Tutorial 4 – Fitting a tire model, Part 2

Welcome to this tutorial in which we will look at how to fit a full tire model to pre-processed data. We will continue where Tutorial 3 ended.

1. Fitting strategies
2. Fitting Mz Pure
3. Fitting Fx Pure and Combined
4. Fitting Fy and Mz Combined
5. Reviewing results

1. Fitting strategies

In the previous tutorials we have looked at how to take the steps from lateral force data to a lateral tire model. A tire however does more than just generate lateral forces. When fitting a tire model that takes both longitudinal and lateral forces and moments into account we can go about it in a couple of different ways. The most apparent way would be to load all the data we have in to OptimumT and let the fitting algorithm do the work. Although this would be possible, it would be more difficult to tune the fitting process to get a good result. By fitting separate parts of the tire model individually, it will be easier to evaluate the quality of the fit and make adjustments.

In this tutorial we will use a fitting strategy that works quite well in most cases. However, depending on your goal and the specific data that you have it can be worthwhile trying other fitting strategies to generate a tire model to better serve your needs.

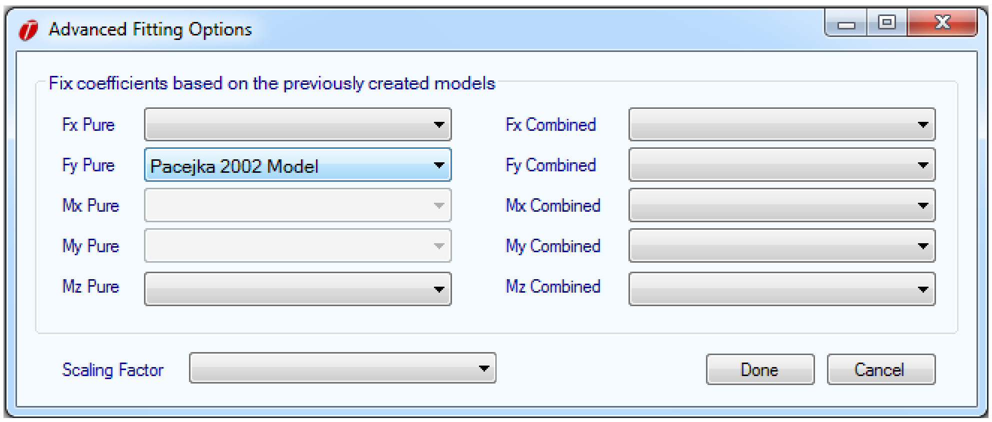
Fitting steps:

1. Fit pure lateral model (Fy)
   a. This means slip angle sweeps and free rolling tire (done in Tutorial 3)
2. Fit pure self-aligning torque model (Mz)
   a. This means slip angle sweeps and free rolling tire
3. Fit pure and combined longitudinal model (Fx)
   a. This depends on how your test data has been generated, but typically in combined testing, the slip ratio is sweeped at different slip angles.
4. Fit combined lateral (Fy) and self-aligning torque (Mz) models
   a. Once the combined longitudinal tire model has been fit, we will use that model as we fit the combined lateral model.

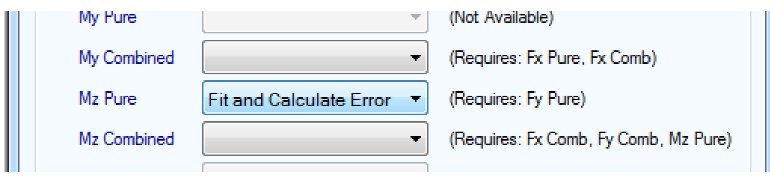
You might wonder why we are not fitting over-turning moment (Mx) and rolling resistance (My). The reason is that the same methodology applies for this. But Mx and My are not used as often in vehicle simulations, so if you are looking to do vehicle handling analysis you primarily will be looking at Fx, Fy and Mz.

2. Fitting self-aligning torque model

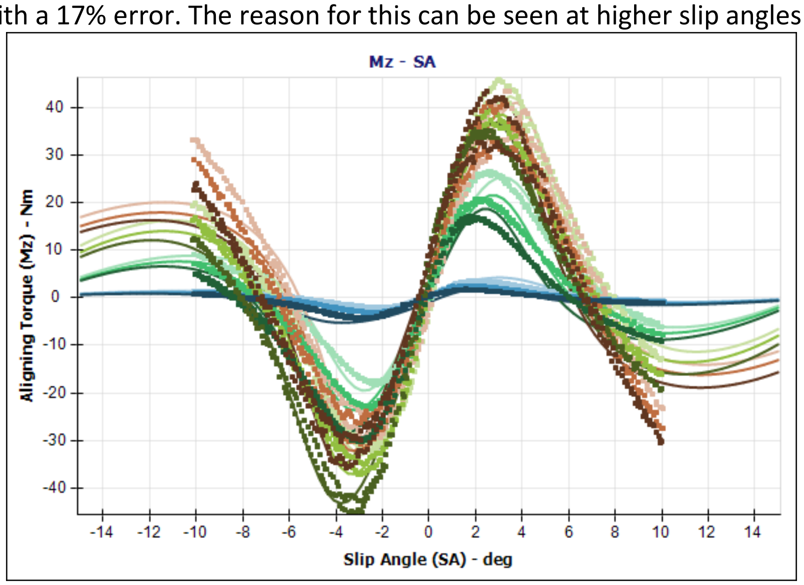
We will continue to use the lateral model we fit in Tutorial 3. To fit the Mz model, we once again click on the collapsed tire data in the tree. Scroll down and select a Pacejka 2002 and click *Fit Model*. The window *Advanced Fitting Options* will appear, in this window we can configure which previously fit models we want to base our new model on. In this case we want to base our new model on our Fy model, that we fit in the previous tutorial. We select it in the Fy Pure box. This means for our new model, the Pure Fy coefficient will be taken from the old model.



In the next window, the *Model Fitting Tool* window, we select that we want to fit and calculate error for the Pure Mz.



We can then proceed and start the fitting process in the same way we did in Tutorial 3. In this case the fitting ended up with a 17% error. The reason for this can be seen at higher slip angles and higher loads.

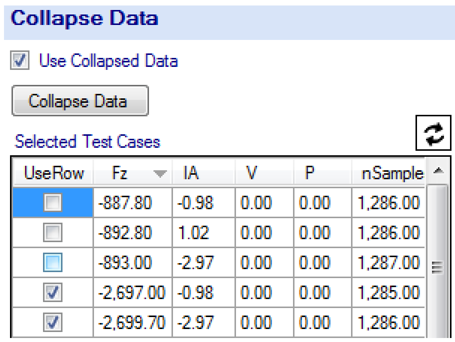


Before we proceed we will try to achieve a better fit for the Mz. There are many different ways we can go about doing this; we can manually adjust/tweak the coefficients until we find something that looks better. We can also work with coefficient boundaries and weight factors to get the fitting algorithm to focus on specific areas of the data.

But in this case I know that the lowest load is not of very much interest for me. So I will exclude that load in my fit, by de-selecting it in the *Collapse Data* table.

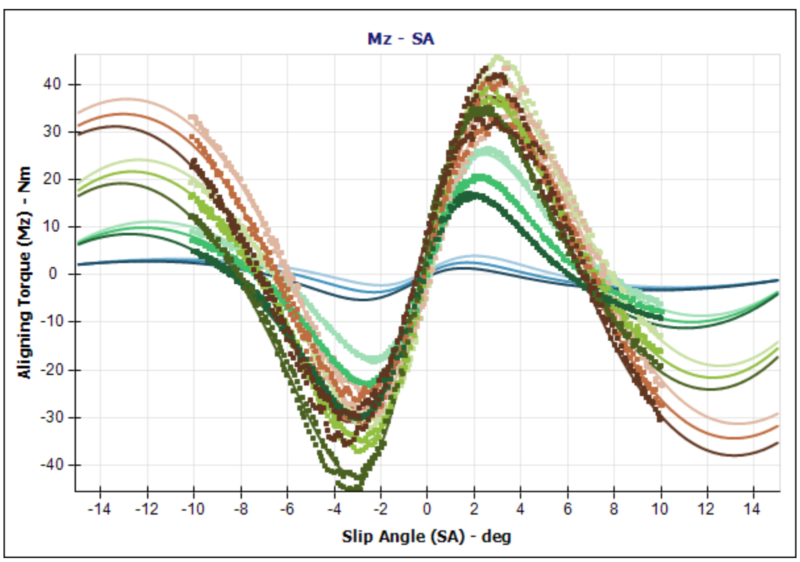
Did you know:

If you manually change a coefficient value, just click in the textbox and press Ctrl-Z to go back to the previous value.



UseRow	Fz	IA	V	P	nSample
☐	-887.80	-0.98	0.00	0.00	1,286.00
☐	-892.80	1.02	0.00	0.00	1,286.00
☐	-893.00	-2.97	0.00	0.00	1,287.00
☑	-2,697.00	-0.98	0.00	0.00	1,285.00
☑	-2,699.70	-2.97	0.00	0.00	1,286.00

I then repeat the fitting process and the lowest load will not influence the fit. This time the fit will come out with an error of only 6%. Generally, the error for Mz fits is larger than Fy and Fx fits (a good Mz fit could have an error of 10%, compared to 5% for a Fy or Fx).



It should also be mentioned that OptimumT's fitting algorithm is nondeterministic; this means that for a given set of input data, the output will be different every time. This means that if you are unsuccessful with one fit, doing a second fit with the exact same inputs might yield a better result. However, if you experience this behavior, it indicates that the coefficient boundaries that you are using are not a good match for the data and they might need to be adjusted. With coefficient boundaries that match the tires characteristics, OptimumT will get very close to the best possible fit every time.

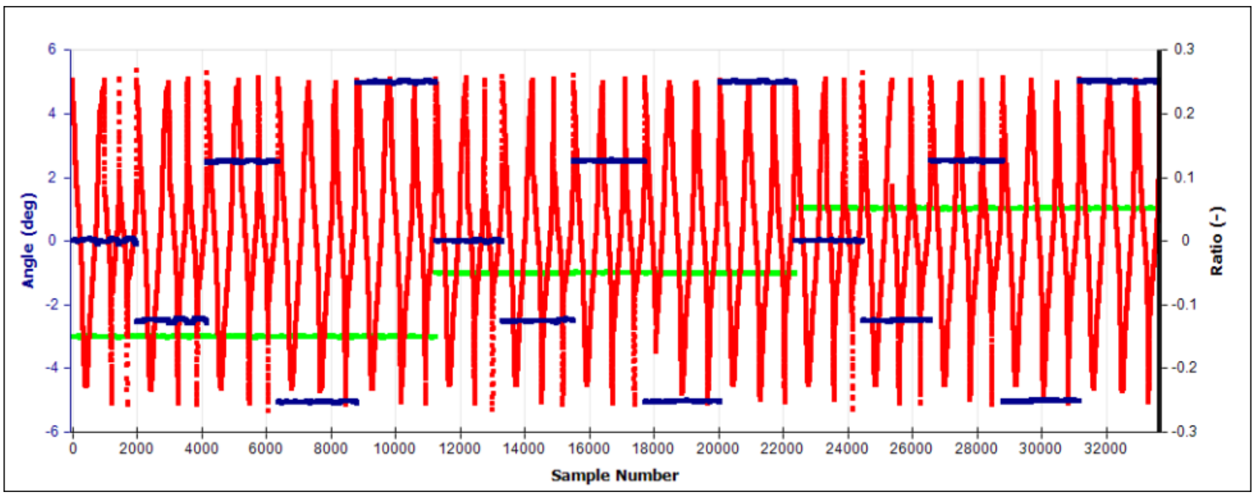
We now have a tire model which has coefficients fit for both Fy and Mz, so let's move on to the longitudinal parts of the model.

3. Fitting Fx (Pure and Combined)

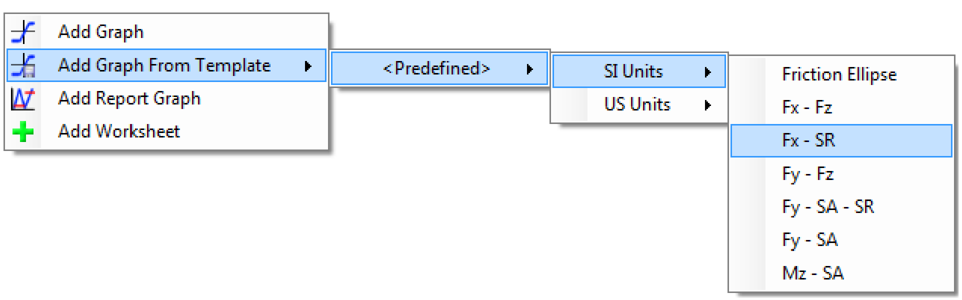
We will now take a look at fitting longitudinal data, which means looking at longitudinal force (Fx) as a function Slip Ratio (SR).

Depending on how your tire data has been collected, you might have to import another data file (it is common to do lateral and longitudinal/combined testing as two separate tests). In this case, we are going to load another file containing combined testing. We follow the same steps as we did for the lateral data file (importing, cropping and collapsing).

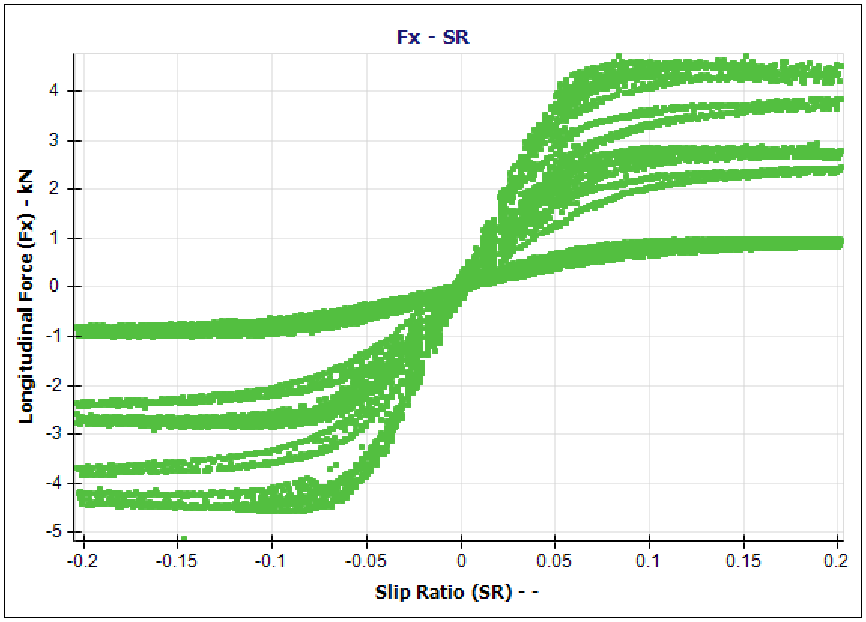
Using the *Report Graph* we can get a better idea of how the test was performed. We can see the different Inclination angles that the tire was tested at (green), the different (fixed) slip angle and all the slip ratio sweeps (red).



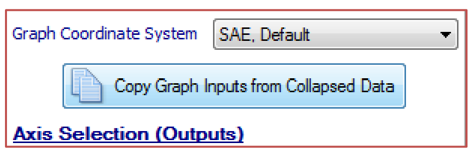
After having done all the pre-processing and looked at the raw data, we can look at the data in a Fx-SR graph. Right-click in the workspace and select to add a new graph from template.



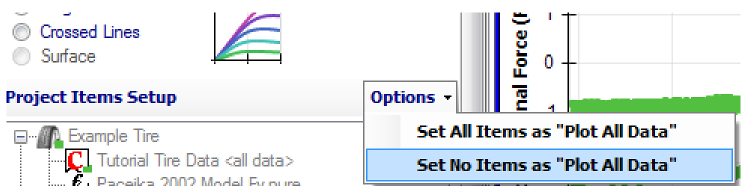
The graph will now show all the data in a single color.



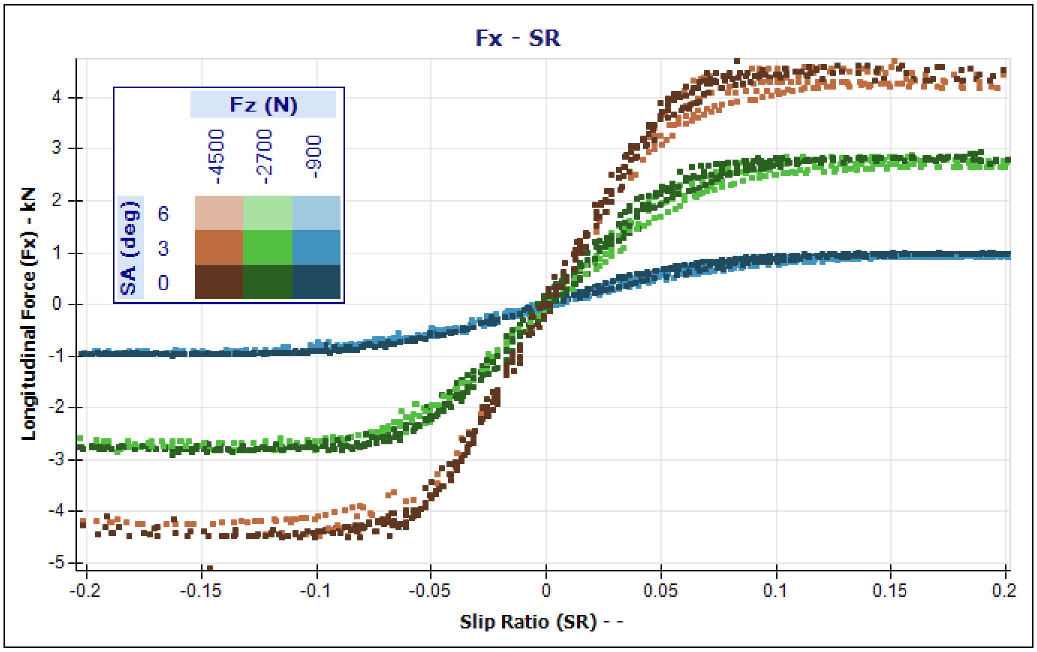
In the same way as was shown in Part 3 of Tutorial 2, we will use the collapsed data to enhance the graph. We will press the *Copy Graph Inputs from Collapsed Data* button to quickly retrieve the graph settings.



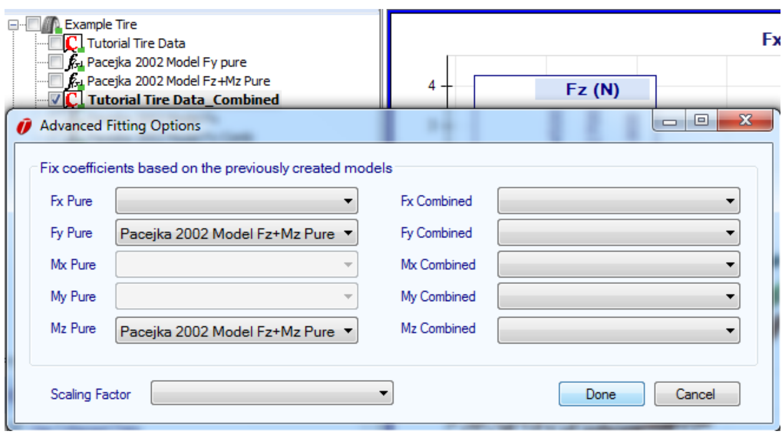
We will then set the graph to use these settings by clicking on, *Set No Items as "Plot All Data"* in the *Options* button.



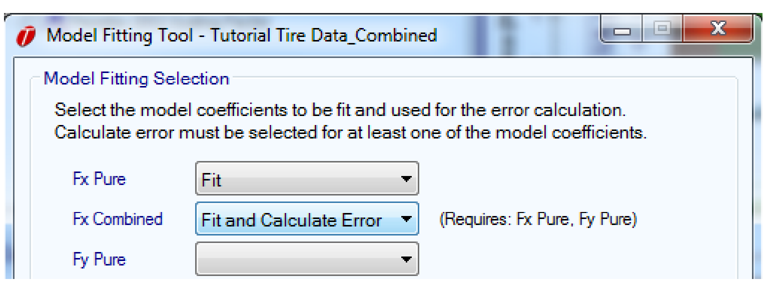
We should now have a graph that is colored by the testing cases identified by the collapsing tool. We can in the graph see that it is colored by Vertical Load and Slip Angle. This data was collected during a combined test. In this test several slip ratio sweeps are done at different vertical loads, slip angles and inclination angles. When we are fitting the pure Fx values, OptimumT will only look at the test case when the slip angle is zero. For the combined models, OptimumT will take all data into account.



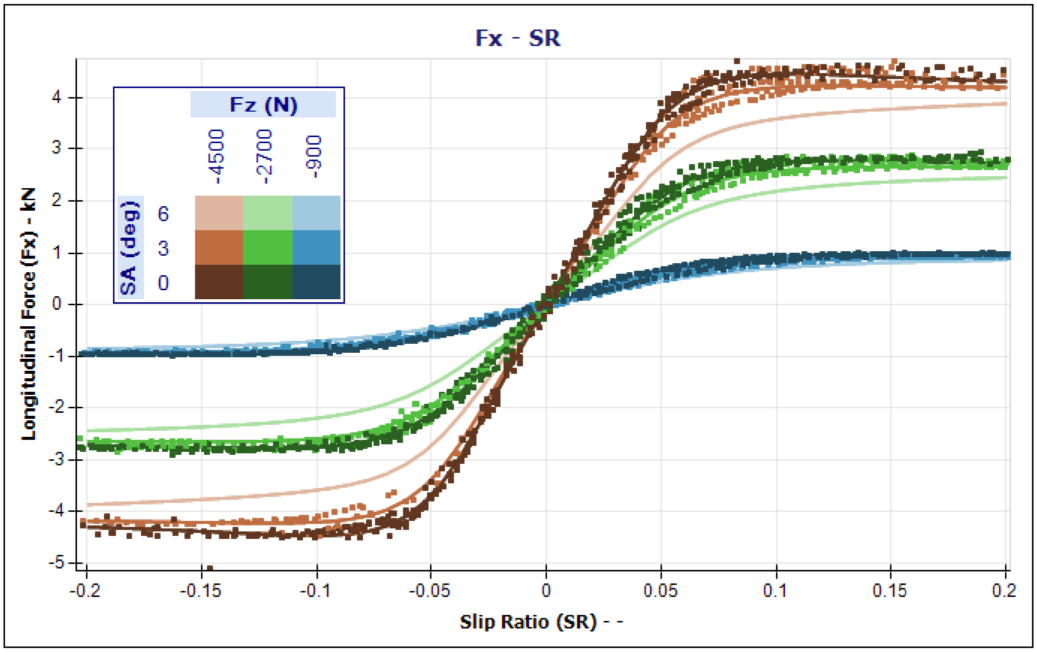
We click on the combined tire data and select "Fit Model". In the *Advanced Fitting Tool* we will select to keep the coefficients from our previously created model, the model containing coefficients for Fy Pure and Mz Pure.



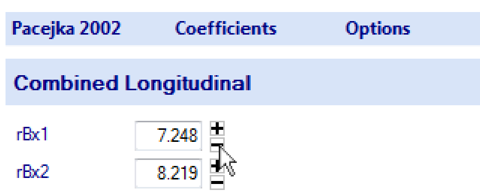
In the next window, we will select to *Fit* for the Fx Pure data and *Fit and Calculate Error* for the Fx Combined data. Since Fx Combined will rely on Fy Pure, a good fit of Fx Combined will typically result in a good fit for Fy Pure.

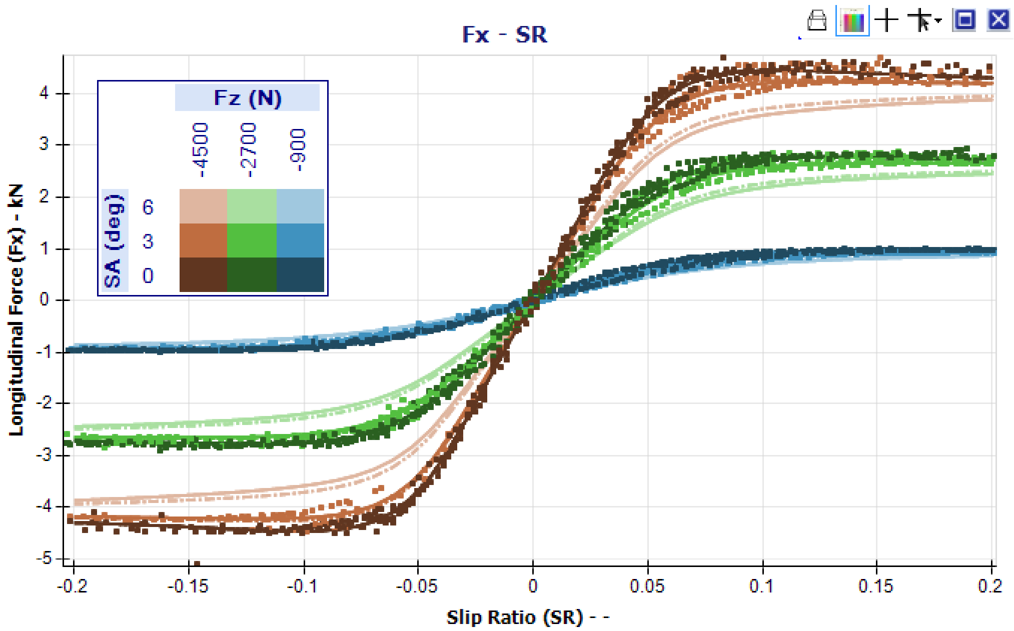


We can then go ahead and start the fitting. In this case it converges with an error of 7% which is not as good as we would like. When comparing the model and the data we can see that the model has some discrepancies in its response to slip angle.



We will now use OptimumT's interactive coefficient adjuster. In the coefficient list we hold click and hold the left mouse button over the minus sign for the rBx1 coefficient. We can instantly see a dashed line in the graph which shows what the model would look like if rBx1 would decrease with 10%. We can preview the changes before me make them!

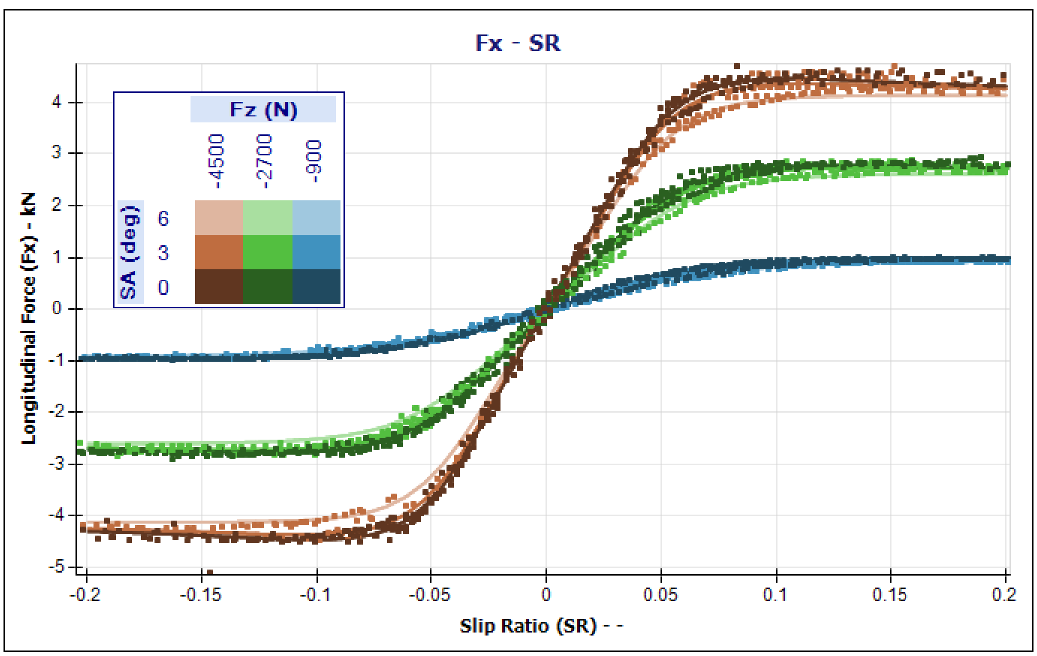




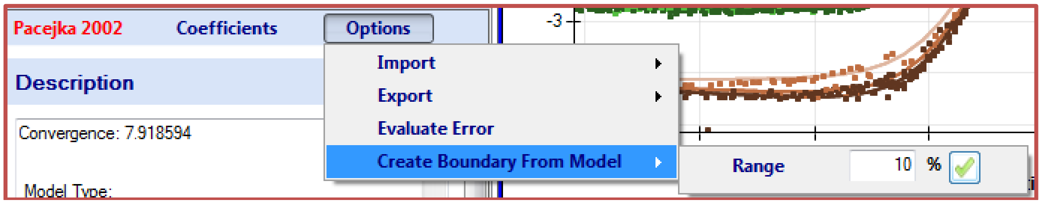
By double-clicking on the minus sign, the coefficient will change 10%. We will keep double-clicking until that the slip angle response matches the data better. In this case it seems like decreasing the rBx1 coefficient from 7.2 to 4.5 provides a better result.

Did you know:

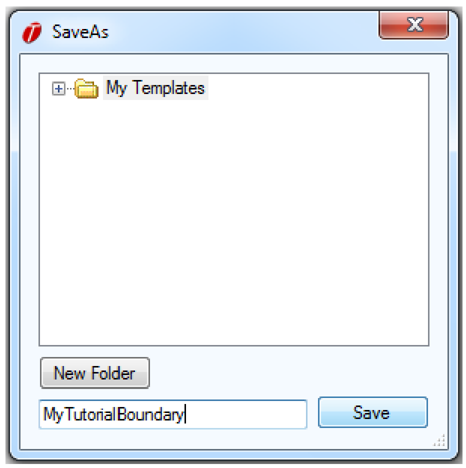
By using the interactive coefficient preview tool in OptimumT, you can quickly understand how different coefficients affect the model.



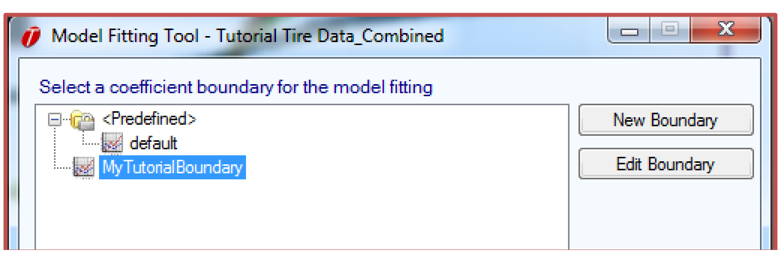
However, by manually adjusting the coefficients, we may have increased the error of the model. Luckily, we can use our newly modify model as a baseline for re-fit. By clicking on *Options* and then *Create Boundary From Model*, we can create a new coefficient boundary based on our current model. In this case we will enter 10%, which means the spread which we accept around our tweaked model.



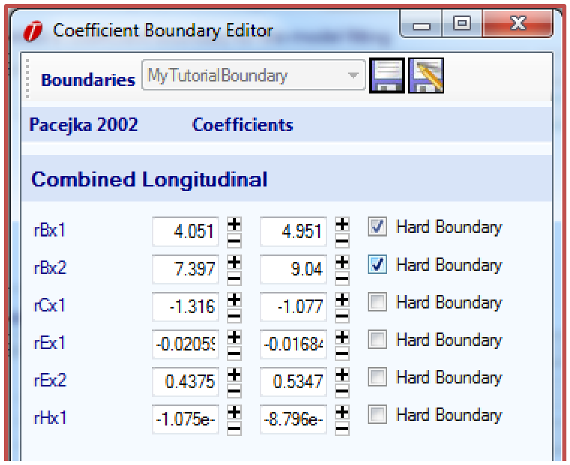
In the next window, we enter the name for our new Coefficient Boundary Model, we will call it *MyTutorialBoundary*.



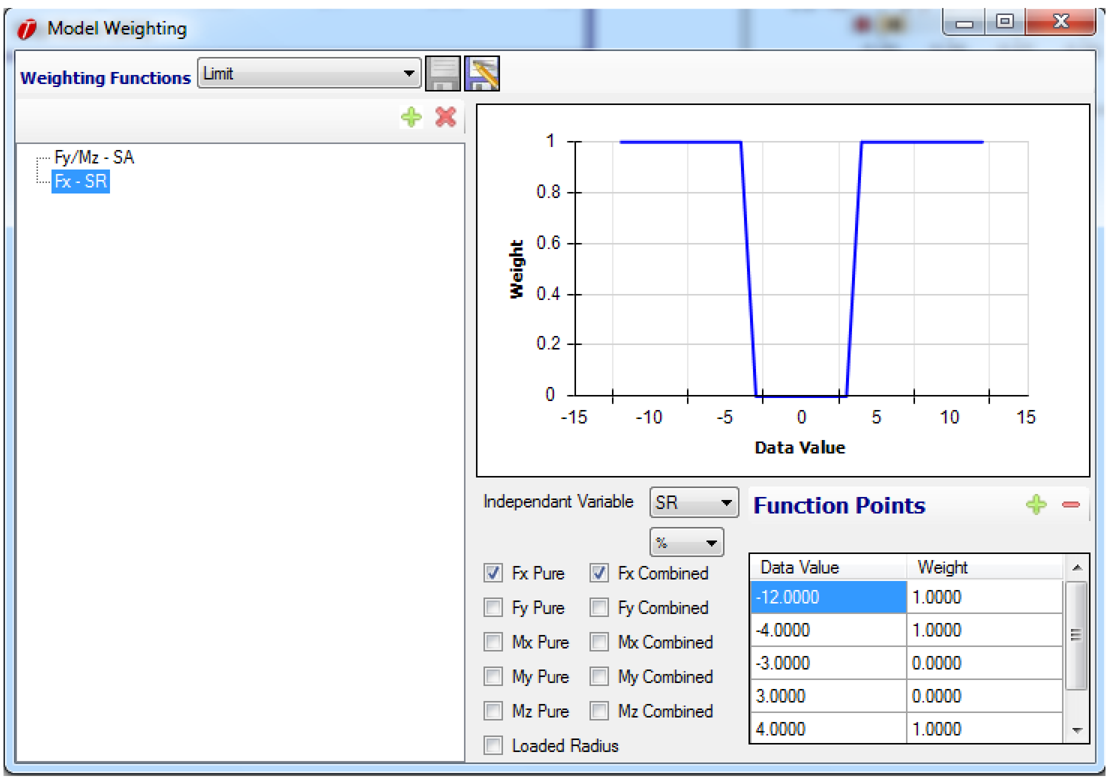
We can then repeat the fit as before, with the difference that we now select *MyTutorialBoundary* when we select the boundary model.



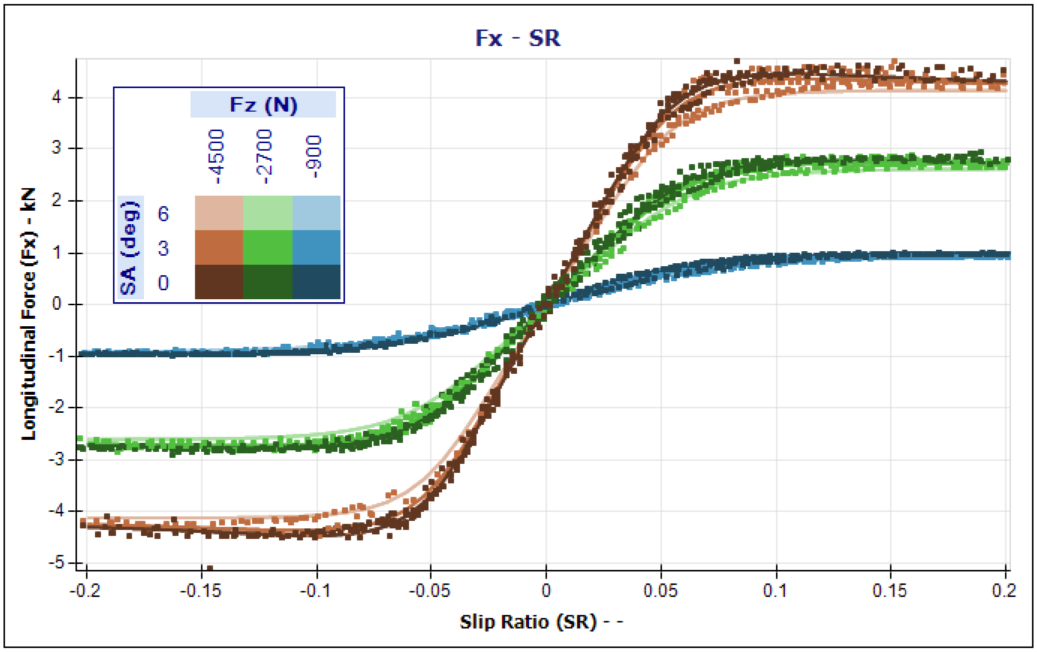
Before continuing we click *Edit Boundary* and click the checkboxes *Hard Boundary* which prevents the solver to use any coefficient values outside the ones specified.



Another option to increase the quality of the fit is to use a weighting factor to force the fitting algorithm to focus on getting a better fit at larger slip ratios. In this case we can use the predefined *Limit* weighting factor which will make the solver focus high slip angles/slip ratios. By careful though, this setting can sometime generate worse results!



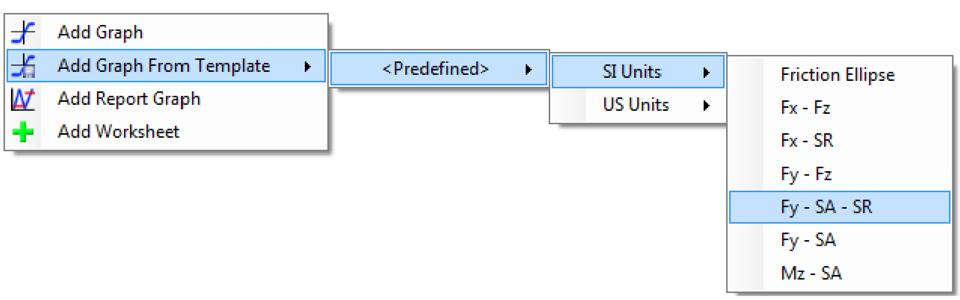
We use these setting to do another a new fit and we now end up with a slightly better fit. Of course, we could keep adjusting the coefficient boundaries and use the weight factors to try to attempt to improve this fit even more. Instead we will consider this to be sufficiently accurate for this tutorial.



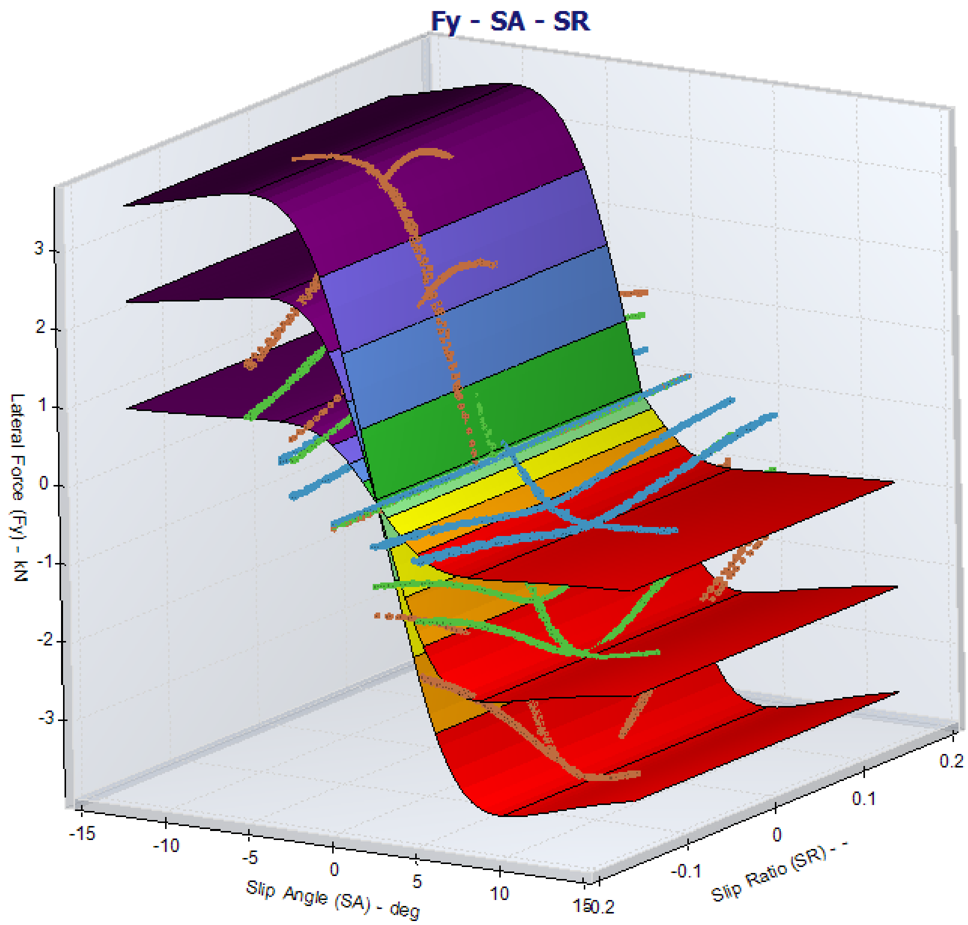
Our tire model now has the following areas fitted:

- Pure Fy
- Pure Mz
- Pure Fx
- Combined Fx

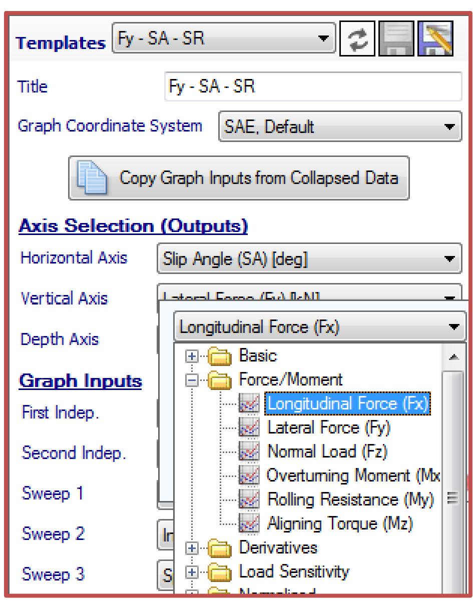
So let's use OptimumT's 3D visualizations to get a better idea of how our tire model currently looks like. We start by inserting a new graph, we will use on of the templates to get things started.

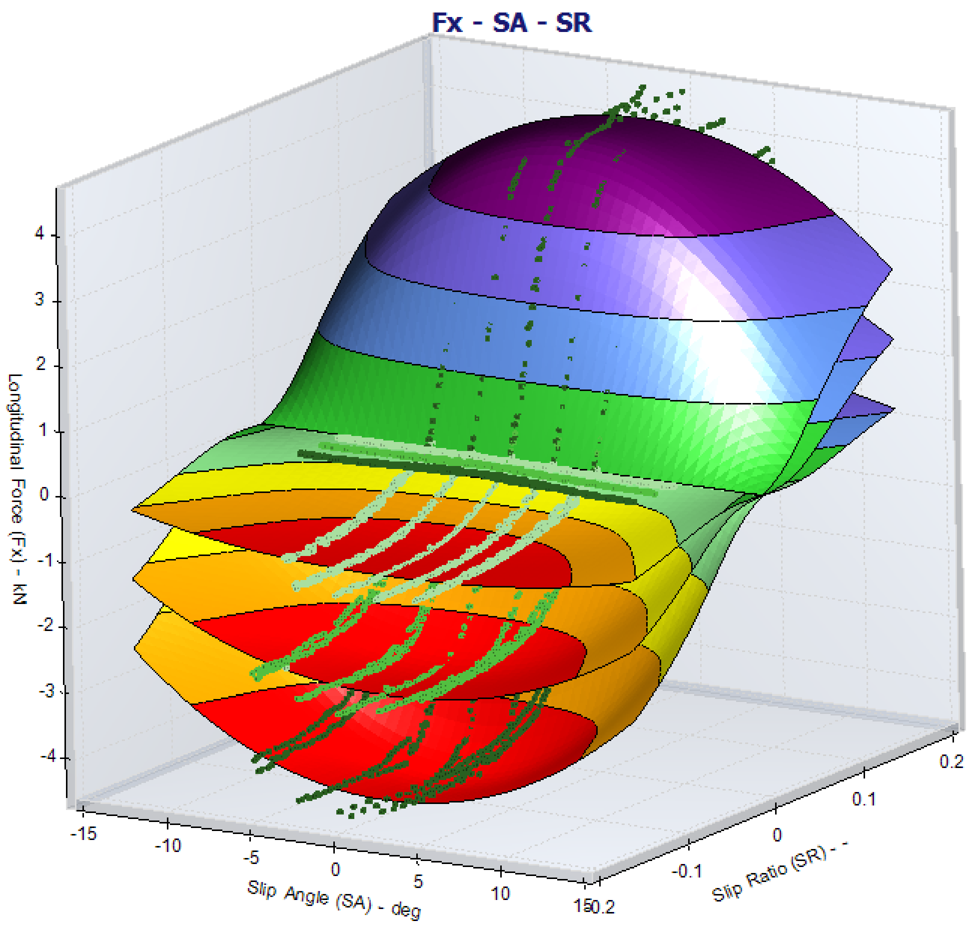


This 3D graph has Slip Angle and Slip Ratio on the X and Y axes respectively, it the shows lateral force on the Z axis. For our current tire model, we can see that the lateral force only depends on slip angle. This is due to the fact that we have not fit Combined Fy yet. We can also see how the surface (which is the tire model) is cutting through the data (which are the points) around SA = 0.



Now, let's put Fx on the Z axis instead. We can do this by changing the input selections in the graph options.



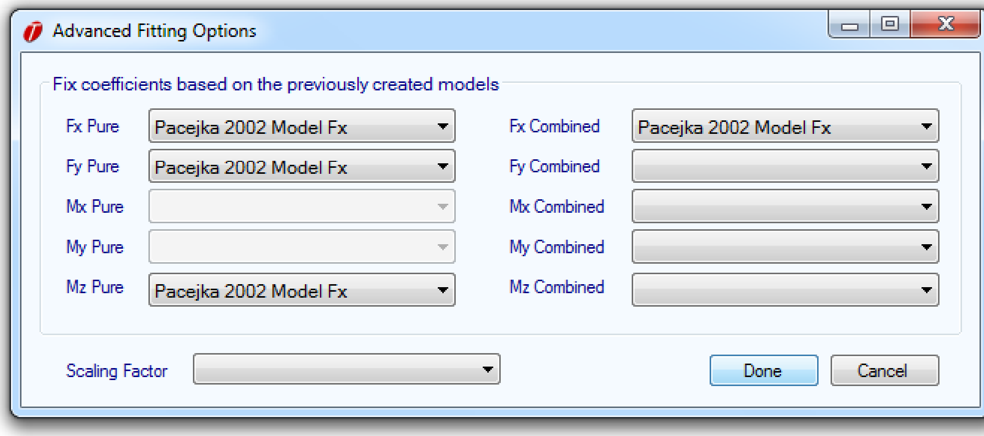


In this graph we can clearly see how longitudinal force decreases with increased slip angles, this response is what is modeled by the Combined Fx coefficients. In this graph we can also more clearly see how the model surface cuts through the data points. This can also be used as a qualitative method to evaluate the model quality.

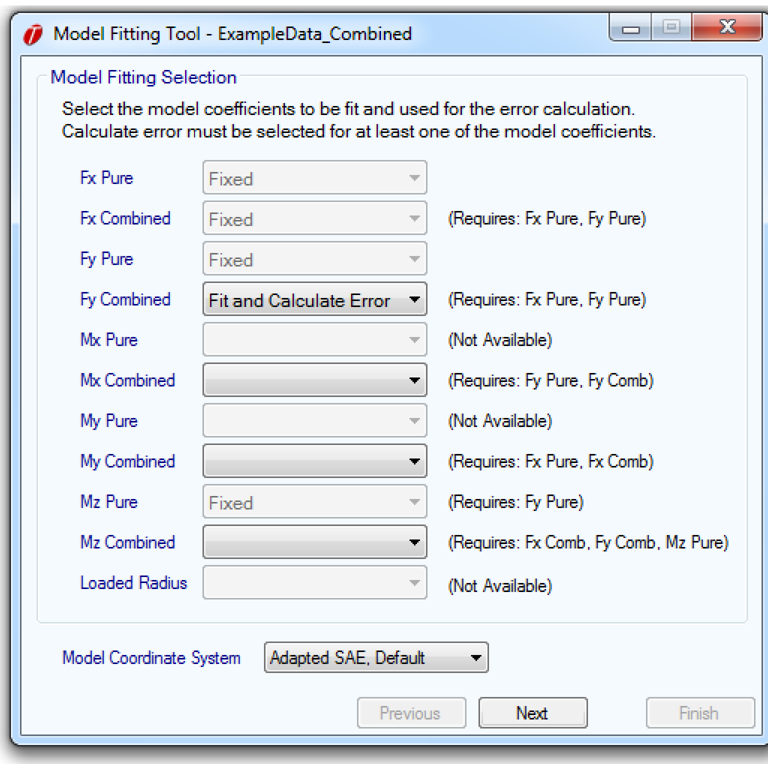
We can now move on the fitting the Combined Fy and Combined Mz part of the model.

4. Fitting Combined Fy

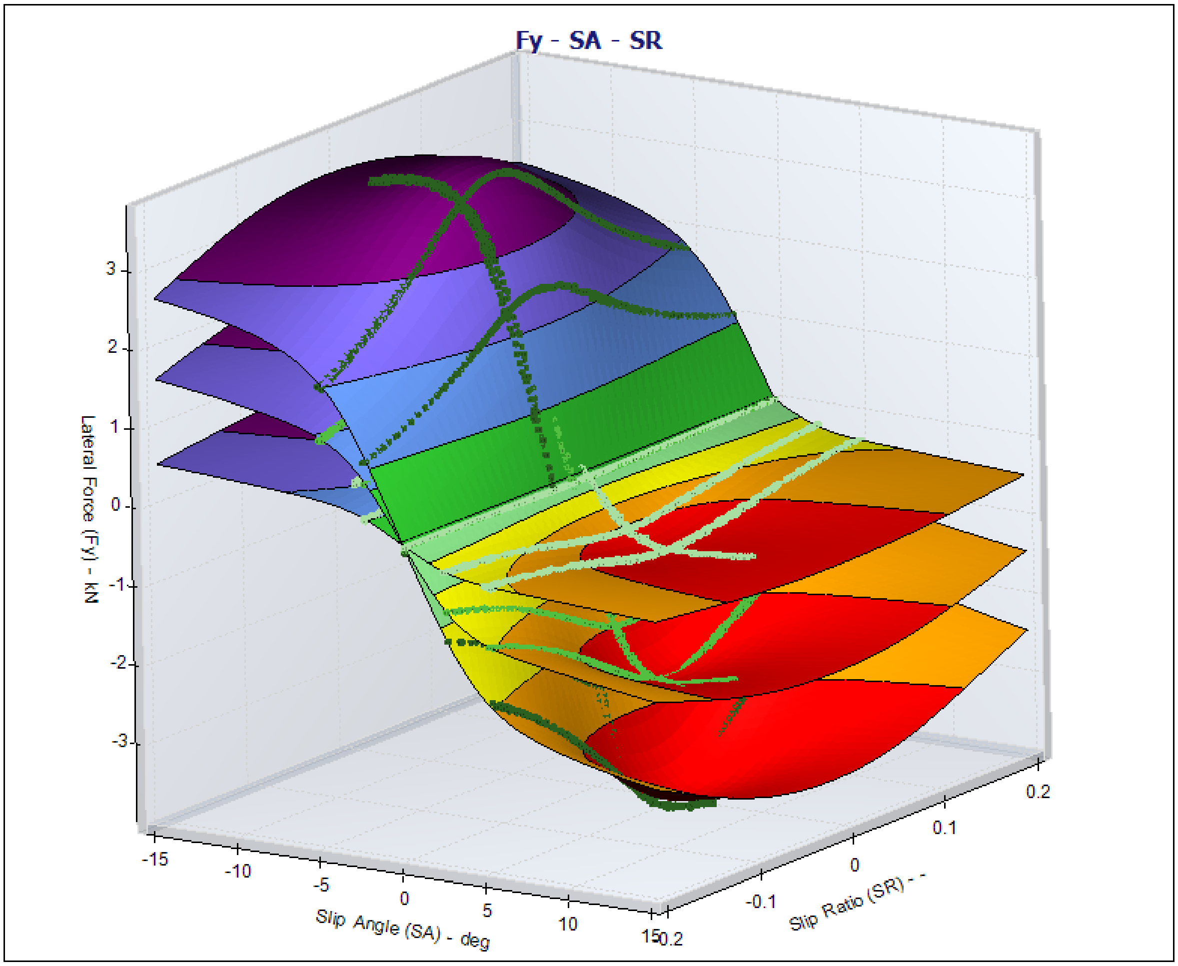
We are now ready to start fitting the combined Fy coefficients. Select the combined test data in the tree and scroll down and select a Pacejka 2002 and click *Fit Model*. The window *Advanced Fitting Options* will appear, in this window we can configure which previously fit models we want to base our new model on. In this case we want to base our new model on our combined Fx model (which also has Fy Pure, Mz Pure and Fx Pure coefficients in it). We select to fix coefficients from that model for the Fx Pure, Fy Pure, Mz Pure and Fx Combined parts. Click *Done* to continue.



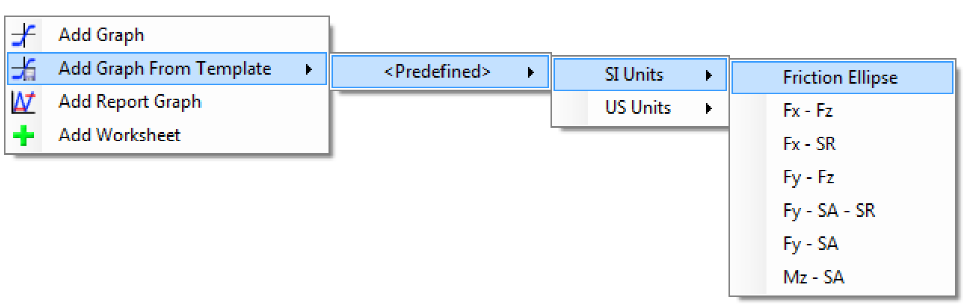
In the next window, we select to fit the Combined Fy coefficients. We can then click *Next* to continue.



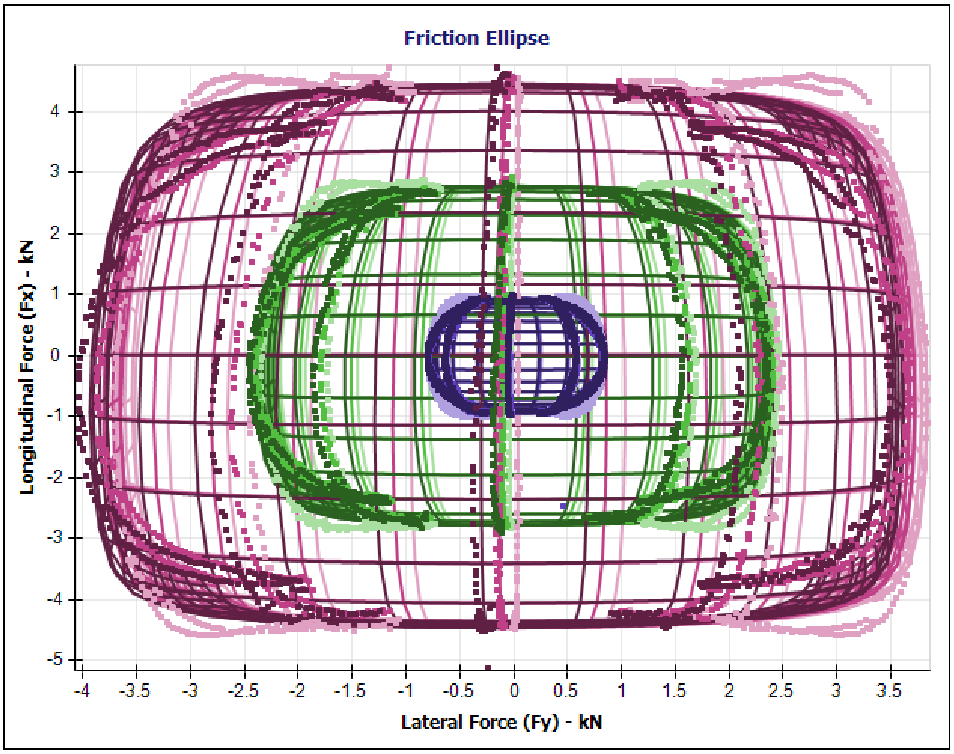
We enter the appropriate tire parameters. We also select to use the default coefficient boundaries and no weight factors to start with. We can then start the fitting process. The fitting finishes with around 7% error, so let's take a look at the model using our previously created 3D visualization.



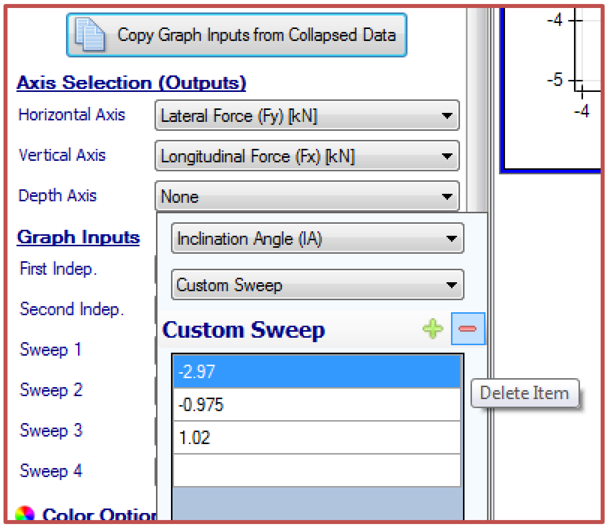
We can now see how the model surface curves of with increasing slip ratio (compare to our earlier model without combined Fy a couple of pages ago). We can once again see how the model surface cuts through the data points, however we see that there are some areas where there is a bigger difference between the model and the data. To investigate this error in greater detail, we are going to use the friction ellipse visualization. We insert it using one of the graph templates.



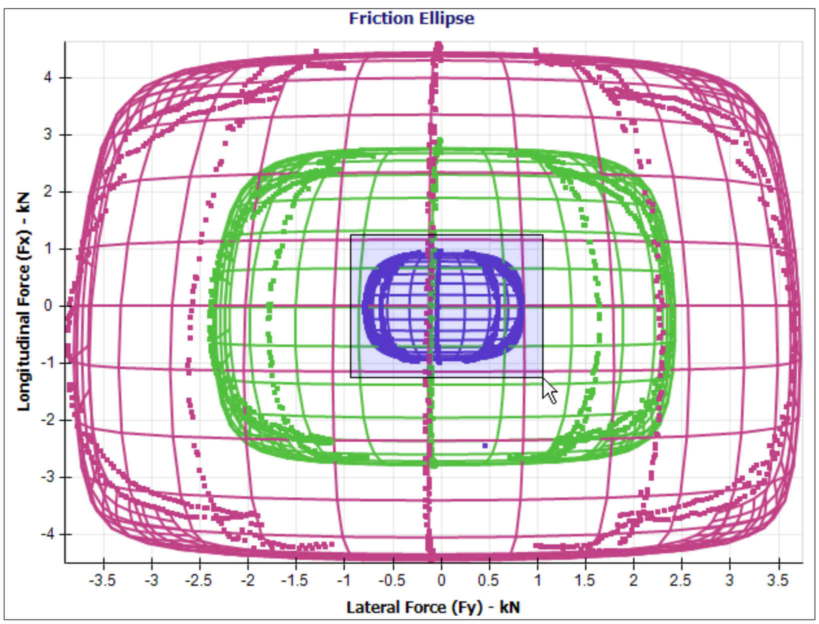
To quickly configure the graph so it matches our test data, we click *Copy Graph Inputs from Collapsed Data* and then to have the data points colored accordingly, we select *Select No Items to Plot All Data*. This procedure was covered in greater detail in Tutorial 2.



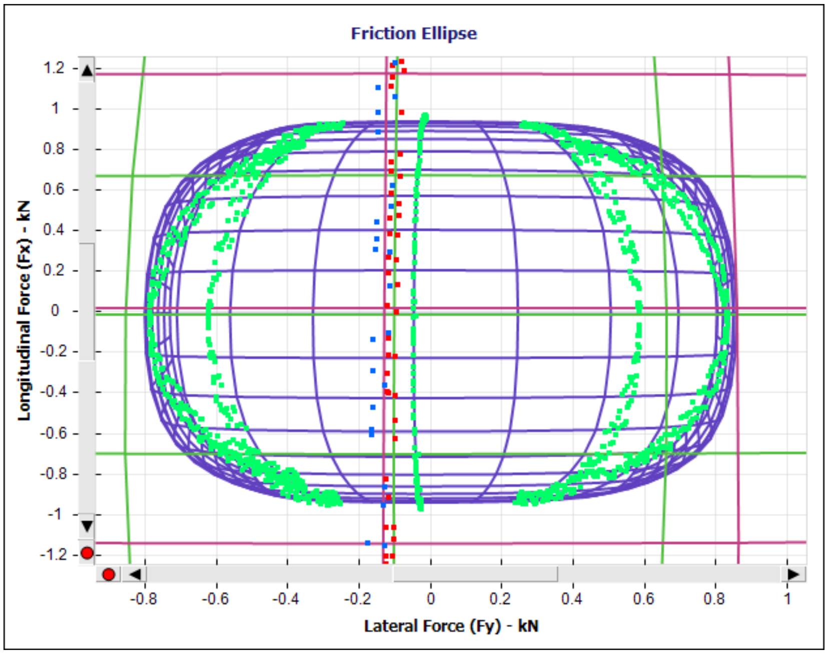
We now have our friction ellipse with both the model and the data. We can clearly see how the model rolls off in combined states (both lateral and longitudinal forces), but to make the graph a bit more clear, we will only look at one of the camber cases. We look at the graph properties on the left hand side of the screen and we edit the *Inclination Angle* sweep. We then remove two of the angles by selecting them and clicking the minus sign. If you wonder were these numbers (-2.97°, -0.975° and 1.02°) come from, they are the test cases that were identified automatically during the collapsing process.



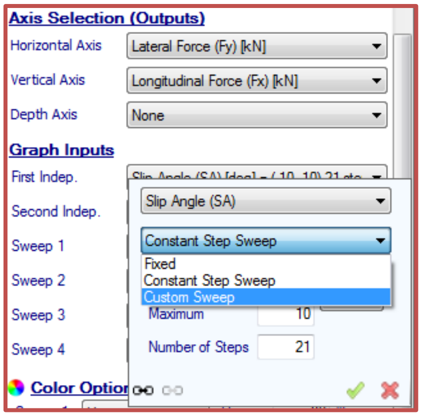
Returning to the graph, we can now see how it is less busy. It becomes apparent that for higher loads (larger friction circle), our model overestimates the forces compared to the data. In other words, our model produces a friction ellipse that is too square. We zoom in on the lowest load by holding down the scroll wheel and select that area.



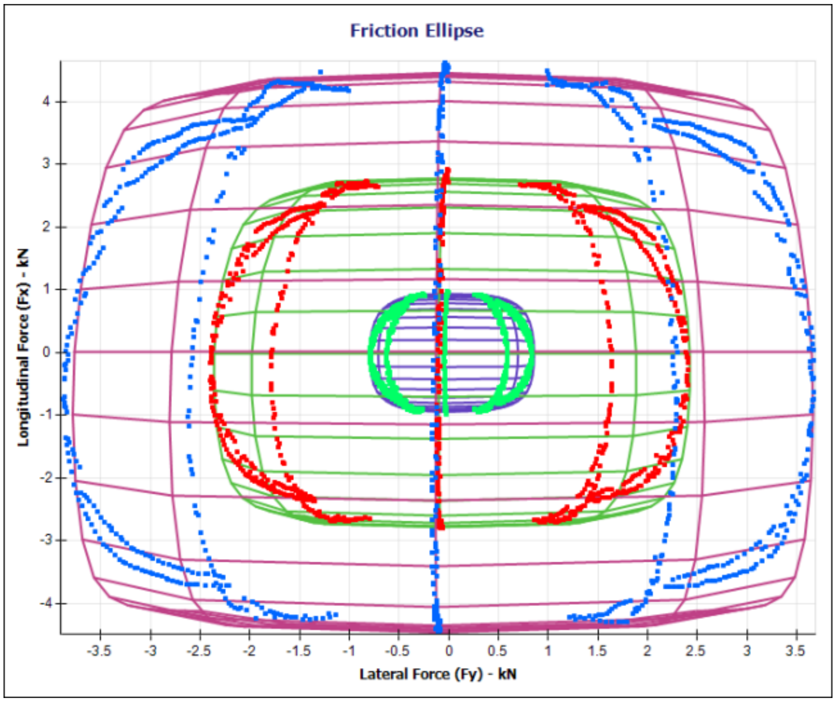
We can see how that we have the same problem at the lower loads, our model produces too square of a friction circle (you might want to change the color of the data to more clearly see differences between model and data). To tweak the model we will use the interactive coefficient adjusters that we used while fitting the Fx coefficients earlier in this tutorial.



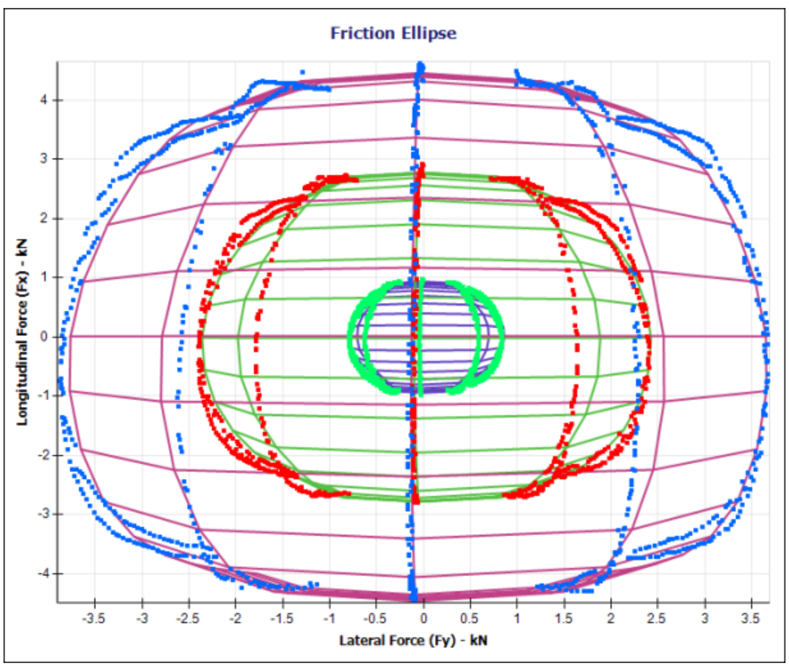
We will focus on the combined coefficients and by using the information in Chapter 8.8 of the OptimumT help file we can get a better idea on which ones to focus on. Before we start making any changes we just do a quick copy (using copy and paste) of our model, should we want to revert back to our original model later. But before we start adjusting, we want to do a last adjustment to the graph; we want to have our circle only to draw lines at the same slip angles as our data. We change the *Constant Step Sweep* to a *Custom Sweep*. The slip angles which we are interested should already be filled in (as we copied graph input from the collapsed data).



Let's now have a look at what our original model looks like. Our task now will be to try to match up the strings of data with the vertical curves of the model.

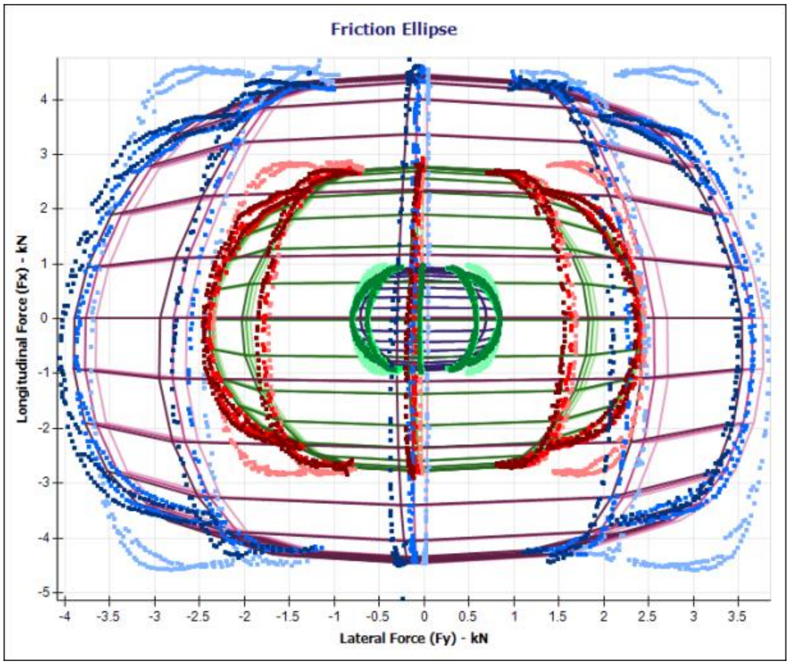


By just quickly playing with the combined Fy and Fx coefficients we after a minute or two, end up with something like this.



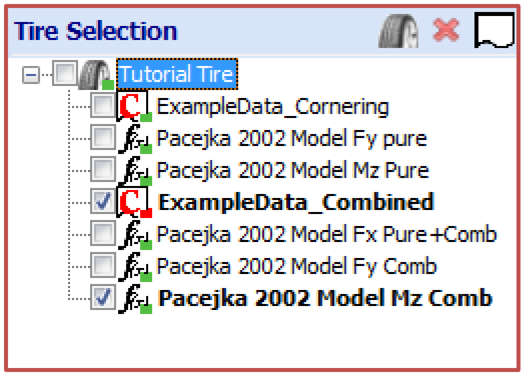
With the help of the interactive adjusters we have gotten a much better correlation at the higher slip angle (largest lateral force). But we still have some differences at the lower slip angle, especially near zero slip ratio. But that part of the model is mainly controlled by the Pure Fy coefficients, which we fit to the cornering data and not the displayed combined data. There are some ways around this which we will cover in Tutorial 5, namely scaling factors.

We need however be careful, because we are currently only looking at one camber angle. By once again copying the graph inputs from the collapsed data we can look at all camber angles.



We can see how one of the camber angles (the lightest color) is behaving quite differently. This is something that would be quite difficult to accurately model due to limitations in the Pacejka tire model. As often in tire model fitting, a compromise needs to be made. In this case, we can decide that we will ignore one of the camber angles. If we want to try to improve the quality of the fit, we can always try to do a re-fit, without using that camber angle. We can also try to create a new coefficient boundary, based on our modified model or try to use weighting factors. There obviously are many different ways to get a good fit.

The last thing to do to finish the model in this tutorial is to fit the combined Mz which is a pretty straightforward story. Armed with the tricks and tips from this tutorial, you should be more than ready to do the fit of the combined Mz. At the end, your project tree should look something like this:



In the next part of this tutorial series, we will take a look at some more tricks that can be used to improve our fits, namely scaling factors. We will also look at different ways of controlling the quality of our models. Furthermore, we will take a look at some of the other OptimumT features we haven't mentioned yet!