, entry field to define the ratio of the maximum rated load (LI), above which it is assumed that the test is a bottoming test;
- **footprint shrinking factor**, entry field to define the shrinking factor by which the footprint comparison images are scaled. This factor has no influence on the identification process;
- **footprint margin**, entry field to define the maximum number of pixel shifts that may be used to align FTire's footprint with the measured footprint;
- **dynamic test tolerance 1**, entry field to define the maximum measurement duration (in s) for which a dynamic cleat test is assumed;
- **dynamic test tolerance 2**, entry field to define the percentage wheel load variation that, among others, is used to automatically identify a dynamic cleat test case;
- **dynamic test tolerance 3**, entry field to define the the maximum vertical wheel motion (in mm), that will assume a 'fixed hub condition';

• **comp. speed-up**, to define computation speed-up settings:
  - **steady-state sweep speed**, radio-buttons to define the steady-state sweep speed that is used for the validation and identification simulations. The following options are available:
    * **fast**;
    * **std** (standard - default);
    * **accurate**;
  - **FTire speed mode**, to define the run-time speed mode. The following options are available:
    * **0**, Standard mode, no model restrictions or modifications;
    * **1** (default), accelerated level 1, no model extensions (other than thermal and tread wear model);
    * **2**, accelerated level 2, no model extensions, no extra output;
    * **3**, accelerated level 3, no model extensions, no extra output, no animation coarse mesh, numerics optimized for speed;
  - **max. number of threads**, to define the maximum number of threads to be used by **FTire/fit** in multi-threading mode. Not effective if no FTire/mt license is available;
  - **min. number of threads**, to define the minimum number of threads to be used by **FTire/fit** in multi-threading mode;
  - **block FTire/fit while data file is edited** (enabled by default), check-box to block all access to the **FTire/fit** GUI while the tire data file is being edited in **cosin/tools for tires**. Enable this option to avoid instances where multiple tire data files are edited simultaneously;

• **FE results id**, to define the finite element identification settings:
  - **accuracy** (4 by default), entry field to define the parameter variations(1..5) during the FE results parameter identification process. 1=low ... 5=high;

  - **number of steps** (3 by default), entry field to define the number of refinement steps(1..10) that are take during the FE results parameter identification process;
  - **variation range** (3.0 by default), entry field to define the parameter variation range during the first FE results parameter identification step. The variation range must be grater than 1.0;
  - **favor FE results over statics measurement** (enabled by default), check-box to define the static measurements preference;
  - **maximum error value in visualization** (5mm by default), entry field to set the maximum error to be used for the color scale in the validation visualization;
- **footprint id**, to define the footprint identification settings:
  - **accuracy** (4 by default), entry field to define the parameter variations(1..5) during the footprint parameter identification process. 1=low ... 5=high;
  - **number of steps** (3 by default), entry field to define the number of refinement steps(1..10) that are take during the footprint parameter identification process;
  - **variation range** (3.0 by default), entry field to define the parameter variation range during the first footprint parameter identification step. The variation range must be larger than 1.0;
  - **favor favor footprints over radial stiffness on cleat** (enabled by default), check-box to define the footprint over radial stiffness on cleat preference;
  - **visualize measured footprint boundary** (enabled by default), check-box to enable the measured footprint visualization in the footprint validation images. The measured footprint outline is indicated with a blue line;
  - **visualize Ftire footprint boundary** (enabled by default), check-box to enable the simulated footprint visualization in the footprint validation images. The simulated footprint outline of FTire is indicated with a red line;
  - **show pseudo-convex hull of footprint boundary** (disabled by default), check-box to show the footprint boundaries in the validation images as a pseudo-convex hull;
  - wheel load controller gain (0.1 by default), entry field to define the wheel load controller gain that is used during footprint validation and identification. Reduce this gain if the the wheel load controller becomes unstable during footprint validation and identification simulations;
- **statics and steady-state id**, to define the statics and steady-state identification settings:
  - **accuracy** (4 by default), entry field to define the parameter variations(1..5) during the statics and steady-state parameter identification process. 1=low ... 5=high;
  - **number of steps** (3 by default), entry field to define the number of refinement steps(1..10) that are take during the statics and steady-state parameter identification process;
  - **variation range** (3.0 by default), entry field to define the parameter variation range during the first statics and steady-state parameter identification step. The variation range must be greater than 1.0;
  - **velocity range for belt ext. ident.** (20m/s by default), entry field to define the velocity range that is taken into account during the belt extensibility parameter identification;
    * **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;

- **force range for vert. stiffn. ident.** (100% of LI by default), entry field to define the wheel load range, relative to the second wheel load value used in the current tire data file, that is taken into account during the vertical stiffness parameter identification;
  - **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;
- **force range for lat. stiffn. ident.** (80% by default), entry field to define the force range, relative to the wheel load or maximum lateral force (defined by radio-buttons), that is taken into account during the lateral stiffness parameter identification;
  - **wheel load** (selected by default), radio-button to define if the force range for the lateral stiffness identification is determined relative to the applied wheel load, or to the
  - **max. lat. force**, radio-button to define if the force range for the lateral stiffness identification is determined relative to the maximum measured lateral force
  - **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;
- **torque range for tors. stiffn. ident.** (100% by default), entry field to define the torque range, relative to the reference torque or maximum aligning torque (defined by radio-buttons), that is taken into account during the torsional stiffness parameter identification;
  - **ref. torque** (selected by default), radio-button to define if the torque range for the torsional stiffness identification is determined relative to the reference torque (ref. torque = 1/4 tread width * wheel load), or to the
  - **max. aligning torque**, radio-button to define if the torque range for the torsional stiffness identification is determined relative to the maximum measured alining torque
  - **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;
- **long. slip range for mu-slip stiffn. ident.** (3% by default), entry field to define the longitudinal slip range that is used for the slip stiffness parameter identification;
  - **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;
- **slip range for corner. stiffn. ident.** (3% by default), entry field to define the slip angle range that is used for the cornering stiffness parameter identification;
  - **average ambiguous values** (enabled by default), check-box to use the mean value, in case of hysteresis ambiguity. If this option is disabled then the first value is used;
- **generate additional output files** (disabled by default), check-box to request that additional output files are generated during the static and steady state simulations;

- **friction id**, to define the friction characteristic identification settings:
  - **accuracy** (4 by default), entry field to define the parameter variations(1..5) during the friction characteristic identification process. 1=low ... 5=high;
  - **number of steps** (3 by default), entry field to define the number of refinement steps(1..10) that are take during the friction characteristic identification process;
  - **variation range** (1.5 by default), entry field to define the parameter variation range during the first friction characteristic identification step. The variation range must be greater than 1.0;

- do not take into account stick-slip cases (disabled by default), check-box to ignore stick-slip cases during friction validation and identification;

- **cleat test id**, to define the cleat test identification settings:
  - **accuracy** (4 by default), entry field to define the parameter variations(1..5) during the cleat test characteristic identification process. 1=low ... 5=high;
  - **number of steps** (3 by default), entry field to define the number of refinement steps(1..10) that are take during the cleat test identification process;
  - **variation range** (3.0 by default), entry field to define the parameter variation range during the cleat test characteristic identification step. The variation range must be greater than 1.0;
    * **set individually** (disabled by default), check-box to enable the individual parameter variation range option. If enabled parameter variation ranges can be set for all parameters relevant for the cleat test identification. The ranges can be defined in the FTire/fit main window;
  - **assume test rig is**, radio-buttons to define the compliance of the cleat test rig. The following options are available:
    * **ideally stiff** (selected by default);
    * **compliant (if data is available the cleat definition file)**;
  - **accelerated pr-processing** (disabled by default), check-box to enable accelerated pre-processing during the dynamic cleat test parameter identification process. Note: this slightly increases the risk of divergence problems during pre-processing;
  - **cleat test evaluation duration**, radio-buttons to define the test duration that is used for the cleat test parameter identification process. The following options are available:
    * **std** (selected by default), standard evaluation duration;
    * **long,** extended evaluation duration;
  - **cleat test evaluation mode**, radio-buttons to define the cleat test evaluation mode that is used for the cleat test parameter identification. The following options are available:
    * **std (on-cleat and off-cleat)** (selected by default);
    * **only on-cleat**;
    * **only off-cleat**;
    * **only on-cleat peak values**;
  - **exponent used in objective function** (2.0 by default), entry field to define the exponent that is used in the objective function. The higher the exponent, the more peak deviations are taken into account;
  - **minimum cleat event delay time** (0.01s by default), entry field to define the minimum time delay of the cleat event that is used in the plots. Actual delay will be drum/track speed dependent
  - **measurement data point increment** (1 by default), entry field to define the measurement data point increment that is taken into account. 1= use every point, 2 = use every second point etc;
  - **write extra plot files during identification** (disabled by default), check-box to request that additional output files are generated during the dynamic cleat test identification;
  - **y-axis range in PSD plots** (-20 .. 60 dB/Hz by default), entry field to define the y-axis range in PSD validation plots of the dynamic cleat tests;

  - wheel load controller gain (1 by default), entry field to define the wheel load controller gain that is used during dynamic cleat test validation and identification. Reduce this gain if the the wheel load controller becomes unstable during dynamic cleat validation and identification simulations;

- **Reset id settings** – reset all id settings back to default;
- **Choose text editor ..** – to specify the text editor that should be used to open ascii files;
- **Choose pdf reader ..** – to specify the pdf reader that should be used to open pdf files;
- **License ..** – to specify and query the **cosin** license server details;

5.6 Help

- **cosin docu** – opens the **cosin** documentation, with links to all cosin documentation, user guides and copyright information;
- **FTire docu** – opens the **FTire Modelization and Parameter specification** documentation;
- **Support checklist** – opens the trouble-shooting guide, **Support checklist**, that should be checked before requesting technical support;
- **Load index table** – opens the load index (LI) table
- **Speed index table** – opens the speed index table
- **TYDEX axis system docu** – opens the **TYDEX axis systems** documentation;
- **TYDEX file format docu** – opens the **TYDEX Description and Reference Manual;**
- **About cosin/tools** – displays information about the installed **cosin** software version, revision and installation directory;
- **www.cosin.eu** – link to the **cosin** website