SOFTWARE NEED ASSESSMENT

OptimumG - Optimum Tire partnership for the 2020 Formula Student season

Direction and management

JACQUET Mathieu (Team Leader) mathieu.jacquet@ecl18.ec-lyon.fr
RODRIGUEZ Arthur (Team Manager) arthur.rodriguez@ecl18.ec-lyon.fr

Suspension system

KAWCZYNSKI Martin (Department leader) martin.kawczynski@ecl18.ec-lyon.fr
SCHIO Michele michele.schio@ecl18.ec-lyon.fr
DELORT Arthur arthur.delort@ecl18.ec-lyon.fr
CHARKALUK Paul paul.charkaluk@ecl18.ec-lyon.fr
ARIAUX Pierre-Emmanuel pierre-emmanuel.ariaux@ecl18.ec-lyon.fr
DE OLIVEIRA Victor Hugo victor.de-oliveira-pinto@ecl18.ec-lyon.fr

Version 0.4 -April 23, 2019

Contents

1 Introduction

1.1 EPSA organization at École Centrale de Lyon

École Centrale de Lyon (ECL) is one of the oldest engineering schools in France. It is located in Lyon. *Écurie Piston Sport Auto* (EPSA) is a spin-off organization composed of ECL students. The history of EPSA started with the competition of the *Société des Ingénieurs de l'Automobile* (SIA) in which several teams of the organization participated. Nowadays, our team is one of the five Formula Student (FS) teams in France.

As engineering students at ECL, we follow engineering fundamental classes during the first two years which give us a general view of the engineering process on the industrial level. At the end of the second year and during all of the third year, we choose where and how to specialize. In order to prepare us for such a specialization choice, during the first two years the school lets us develop an annual engineering project. The Formula Student project is today the biggest of the school students' projects.

During Mars and April 2019, some of the members of the team attended for the first time to the two OptimumG seminars in Nevers, France.

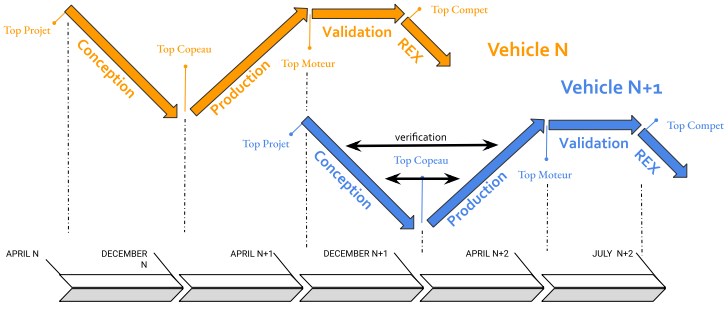
1.2 Tire models at the EPSA

The study of tire models, such as the Pacejka one, had started some years ago and since then we realized different Matlab and Python scripts which show and prove our understanding of the models. On the rush of the design process for the next season's vehicle, those scripts barely suffice and do not allow us to quickly analyze a great amount of raw data, such as the Tire Testing Consortium (TTC) data.

The design process at the suspension department follows the same algorithm as in all of the departments of the team as described in fig.1(a). Before deciding the suspension geometry structure we proceed by detailing the performance of the system though out the study of different load-cases. In a first approach we start by listing all of the relevant load-cases in function of the competition goals that we had fixed for the season. During a second time we run these load-cases by means of CAD software [6] and we propose a preliminary design concept. Finally we study such a concept with respect to the suspension kinematics, cost and feasibility and we iterate this procedure until we complete the preliminary design phase with a realistic solution.

In such a stage Optimum Tire could help us take a fundamental step towards a load-case validation with respect to the tire behaviour and thus reduce the gap between our engineering model and the real world. Since EPSA teams are among the youngest of the European FS, our cars have a history of being a little heavy and we are addressing this problem, season after season, in order to improve the vehicle performance. We list here two of the main tasks of the suspension department within the next season vehicle preliminary design:

- **wheel diameter** : study the difference between 10" and 13" tires
- **tire maker** : choose a tire maker and model



(a) Engineering process at EPSA

(b) tested software version

Figure 1

1.3 Report's goal

After the two seminars held by OptimumG at Nevers in April 2019 we understood the need for using tire models during the preliminary design phases. Our team as FS team has access to the Tire Test Consortium (TTC) raw data for the different tire models available for the competition. As of today the few hundreds lines of Matlab and Python code that we wrote in order to fit Pacejka's models to some old TTC data do not suffice in giving us the tire properties needed during preliminary design. Having a software such as OptimumTire on our side would surely make a great difference.

The main objective of this report is to present to you the kind of feedback we can produce while using a software such as OtimumTire. You can see below an example of report we have done while following some of the tutorials of OptimumTire.

Moreover, if you are interested in having a feedback from the user point of view of specific characteristics of your software, we shall be grateful to help you.

1.4 Report's structure

By following the first three tutorials proposed on the OptimumG (documents [1], [2] and [3]) website and by searching for the questions we posed ourselves in the official documentation (documents [5] and [4]), we eventually noted down:

- some of the difficulties we encountered,
- some graphical features of the software user interface that we think might me improved,
- some presentation aspects related to the fore-mentioned documents that we think might be structured differently

In the following two sections, we redacted a short description of those notes. We state here and for the rest of the document that the OptimumTire software was tested on a up to date Windows 10 environment. The OptimumTire software version has been captured in fig.1(b).

2 First impressions on the Optimum-Tire documentation

2.1 Tutorial part 1 : Importing data

We executed tutorial [1] with some raw TTC data. Even if OptimumTire prompted the user to load a `.csv` file, we gave the software a `.dat` file and everything seemed to work properly. We managed to create a custom *TTC_SI* template and import the `.dat` data. The only thing we could not figure out was the reading of tire temperatures as in fig. 2. We eventually searched in the help document [5] (section 2.3).

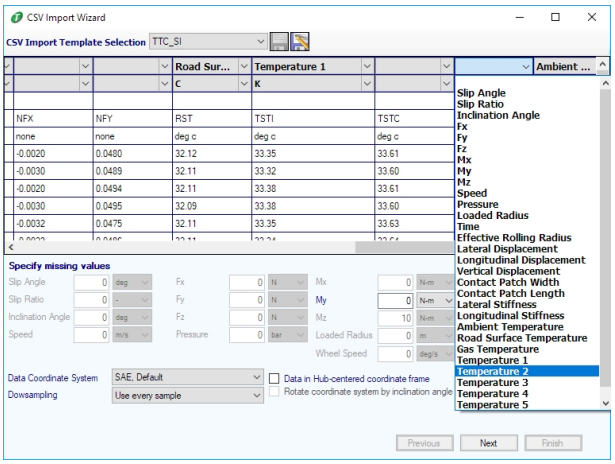
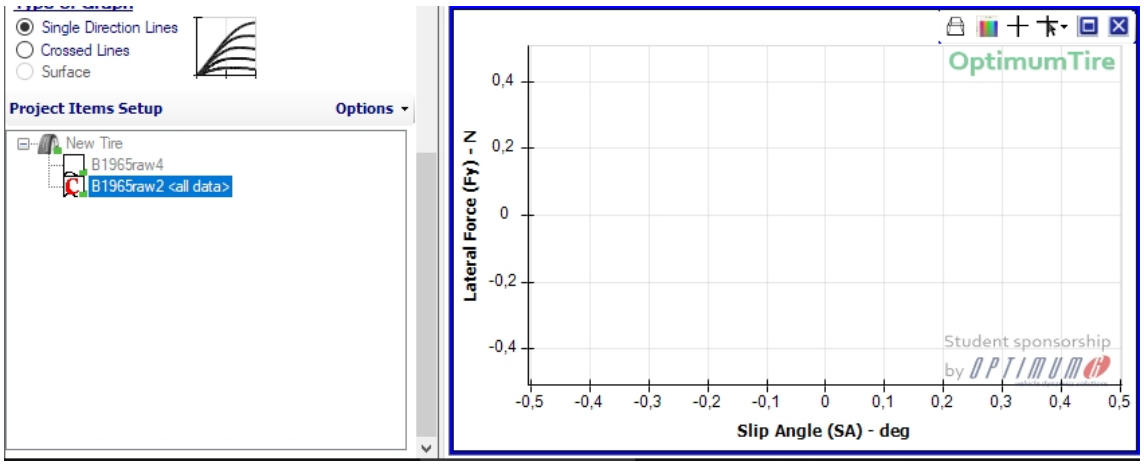
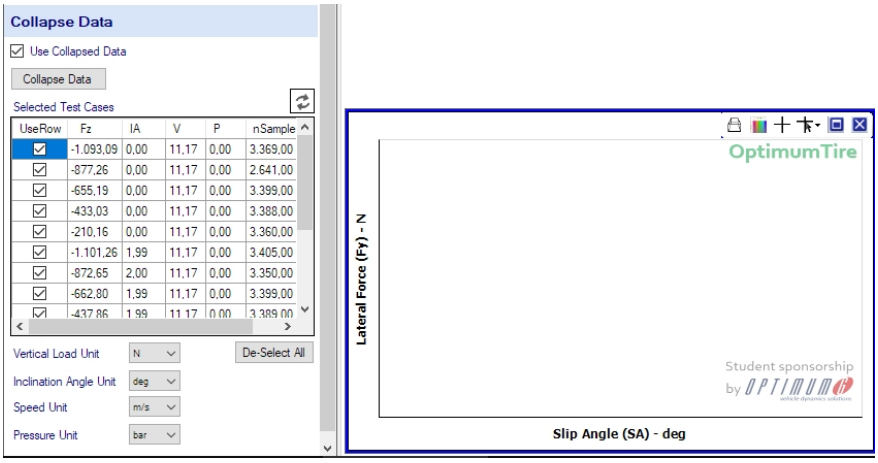


Figure 2: Temperature fields on the raw data import window



(a)



(b)

Figure 3: collapse tool plotting issues

2.2 Tutorial part 2 : Data Pre-processing

During tutorial [2], we found some difficulties in understanding the collapse tool and visualizing the collapsed data. Once understood the tool works very well but at the beginning it was not clear what tolerance meant. From our point of view, words like *half-step-size* or *half-step-difference* might represent a valid alternative to use instead of *tolerance* in the collapse window. More on the collapse window in section 3.2.

We think the crop tool is effective and very well structured but it might be improved by allowing the user to save a copy of the cropped data instead of overwriting the loaded data in the tire tree. This would possibly result in a quicker work-flow of big data pre-processing since the raw-data reading (from the hard drive) takes some time.

We eventually had some problems in plotting collapse data when we imported some raw TTC data different form the one used in the tutorials. The import process seemed to work properly even if we were using a `.dat` file and the plotting windows seemed to display correctly the imported data since both the graph tool and the crop tool had no problem displaying all of the data points. The issue became evident after we collapsed the data (fig. 3(a)) : no points were plotted even after a quick check of the *Set All Items as "Plot All Data"* property.

Initially we thought the issue was related to Windows software requirements (we were running OptimumTire on Windows 10 machine) but this did not explained why the collapse points form the tutorial data plotted correctly. We tried then to convert the `.dat` file into a `.csv` format using the same syntax as in the tutorial data. On a first try we did so by using Microsoft Excel and on a second try we used OptimumTire CSV Export tool. Both cases resulted in not plotting any point after the data was correctly collapsed by the software (fig. 3(b))

2.3 Optimum-Tire help File

During the tutorial [1] we happened to have a question about tire temperatures. When importing from a TTC data file we had three temperatures (inner, center and outer) and we could not associate these temperatures to the temperatures available in the importing window (temp 1,2,3,4 and 5 as in fig.2). By searching the word *temperature* in the help file (document [5]) we did not find any match.

3 First impressions on the Optimum-Tire user interface

3.1 About the raw data import flow

During tutorial [1] when importing raw data the software interface asks the user a `.csv` file. All of our TTC raw data is stored in `.dat` format, which is a text file like the first one. We might suggest the user interface could be updated to ask a data file without specifying the format since the software itself asks for the data separator character before starting the import process (fig. 4).

3.2 About the collapse tool

During tutorial [2] when we used the collapse tool we found the method of entering the tolerances values a little repetitive. OptimumTire prompts the user with some text fields in a first window (fig.5(a)) and only later allows to visualize the curves on a second window (fig.5(b)) which must be closed in order to enter back the values on the first window. We think it might be quicker to allow the user to plot the curves and enter the tolerances values in the same window.

By the same argument we think it might be more effective to simplify the third and fourth step of the data collapse tool (fig.s5(c) and 5(d) respectively) in a single step which allows the user to graphically verify the collapsed result. We would like to propose in fig.6 a possible layout solution for a single window collapse tool. The peculiar characteristics of such a layout would be to allow the user to graphically verify and obtain the information needed for properly filling the collapse data form.

3.3 On the model fitting window

During tutorial [3] we found the model fitting process very effective and smooth. However we would like to underline some minor strains which could be ameliorated in a second moment. In the convergence graph window of fig.7(a) all of the buttons might be compacted at the same place. In the fitting summary window of fig.7(c) we could not figure out why the window was displayed below the convergence graph window (fig.7(a)) and why there was such a big window for such little textual information. When re-opening the model fitting tool, the Advanced fitting options window appeared (fig.7(d)) and we wondered why those information could not have been integrated with the main model fitting tool window. Finally we encountered some problems when re-fitting a model after having deselected some collapsed data series (fig.7(b)).

3.4 Some more general thoughts

OptimumTire user interface is both simple and complete, however we would like to suggest some mouse feature that we think should be standardized between scientific software. We would like to address the mouse behaviour

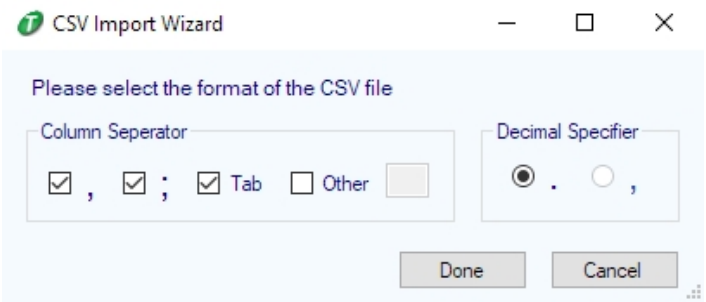
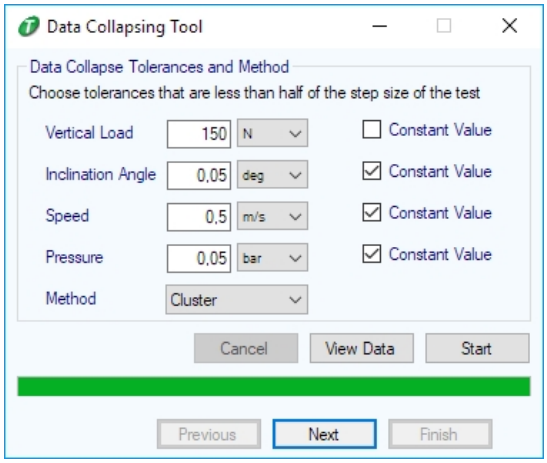


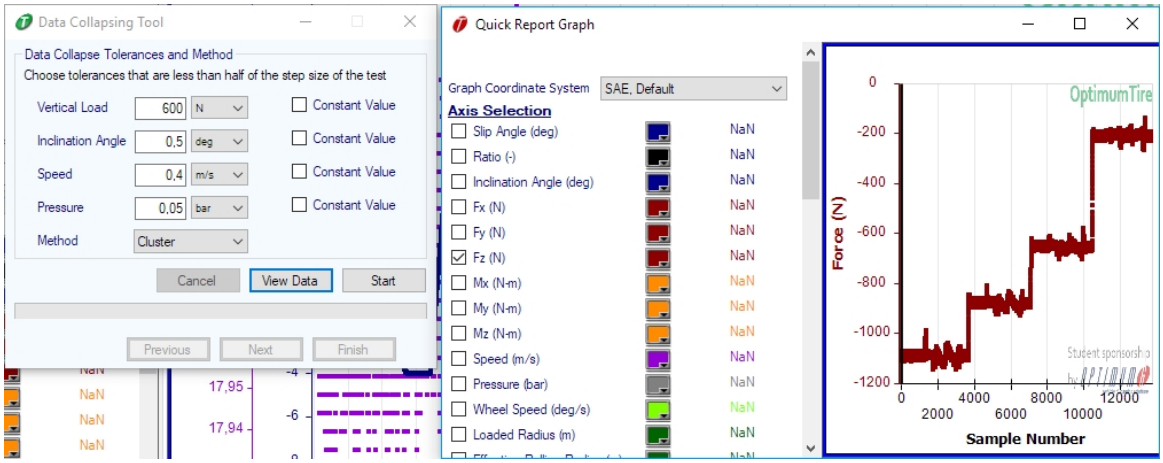
Figure 4: the `.csv` import window asks for the separator character

over plot figures (fig.8(b)). We found the middle button zoom feature and the absence to a direct click-to-pan tool a little odd. From our point of view, over a plot figure, left click and drag should zoom, middle button click should allow to pan and in order to move along the vertical (horizontal) axis the scroll wheel (with `shift` respectively) should be used.

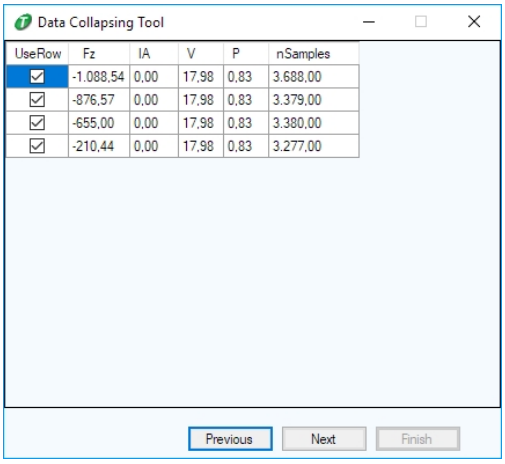
When creating a new worksheet we would like to be able to name the new worksheet directly on the sheet first creation and not to go through the right click contextual menu as in fig.8(a).



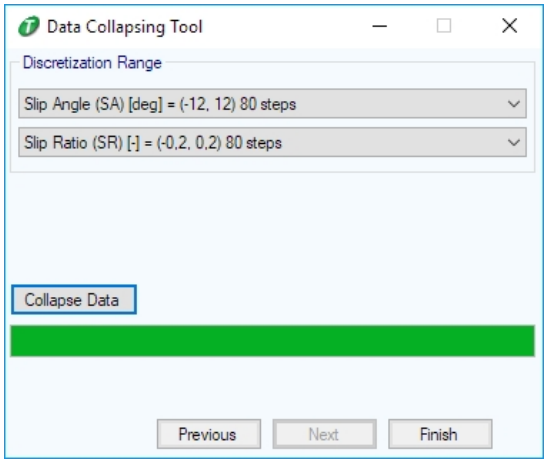
(a)



(b)



(c)



(d)

Figure 5: collapse tool screen-shots

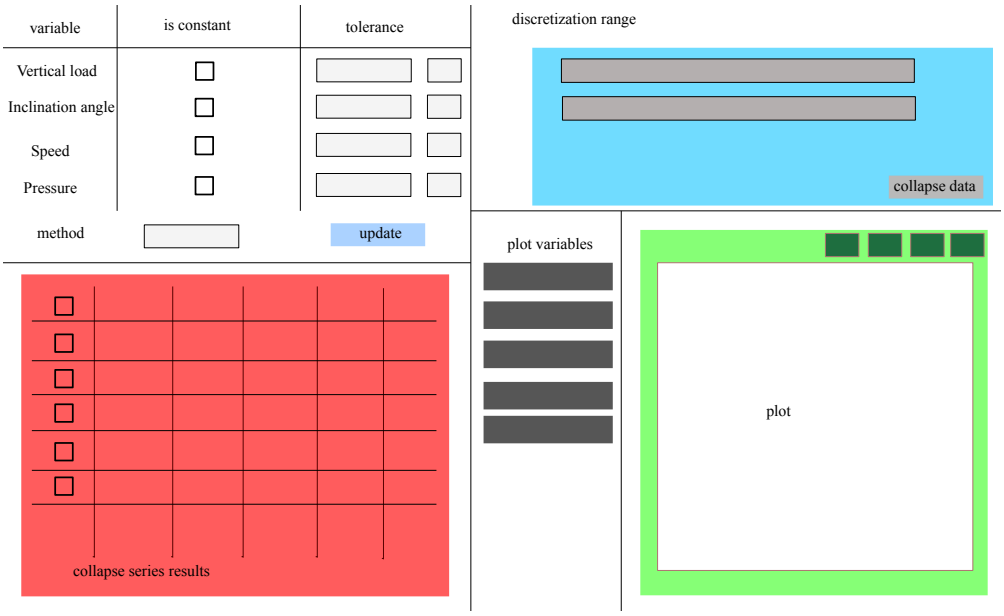
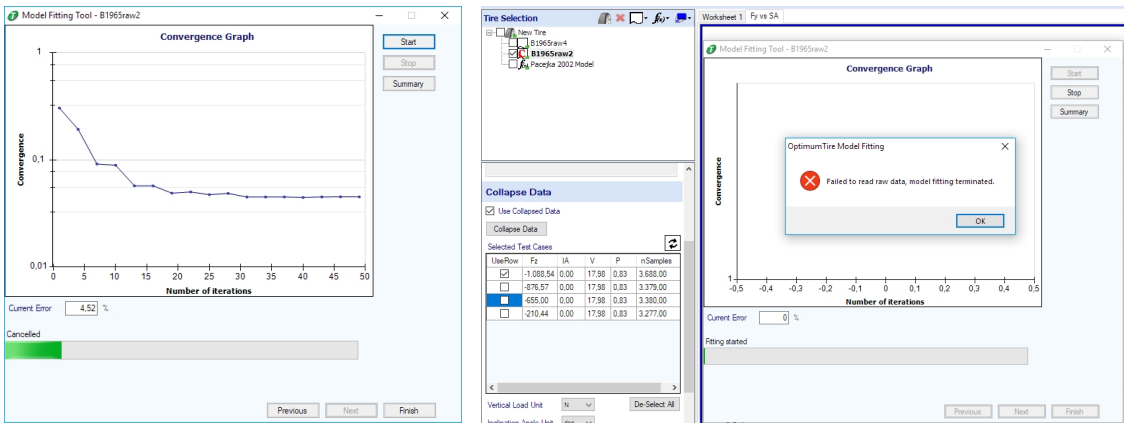
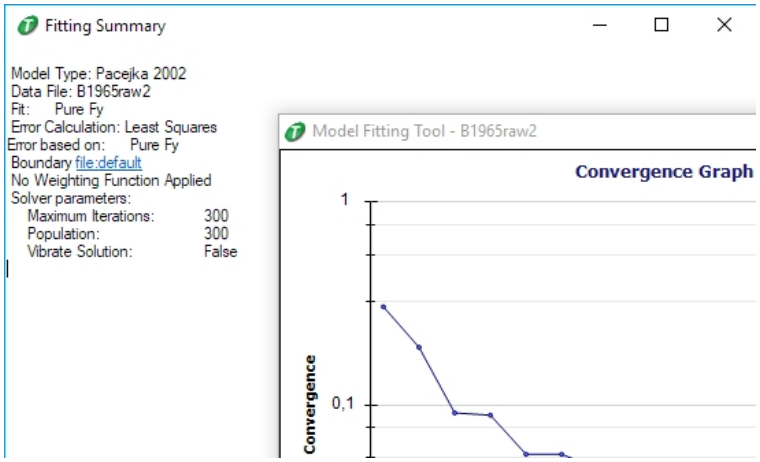


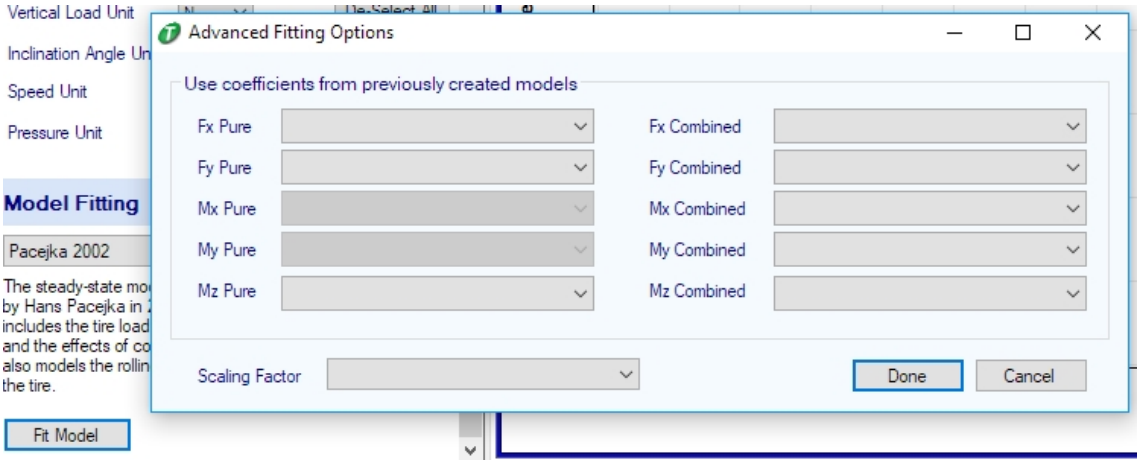
Figure 6: A possible layout for the collapse tool



(a) (b)

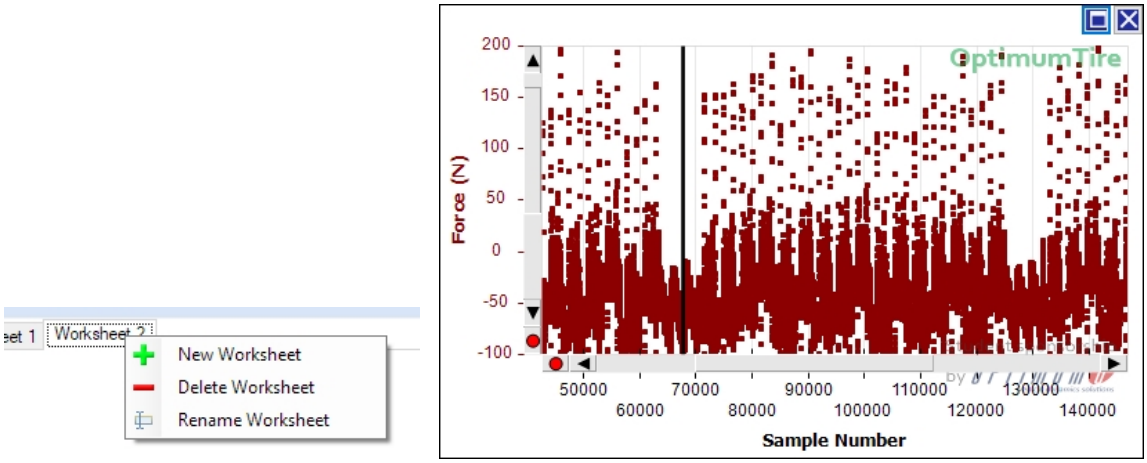


(c)



(d)

Figure 7: model fit tool screen-shots



(a) (b)

Figure 8: more general user interface thoughts

4 References

[1] Tutorial part 1: Importing Data, OptimumG® website, OptimumTire Tutorials (last retrieved: April 23, 2019)

[2] Tutorial part 2: Data Pre-processing, OptimumG® website, OptimumTire Tutorials (last retrieved: April 23, 2019)

[3] Tutorial part 3: Fitting a Model (Cornering), OptimumG® website, OptimumTire Tutorials (last retrieved: April 23, 2019)

[4] OptimumTire Product Specification, OptimumG® website, OptimumTire Documentation (last retrieved: April 23, 2019)

[5] OtimumTire Help File, OptimumG® website, OptimumTire Documentation (last retrieved: April 23, 2019)

[6] MECAmaster: 3D Tolerancing/Loads Simulation and Analysis, MECAmaster, website main page (last retrieved: April 23, 2019)