METHODS FOR OBTAINING IMPROVEMENTS TO THE STEADY-STATE, PURE SIDE-SLIP PACEJKA TIRE MODEL IN CAPTURING CONVENTIONAL AND UNCONVENTIONAL TIRE BEHAVIOR.

by

Jugal V. Popat

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Mechanical Engineering

Charlotte

2015

Approved by:

Dr. Peter T. Tkacik

Dr. Mesbah U. Uddin

Dr. Kent A. Day

ABSTRACT

JUGAL V. POPAT. Methods for obtaining improvements to the steady-state, pure side-slip Pacejka tire model in capturing conventional and unconventional tire behavior.
(Under the direction of DR. PETER T. TKACIK)

Modified codes have been developed to improve semi-empirical tire modeling using the 2002 and MF6.1.2(2012) versions of the steady-state, pure side-slip Pacejka model. The 2012 version, in its pure form, is not suitable to model unconventional tire characteristics. Analysis is performed by examining improvements to handle the effect of change in camber and pressure on lateral force for pure cornering tests. A split optimization method is developed to model the unconventional camber characteristics. Fitting using this method leads to an overall improvement in the quality of the fit. Additionally, a modification of the 2012 Pacejka model is proposed to enable it to model unconventional pressure effects. This further improves the quality of fit, for both, pressure and camber characteristics. Due to the difficulty in obtaining tire data, this model uses the multi-tire test results obtained from the Formula SAE Tire Testing Consortium (TTC). Similarly, the underlying code of this thesis is obtained from the TTC website. The code is dissected into preprocessing, fitting and post-processing and each section is analyzed in detail in order to obtain the best-fit coefficients. The work done in this research can be used as a platform to dive into advanced topics in empirical and semi-empirical tire modeling.

DEDICATION

To my parents, my advisor and my friends for being a great source of support throughout this journey.

ACKNOWLEDGEMENTS

I would like to begin by thanking my advisor, Dr. Peter T. Tkacik, for his constant guidance and motivation. Dr. Kent Day for providing a different perception of the problem at hand. Together their key inputs helped steer this research work towards its goal. I'm grateful to the people at the Calspan Tire Research and Testing Facility (TIRF) and the individuals heading the Formula SAE Tire Testing Consortium (TTC) for taking time out of their busy schedules to conduct tire testing. Especially, Mr. Doug Milliken, Dr. Edward Kasprzak and Mr. Brian Seater for maintaining the data from all rounds of testing, and for their invaluable inputs on the forums of the TTC website. Mr. Bill Cobb for sharing a wealth of information on the TTC forums. It is one of his codes that has enabled me to dive into empirical tire modeling and without which this research would not have been possible. I would also like to thank the Mechanical Engineering Department, College of Engineering and Mosaic Computing for providing access to MATLAB and other computing facilities. Lastly, I'm grateful to Mr. Kile Stinson and the UNCC Formula SAE team for providing initial access to the tire data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Tire Modeling

The tire is the only contact between the vehicle and the road. It serves as a transfer mechanism for forces between vehicle and the road. Understanding this transfer mechanism is of utmost importance from a vehicle dynamics point of view. The accuracy of a vehicle dynamics simulation model is largely governed by the underlying model used to predict the physics of tires. A tire model gives the force and moment value at any instant for given parameters such as vertical load, inclination angle and slip angle of the tire. The vehicle dynamics simulation uses this data as input for the multi-body dynamics calculations. Thus, an error in estimation of forces at the tire model level can translate to a greater error in vehicle dynamics prediction. This highlights not only the need for a tire model but the need for a highly accurate tire model. Tire modeling, hence, is a primary area of research within vehicle dynamics simulations.

1.2 Motivation

Tire modeling is usually undertaken by tire manufacturers, vehicle manufacturers and racing teams. The models can be used by tire manufacturers at the design stage to analyze effect of parameters such as coefficient of friction or pressure versus temperature variation which change with change of rubber compound. They can be used by vehicle manufacturers and racing teams to enhance vehicle ride, stability and handling simulations. The commercial vehicle manufacturers may use them to predict response in hazardous maneuvers and develop safety systems for chassis stability. Racing teams use simulations to predict and extract maximum performance from the

vehicle. Although the motivations are different, the importance of an accurate tire model is not diminished. These organizations may spend millions of dollars in an attempt to outdo their counterparts. They are not too keen on revealing their work for fear of it being picked up by their competitors. The biggest cost is losing their possible advantage or the competition gaining an advantage by applying the modeling philosophy in a better manner. Hence, it is a secretive subject.

The aim of this thesis is to provide a starting point for anyone interested in tire modeling and methods to enhance them. An attempt has been made to introduce the basics of empirical tire modeling. Specifically, using the Pacejka model, also known in the industry as the "*Magic Formula*". The work presented ahead provides an insight into factors affecting modeling using the steady-state, pure side-slip Pacejka model and presents a comparison between the 2002 and 2012 versions. Modification of the process of modeling to handle unconventional tire effects with camber is analyzed. A modified version of the 2012 Pacejka model is proposed that enables it to model unconventional pressure effects on the tire. This research will serve as a good platform to dive into advanced topics in empirical and semi-empirical tire modeling.

1.3 Organization of Thesis

Chapter 1 indicated the motivation for this thesis.

Chapter 2 presents the different modeling philosophies and dives further into empirical modeling using the Pacejka model. A background of tire testing is also provided.

Chapter 3 includes description of the under-lying MATLAB code used for this research.

Chapter 4 deals with individual aspects of the code and their analysis. It builds up to the modifications required in the original code to implement the 2012 version of the Pacejka model and enable it to capture unconventional camber and pressure effects.

Chapter 5 enlists closing arguments and future scope of this work.

CHAPTER 2: BACKGROUND

2.1 Introduction

On the basis of modeling philosophy, tire models can be classified into three major types: empirical, physical (analytical) and semi-empirical. Each of these can be further classified into steady-state and dynamic/transient models based on the time response they are able to capture. Their descriptions follow in subsequent sections under Section 2.2. The definition, history and development of Pacejka's *Magic Formula* is covered under Section 2.3. Description of tire testing procedure to obtain force and moment data is included under Section 2.4.

2.2 Tire Modeling Philosophies

2.2.1 Physical Modeling

The underlying philosophy of this approach is that the behavior of a tire can be partially or completely defined by establishing an analogy with simpler physical phenomena. There have been briefly three analogies used for this type of modeling. Accordingly, they are named as Brush model, String model and Beam model [1]. With increased computational capability, these models have developed into highly complex implementations over the years. At the top of the complexity chain are finite-element models which not only simulate the effects of friction and distortion of the rubber compound but also account for heat transfer effects. A brush model at different slip conditions is shown in Figure 2.1.

The specific properties of interest are tread deflection, carcass/belt deflection, distribution of contact pressure and tire-road friction properties [1]. Simple physical models generally define mathematical models for only one or two of these proper-

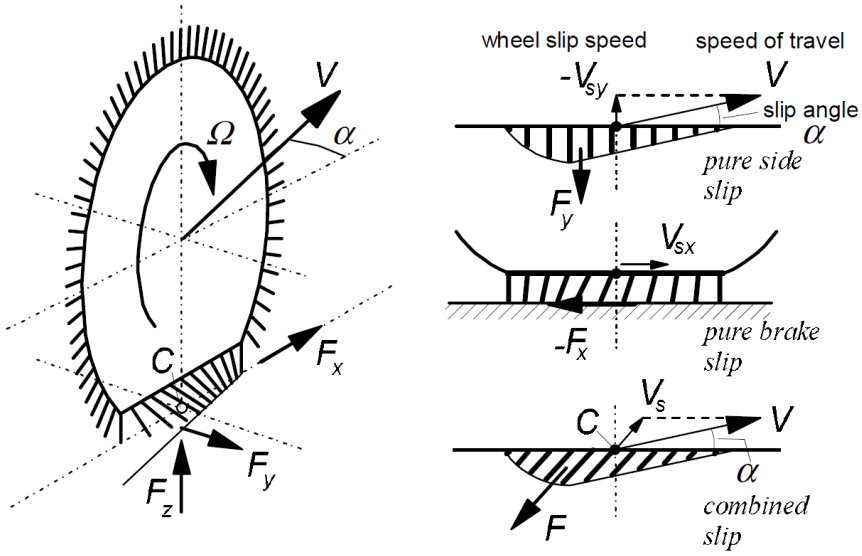


Figure 2.1: Tire brush model at different slip conditions [2].

ties and make assumptions for the rest. Complex physical models attempt to define mathematical models for all the properties. The decision for extent of modeling the aforementioned properties depends upon specific application. Figure 2.2 helps compare the qualitative accuracy of the different models. The curves a and b are obtained for models which define the carcass as a stretched string and an elastic beam respectively. The curves c and d are brush models where the tread elements are modeled as a brush. The curve c assumes a symmetrical parabolic distribution for carcass deflection [3] whereas curve d assumes no carcass deflection. All of them assume a non-symmetrical pressure distribution. It is thus seen that two different methods of physical modeling can yield similar results. The choice becomes a question of required specific end result. The ability of a physical model to capture combined slip and turn slip effects leads to further complex set of equations.

2.2.2 Empirical Modeling

Empirical models are derived using a purely mathematical approach of curve fitting and interpolation on a given set of tire data. The underlying functions for curve fitting can range from simple linear functions to complex non-linear function definitions. Tire

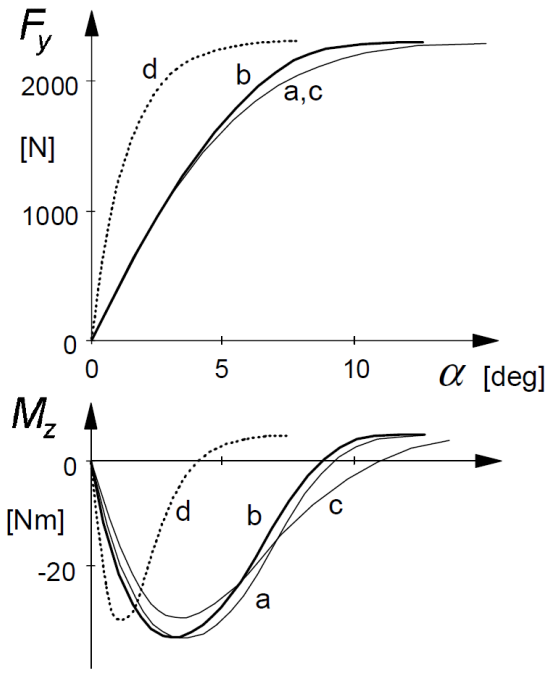


Figure 2.2: Comparison plot of different physical models for Lateral Force (Fy) and self-aligning moment (Mz). a) String b) Beam c)Brush with carcass compliance d) Brush with rigid carcass [2].

data in this context means the force and moment data obtained by subjecting the tire to different operating conditions on a test machine. More about tire testing and data is covered in Section 2.4. These models are mainly chosen because of their ability to capture all tire behavior in a short amount of time. The increased accuracy and less computation time make it best suited for vehicle dynamics simulations [2]. Even simple physical models are unable to solve the equations quickly enough to be used in a vehicle model. However, empirical models may exhibit large deviations when applied to different conditions than those that they were developed on.

An example of an empirical model is the similarity method by normalization [2, 3, 4]. Here, similarity is defined as obtaining the same characteristic curve for different operating conditions of the tire by applying horizontal and vertical shifts. The characteristic curve is obtained by testing the tire in pure slip conditions (side-slip or longitudinal slip) at nominal values of vertical load and road friction and in

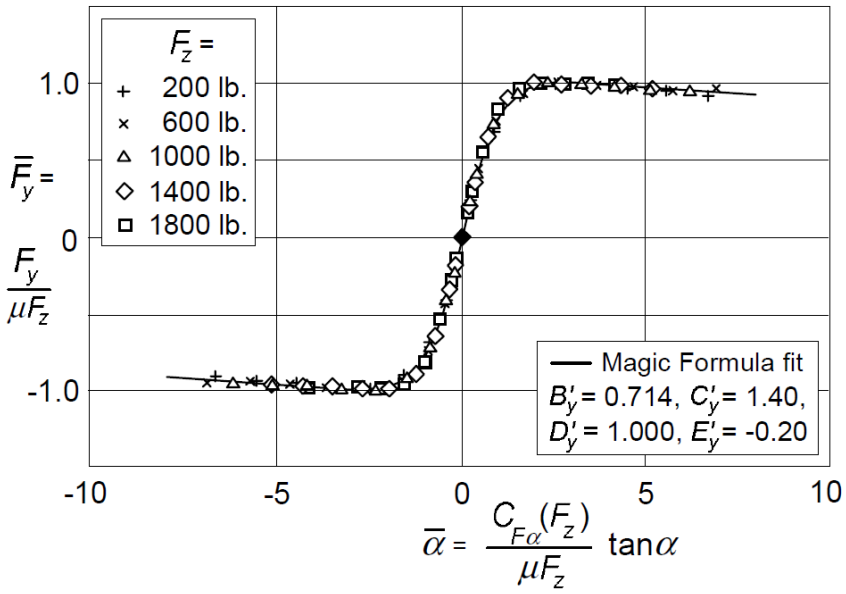


Figure 2.3: Non-dimensional lateral load ($\bar{F}y$) vs. non-dimensional slip angle ($\bar{\alpha}$) for multiple normal loads (Fz) [2].

the absence of camber. Figure 2.3 shows the curves obtained for normalized values of lateral force and slip angle at different vertical loads. The normalization is done to account for change in vertical load and hence, includes the term Fz in the denominator. The coefficients B_y', C_y', D_y', E_y' belong to a non-dimensional version of the *Magic Formula* for modeling lateral force. Description of the *Magic Formula* can be found in section 2.3. More about the process of normalization and non-dimensional version of the *Magic Formula* can be found in [5].

2.2.3 Semi-empirical Modeling

As the name suggests, semi-empirical models use a combination of physical and empirical modeling philosophies. The models can either have one specific property being modeled using physical equations or can have empirical relationships modified on the basis of phenomena observed from physical models. An advantage of using them over empirical models is that they are accurate even when applied to extended data sets. They still require less computation capability than simple physical models since the underlying definition is purely mathematical. The *Magic Formula* model

(also known as the Pacejka model, after its creator Dr. Hans B. Pacejka) is the most popular semi-empirical tire model. It is described in the next section.

2.3 The Magic Formula

In its simplest form, the sine version of the *Magic Formula* is as shown in Equation 2.1 where Y is the output characteristic obtained for input x to the tire. The coefficients appearing in the equation are responsible for individual aspects of the output curve. B is called the stiffness factor because it modifies the slope of the curve at the origin. C controls the extent of the sine function and hence, is known as the shape factor. D sets the peak value of the curve on the vertical axis and E controls the curvature and horizontal position where the curve reaches its peak. To be able to shift the curve from the origin, the horizontal and vertical shift factors S_H and S_V are applied. It can be summarized in Figure 2.4.

$$\begin{aligned} y &= D\sin[C\arctan\{Bx - E(Bx - arctan(Bx))\}] \\ Y(X) &= y(x) + S_V \\ x &= X + S_H \end{aligned} \tag{2.1}$$

$$\begin{aligned} y &= D\cos[C\arctan\{Bx - E(Bx - arctan(Bx))\}] \\ x &= X + S_H \end{aligned} \tag{2.2}$$

$$G = D\cos[C\arctan(Bx)] \tag{2.3}$$

The sine version of the *Magic Formula* is suitable to model force characteristics only for conditions of pure slip. To model moment characteristics, a cosine version is used as shown in Equation 2.2. For combined slip operation, a weighting cosine function was proposed in [6]. Equation 2.3 depicts the function.

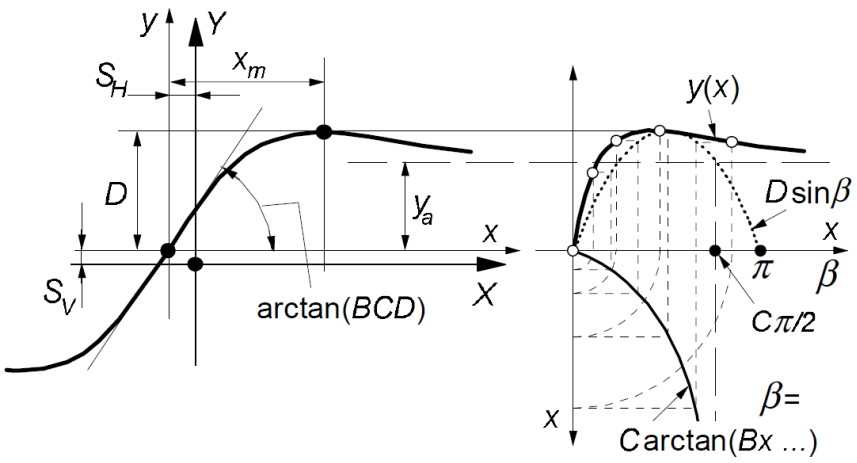


Figure 2.4: Contribution of coefficients appearing in the *Magic Formula* to the output curve [2].

Purely empirical and semi-empirical models were developed based on this formula. Most of the existing models are a result of the work done by Dr. Hans B. Pacejka and we will refer to them as Pacejka models in this research. Using the *Magic Formula* relationship, equations are defined for the individual coefficients based on phenomena observed in physical modeling. Thus, the coefficient equations include parameters which change in value as the input varies. These parameters will now be referred to as coefficients of the Pacejka model. Different versions of the Pacejka model exist based on differences in the definition of equations and ability to model additional inputs. The models used in this thesis are the 2002 model and the 2012 (MF6.1.2) model. The corresponding equations are obtained from [7] and [2] respectively. The 2009 (MF5.2) version was predefined in the code obtained for this research and hence, will also be used for analysis. We will only look at the equations for modeling lateral force under pure side-slip operating conditions (suffix $y0$ and y are used). The corresponding 2002 model equations are shown below.

$$F_{y0} = D_y \sin[C_y \arctan\{B_y\alpha_y - E_y(B_y\alpha_y - arctan(B_y\alpha_y))\}] + S_{Vy} \tag{2.4}$$

$$\alpha_y = \alpha^* + S_{Hy} \tag{2.5}$$

$$C_y = p_{Cy1}.\lambda_{Cy} \tag{2.6}$$

$$D_y = \mu_y.F_z.\zeta_2 \tag{2.7}$$

$$\mu_y = (p_{Dy1} + p_{Dy2}df_z).(1 - p_{Dy3}\gamma^{*2}).\lambda_{\mu y}/(1 + \lambda_{\mu V}V_s/V_0) \tag{2.8}$$

$$E_y = (p_{Ey1} + p_{Ey2}df_z).\{1 - (p_{Ey3} + p_{Ey4}\gamma^*)sgn(\alpha_y)\}.\lambda_{Ey} \tag{2.9}$$

$$K_{y\alpha 0} = p_{Ky1}F'_{z0}sin[2arctan\{Fz/(p_{Ky2}F'_{z0})\}].\lambda_{Ky\alpha} \tag{2.10}$$

$$K_{y\alpha} = K_{y\alpha 0}.(1 - p_{Ky3}\gamma^{*2}).\zeta_3 \tag{2.11}$$

$$B_y = K_{y\alpha}/(C_yD_y + \epsilon_y) \tag{2.12}$$

$$S_{Hy} = (p_{Hy1} + p_{Hy2}df_z).\lambda_{Hy} + p_{Hy3}\gamma^*.\lambda_{Ky\gamma}.\zeta_0 + \zeta_4 - 1 \tag{2.13}$$

$$S_{Vy} = F_z.\{(p_{Vy1} + p_{Vy2}df_z).\lambda_{Vy} + (p_{Vy3} + p_{Vy4}df_Z)\gamma^*.\lambda_{Ky\gamma}\}.\lambda'_{\mu y}.\zeta_2 \tag{2.14}$$

$$K_{y\gamma 0} = \{p_{Hy3}K_{y\alpha 0} + F_z(p_{Vy3} + p_{Vy4}df_z)\}\lambda_{Ky\gamma} \tag{2.15}$$

As seen from the equations, the basic $MagicFormula$ is extended to increase its capability of modeling variation in inputs. Here, the inputs are side-slip angle (α), normal load (F_z) and change in normal load from a nominal value (DF_z) and the inclination angle or camber angle (γ). The coefficient definitions are based on the phenomena observed from highly accurate physical models and from experimental data. Hence, they are semi-empirical models. Slip angle appears as a shifted slip angle α_y to account for horizontal shift of the output curve due to normal load and camber. The term α^* in Equation 2.5 is equal to tangent of the input slip angle α. The term γ^* appearing in the equations is introduced to account for effect of spin due to camber and is given by sine of the input camber γ. Camber affects peak lateral friction coefficient (μ_y), cornering stiffness with side slip ($K_{y\alpha}$) and horizontal shift

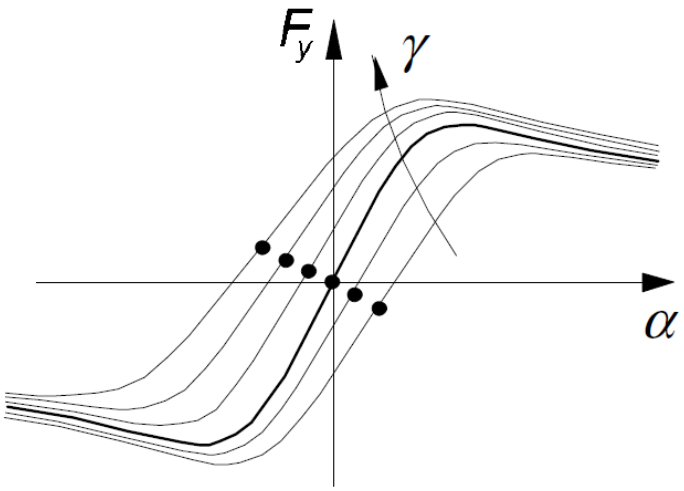


Figure 2.5: Influence of camber on lateral force curve [2].

of the curve (S_{Hy}). The variation of cornering stiffness with camber is defined by assuming that cornering stiffness decreases with increasing γ. It is assumed that the curve shifts horizontally towards greater lateral force at zero slip angle. Additionally, the curvature at high values of slip angle can be different for positive and negative slip angles. The coefficient E_y is accordingly defined to capture this phenomenon. Expected output with increasing γ is shown in Figure 2.5.

The 2009 model equations differ insignificantly from the 2002 equations whereas the 2012 model equations have additional coefficients. The equations of interest are for lateral friction coefficient (μ_y), cornering stiffness with side slip ($K_{y\alpha}$), cornering stiffness with camber ($K_{y\gamma 0}$) and horizontal shift of slip angle (S_{Hy}). They are shown in Equations 2.16, 2.17, 2.19 and 2.20 respectively. The main difference is the ability of the 2012 version to model effect of change in inflation pressure on tire output characteristics.

$$\mu_y = (p_{Dy1} + p_{Dy2}df_z).(1 + p_{py3}dp_i + p_{py4}dp_i^2).(1 - p_{Dy3}\gamma^{*2}).\lambda_{\mu y}/(1 + \lambda_{\mu V}V_s/V_0) \tag{2.16}$$

$$K_{y\alpha} = p_{Ky1}F'_{z0}(1 + p_{py1}dp_i)(1 - p_{Ky3}|\gamma^*|) \tag{2.17}$$

$$. \sin\left[p_{Ky4}\arctan\left\{\frac{F_z/F'_{z0}}{(p_{Ky2} + p_{Ky5}\gamma^{*2})(1 + p_{py2}dp_i)}\right\}\right].\zeta_3\lambda_{Ky\alpha} \tag{2.18}$$

$$K_{y\gamma 0} = F_z(p_{Ky6} + p_{Ky7}df_z)(1 + p_{py5}dp_i)\lambda_{Ky\gamma} \tag{2.19}$$

$$S_{Hy} = (p_{Hy1} + p_{Hy2}df_z).\lambda_{Hy} + \frac{K_{y\gamma 0}\gamma^* - S_{Vy\gamma}}{K_{y\alpha} + \epsilon_K}\zeta_0 + \zeta_4 - 1 \tag{2.20}$$

The additional input of inflation pressure is defined in the 2012 Pacejka model equations using change in pressure from nominal value (dp_i). The pressure coefficients are *PPY1, PPY2, PPY3, PPY4, PPY5*. The assumptions for modifying the equations for pressure can be found in [8]. Besselink et al. defined the pressure coefficients on the basis of data obtained from experiments and a modified brush model. The analytical model is known as a 'Treadsim' model and is defined in [7]. It was found that the effect of pressure on peak lateral friction coefficient was tire dependent. To deal with this aspect, a quadratic polynomial was used as can be seen in Equation 2.16. However, the effect of varying pressure on cornering stiffness was assumed to be tire independent. Based on experimental data and the *Treadsim* model, at low loads, an increase in pressure caused a decrease in cornering stiffness. This was attributed to the increase in vertical stiffness of the tire resulting in a reduction in contact length. As per the analytical model, cornering stiffness varies directly with square of half the contact length. Thus, a decrease in cornering stiffness was observed. For high loads, the reverse is obtained. Increased pressure led to a greater value of cornering stiffness. This was attributed to the increase in carcass stiffness which dominates increase in vertical stiffness effects at high loads. This phenomenon was assumed to be true for

all tires based on the observed experimental data. Tire independence justified the use of a linear polynomial as can be seen in Equations 2.17 and 2.19. The horizontal shift definition was probably modified to capture inter-dependence effects of camber and pressure. The research done ahead aims to analyze the model performance for a tire that shows different behavior with respect to camber and pressure than that assumed in defining the model equations. Although there are some commercial codes available that may be able to handle these effects, this research will provide an insight on how the fit can be improved at a model level.

Apart from the main coefficients of the Pacejka model equations, there are two types of additional factors to increase its adaptability. The λ terms are user scaling factors to scale the output curves for different operating conditions of the tire and are set a default value of one. The ζ terms are used to account for turn slip and camber (spin) and are important for large camber angles such as for motorcycle tires. They are set to one for small camber angles. The value of the coefficients appearing in the model equations are determined by applying a suitable regression technique to fit the curves obtained from the equations to the tire data. Research is also done to improve the ability of the regression technique by analyzing its sensitivity to starting coefficients, number of iterations and repeatability of results. Tire force and moment data is obtained by running it on a tire testing machine. Specifics regarding tire testing for the data used in this research are covered in the next section.

2.4 Tire Testing

The Formula SAE (FSAE) *Tire Testing Consortium* (TTC) is led by Dr. Edward Kasprzak, Mr. Doug Milliken and Dr. Bob Woods. Since 2004, the TTC has organized tire testing at the *Calspan* Tire Research Facility (TIRF) in Buffalo, New York. The test machine is a flat-track tire testing machine built in-house. A tire under testing is shown in Figure 2.6. As of May 2015, the normal load capacity of the machine is 12000 lbs whereas the lateral and longitudinal forces upto 9000 lbs can be measured.

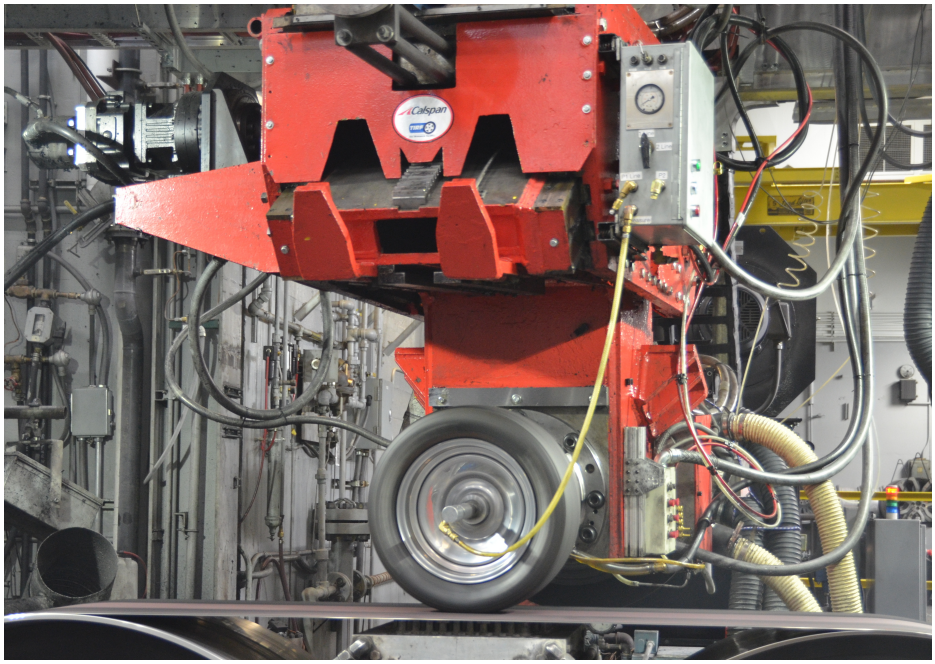


Figure 2.6: A tire under testing at *Calspan* TIRF.

The slip angle and inclination range is $\pm 25^{\circ}$ with an ability to run 10 inch diameter rims. Thus, FSAE tires can be tested on this machine but can lead to issues such as load control fluctuations and tire debeading at low pressures. The data regarding sensor resolution and accuracy is unavailable.

The data is stored in the form of input and output channels shown in Table 2.1 and can be imported in the code in excel or MATLAB (.m) format. The sign convention used is the SAE sign convention shown in Figure 2.7. According to it, positive slip angle is associated with a left hand turn. Consequently, a positive slip angle will produce a negative lateral force and a negative self-aligning moment as will be seen in the plots in subsequent chapters. Depending on the test protocol, a single raw data file can have more than sixty thousand data points.

The test protocol is decided by the TTC based on their experience. It can be broadly classified into pure cornering tests and drive/brake tests with increasing side slip. The pure cornering test is performed at free rolling conditions of the tire (longitudinal slip = 0) for multiple normal loads, inclinations and pressures. The drive/brake

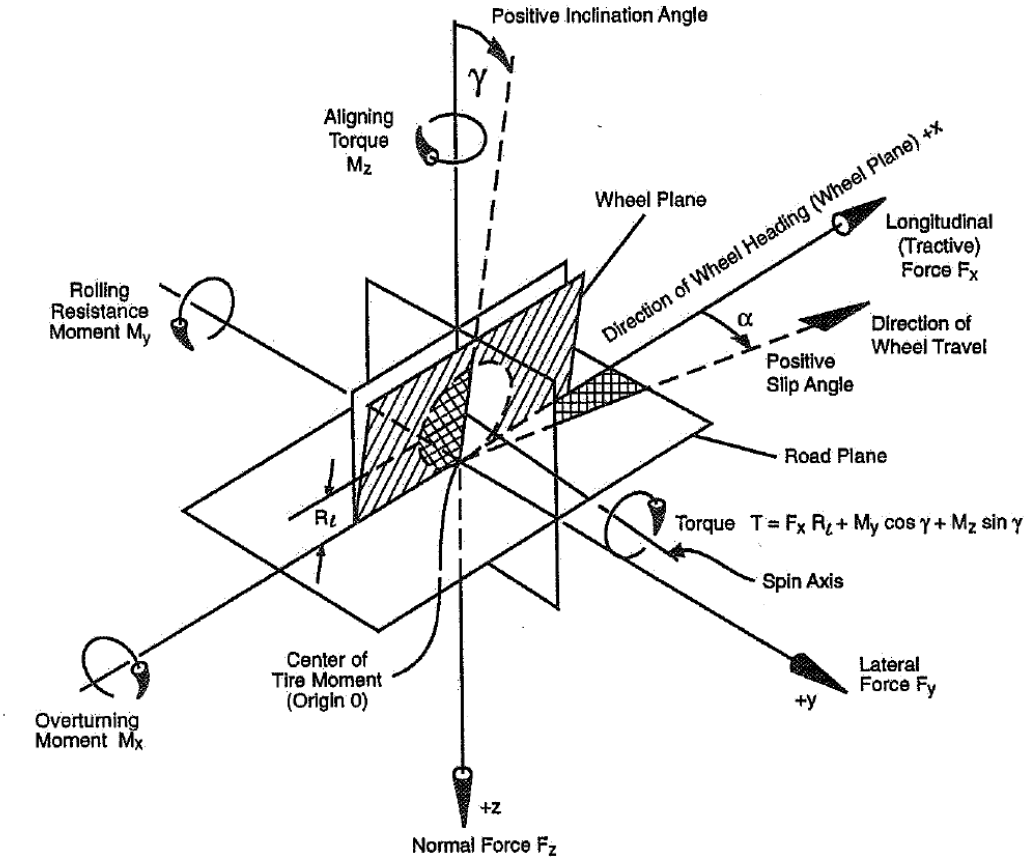


Figure 2.7: SAE sign convention used by *Calspan* TIRF [4].

tests include slip ratio sweeps while keeping the slip angle at 0° and then at two other values of slip angle. The zero slip angle condition data is used for pure longitudinal slip conditions and the data at remaining slip angles can be used for combined slip analysis. For each slip angle, multiple normal loads, inclinations and pressures are applied as input. Before recording the force and moment data, tire warm-up and break-in runs are performed to help it attain steady-state operating conditions. At the end of the runs, the tire is run at high inclination to clean the belt of tire debris. Details about the warm-up and the entire test protocol can be found in "Summary Tables" or "Run Guides" accompanying the raw data files. The next chapter provides a summary for an example code used to fit a MF5.2 Pacejka model to the raw data obtained from testing.

Table 2.1: Input and Output channels used to store the raw data. Obtained from round 6 of the TTC.

Channel	Units	Description
AMBTMP	degC or degF	Ambient room temperature
ET	sec	Elapsed time for the test
FX	N or lb	Longitudinal Force
FY	N or lb	Lateral Force
FZ	N or lb	Normal Load
IA	deg	Inclination Angle
MX	N-m or lb-ft	Overturning Moment
MZ	N-m or lb-ft	Aligning Torque
N	rpm	Wheel rotational speed
NFX	unitless	Normalized longitudinal force (FX/FZ)
NFY	unitless	Normalized lateral force (FY/FZ)
P	kPa or psi	Tire Pressure
RE	cm or in	Effective Radius
RL	cm or in	Loaded Radius
RST	degC or degF	Road surface temperature
SA	deg	Slip Angle
SL	unitless	Slip Ratio based on RE
SR	unitless	Slip Ratio based on RL
TSTC	degC or degF	Tire surface temperature - Center
TSTI	degC or degF	Tire surface temperature - Inboard
TSTO	degC or degF	Tire surface temperature - Outboard
V	kph or mph	Road Speed

CHAPTER 3: MODELING PROCEDURE

3.1 Introduction

The goal is to analyze the model equations for handling unconventional tire behavior. We will first look at the steps involved in constructing a code for the available tire data. The bulk of the MATLAB code used for this research was contributed to the *Tire Testing Consortium* TTC by William 'Bill' Cobb, Vehicle Dynamics Center at General Motors [9]. The code was intended to be used by the FSAE member teams of the TTC as a starting point for tire data analysis [10]. The underlying model is the 2009 (MF5.2) version of the Pacejka model. The code can be divided into three sections: preprocessing, fitting, and post-processing. Preprocessing involves importing and storing the data, analyzing and deleting outliers and finally storing it to be used in fitting. It is covered in Section 3.2. Fitting deals with defining the starting coefficients for a Pacejka model and running a fitting routine to obtain the best-fit coefficients. It is covered in Section 3.3. The end of the code comprises of post-processing in order to visualize and analyze the fit. Section 3.4 contains explanation of this process. The sections also contain information about the MATLAB functions used to benefit the reader. The procedure is similar to that suggested in [11]. The succeeding chapter will deal with analyzing specific aspects of this code and changes introduced to implement different data sets and versions of the Pacejka model.

3.2 Data Preprocessing

3.2.1 Importing and Storing

The first task is to import the data into MATLAB and store it in the workspace. The $uigetfile$ function is used to store the filename and pathname that is selected by

the user. It is used in conjunction with the *importdata* function to store the entire raw data as a structure variable. Channel names and corresponding data can be extracted from it for the purpose of storing in usable variables. Before proceeding with storage in MATLAB variables, it helps to delete unrequired data such as that obtained during tire and surface belt warm-up. Information regarding number of data points of warm-up can be evaluated from the 'Run Guide' accompanying the data set. The warm-up procedure can then be traced in raw data files and deleted. The number of warm up points differ with each tire test on the machine. The data is stored by splitting into different variables. Variable names are extracted from the channel names by using the *strtok* function. Dynamic assigning can be employed to improve the ability of the code to read and store data according to channel names. Now the raw data is split into usable variables but still cannot be used for fitting.

3.2.2 Data Smoothing

As highlighted in Section 2.4, the raw data can have more than sixty thousand data points. The goal is to fit a steady-state Pacejka model. Hence, it is necessary to remove transient and hysteresis effects of the tire. Also, the noise in input controllers and output measurement sensors can introduce more fluctuation in the data. The next step in preprocessing is to smooth out these effects from the raw data.

The data under investigation corresponds to a pure cornering test. The test protocol specifies a tire under free rolling (longitudinal slip equal to zero) on which slip angle sweeps are performed for various loads and inclination angles. We begin by finding the starting and ending points for the sweep. The run guide mentions slip angle sweeps begin from -4° to +12° to -12° and end at +3°. A spline fit shifted by -3.5°, which is roughly the starting angle of the sweep, is applied to the data. The spline fit results in a continuous function shifted by -3.5° to locate the zero level. Thus, the zero locations of the spline correspond to sweep start and end points and can be used for referencing them. The testing protocol may also include a shutoff between two sweeps

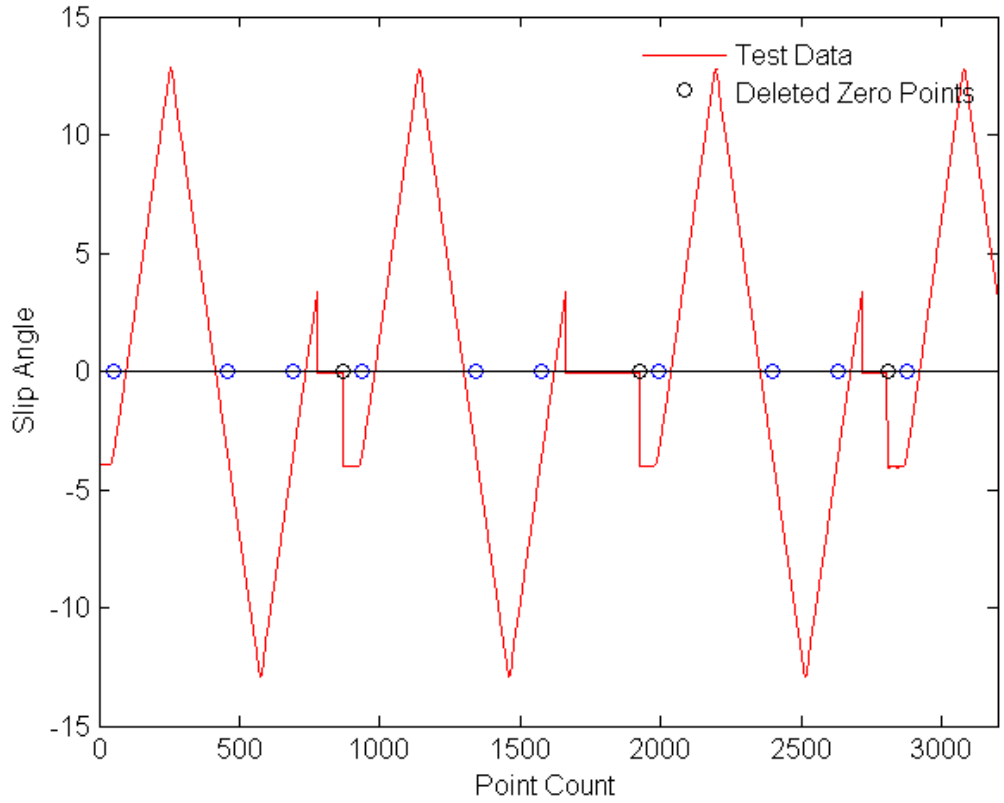


Figure 3.1: Visualizing sweep start and end points [9].

to change loads and/or inclinations and allow the tire to reach steady-state. They are also captured by the spline as zero points and hence, sweep start and end points. These "false zeros" have to be eliminated. The technique employed is to compare with the previous and next data site corresponding to the zero value of the spline. If they lie within a hundred points of the current zero data site, then they can be omitted. Figure 3.1 shows the plot obtained and zoomed in for the first few sweeps. The black circles depict the eliminated zero points or 'shutdown' points.

We are now in a capacity to store and process data for individual sweeps. The next part of the code deals with storing lateral force (Fy), aligning moment (Mz), overturning moment (Mx) and normal load (Fz) data for a single sweep at a time. Hysteresis effects observed at low values of slip angle are taken care of by the "sa" program. It is a compensating method developed by *Calspan's* Tire Research Facility

(TIRF) and is performed on the raw data. In Section 2.4, the normal load capacity of the test machine was stated to be 12000 lbs with a minimal normal load of 50 lbs. However, the test procedure for these small tires have a maximum load of only 350 lbs. This small range can be problematic for the machine load controller. Hence, fluctuations are introduced which lead to fluctuations in output characteristics. Figure 3.2 shows a sample of normal load control fluctuations. It is seen that the accuracy is about ±50 N or ±11 lbs. The code employs a normalization method to reduce this error. It comprises of multiplying each data point of an output characteristic (Fy,Mz,Mx) with mean of the normal load for the sweep divided by normal load at that instant (Fz). An example equation for correction for lateral force is shown in Equation 3.1. Here, fy_i and fz_i store the instantaneous values of lateral force and normal load for a sweep respectively and fy_i' is the lateral force corrected for mean value of fz. Consequently, individual Fz values for the sweep are changed to a single mean Fz value for all points in the sweep.

$$fy_i' = \frac{fy_i * mean(fz)}{fz_i} \tag{3.1}$$

Another effect that is not suitable for steady-state modeling using the Pacejka model is peaks observed in self-aligning moment data. These peaks are a result of response of the machine to change in direction at maximum slip angle. We will discuss the procedure of spline fitting to smooth the data ahead. It is necessary to remove these outliers for a better spline fit. Typically, peak Mz is obtained near and before the lateral force attains its peak value. This region is selected by storing location of the maximum slip angle point and selecting hundred points around it. A cubic polynomial fit is applied to Mz data in this range. The values obtained by this cubic polynomial fit in the selected range is substituted in place of the data obtained from testing for a specified condition. The condition for the code to identify a point as an outlier is the difference between raw data value and that obtained by polynomial

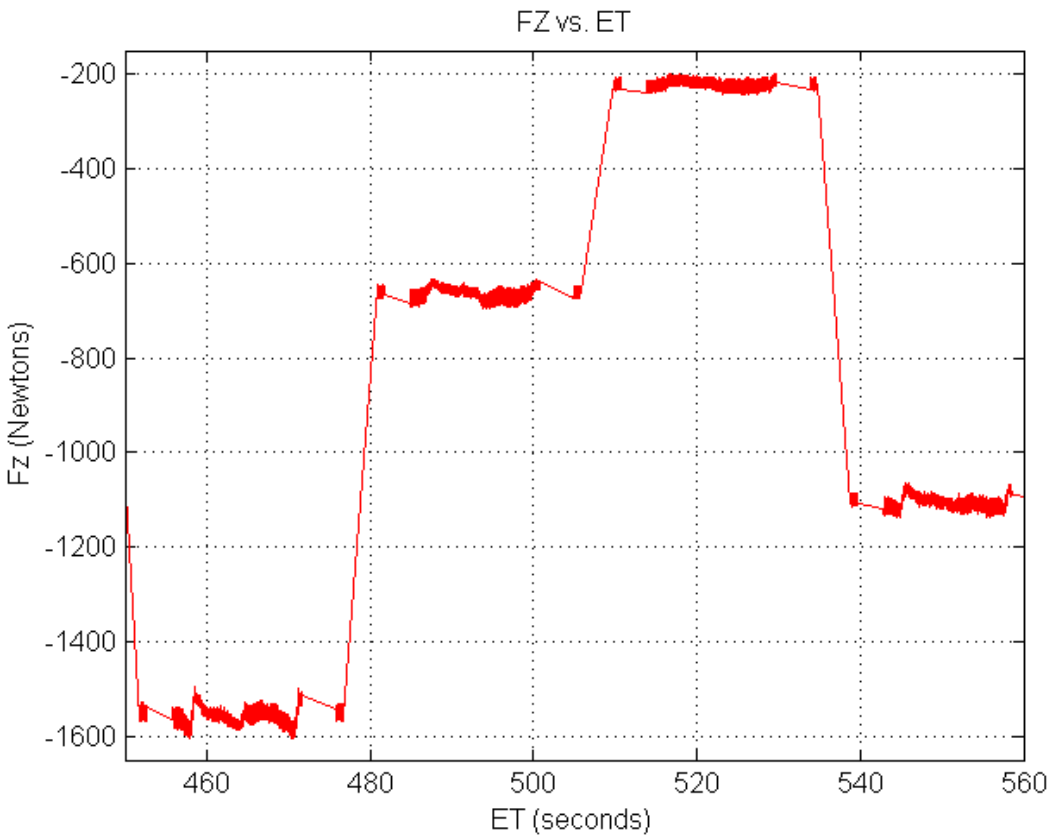


Figure 3.2: Fluctuation in Normal load control.

fit should be greater than 7 N-m. Substitution is carried out with the help of logical indexing. The process is repeated for minimum Mz and does not warrant a separate explanation.

The final task in preprocessing involves smoothing Fy, Mz and Mx using a cubic smoothing spline fit. The "sa" program by *Calspan* mentioned earlier compensates for hysteresis only where SA goes to zero. There is still chatter observed at high values of slip angle. Similar effect is observed for Mz and Mx. The cubic smoothing spline with a defined fit tension captures the characteristic curves. The new values obtained from the fit replace the raw data for Fy, Mz and Mx. The advantage is the ability to capture the same curve behavior with a reduced data set. This in turn helps improve fitting as the routine won't be bogged down by a large data set. Details pertaining to fitting are covered in the next section.

3.3 Fitting

The fitting process begins by defining the nominal vertical load (Fzo), unloaded tire radius (Ro), the nominal inflation pressure (p_{i0}) and the user scaling factors. This is followed by setting the initial guesses for the coefficients appearing in the equations of the model. These will be referred to as the "starting coefficients" from here on. The value of Fzo, Ro and p_{i0} is obtained from the data. The value of all user scaling factors is set to 1. The starting coefficients are set as suggested by the *Netherlands Organization for Applied Scientific Research* (TNO), which is the official research group for Pacejka model developments. For the MF5.2 version of the Pacejka model, for pure slip, there exist twenty four user scaling factors, eighteen coefficients for fitting Fy, twenty-five coefficients for Mz and three for Mx fit. Thus, this code attempts Fy, Mz and Mx fit using a Pacejka model with forty-six coefficients. The coefficients and their initial guesses are shown in Table 3.1.

Table 3.1: Initial guesses of the coefficients.

Coeff.	Value	Coeff.	Value	Coeff.	Value	Coeff.	Value
PCY1	-1	PDY1	-3	PDY2	-0.15	PDY3	1
PEY1	1	PEY2	-0.1	PEY3	0	PEY4	-1
PKY1	-50	PKY2	3	PKY3	2	PHY1	0.003
PHY2	0.0025	PHY3	-0.2	PVY1	0.02	PVY2	0.02
PVY3	-3	PVY4	-0.05	QBZ1	10	QBZ2	-2
QBZ3	-1	QBZ4	-1	QBZ5	-1	QBZ9	20
QBZ10	-2	QCZ1	1.18	QDZ1	-0.1	QDZ2	-0.01
QDZ3	10	QDZ4	-100	QDZ6	0.01	QDZ7	-0.0002
QDZ8	0.1	QDZ9	-0.1	QEZ1	-1.6	QEZ2	-0.36
QEZ3	-1	QEZ4	0.17433	QEZ5	-0.9	QHZ1	0.005
QHZ2	-0.0019	QHZ3	0.005	QHZ4	-0.08	QSX1	0.02
QSX2	5	QSX3	0.2				

The starting coefficients corresponding to Fy fitting are supplied for the first iteration of the fitting routine. A generic fit is obtained using these coefficients. The MATLAB function $lsqcurvefit$ is used to optimize them. As the name suggests, it

employs the least-square regression technique to a non-linear function. Here, the non-linear function supplied contains the Fy fit equations of the MF5.2 Pacejka model. The inputs are combined to form an array of slip angle, normal load and inclination angle data. For the 2012 version of the Pacejka model, inflation pressure is included as an additional input. $lsqcurvefit$ optimizes the coefficients to minimize the sum of, the square of the difference between Fy value obtained from the model equations and the now processed data.

The output of $lsqcurvefit$ consists of the optimized coefficient set, norm of the residual, residual values and an indicator of the condition that ends the optimization. They are used to analyze the fitting routine and will be referred to in the next chapter. The coefficients are plotted as individual bar plots in a single parent plot. This completes one iteration of the routine. For the next iteration, the coefficients obtained from the current iteration are modified and then supplied. The modification is termed as bootstrapping and involves subtracting a random value of the order of e-15 from the best-fit coefficients of the current iteration. The next iteration is performed and the routine stops at twenty iterations. A sample plot to observe variation in coefficients is shown in Figure 3.3. A similar routine is employed for fitting Mz and Mx. To visualize the quality of the fit, post-processing steps are performed as described in the next section.

3.4 Post-processing

A three dimensional plot of slip angle vs. vertical load vs. lateral force at zero camber angle is depicted in Figure 3.4. Two sets of data, one obtained from processing the raw data and one obtained from the MF5.2 Pacejka model fit are plotted. Additionally, a cubic smoothing spline is reapplied to the processed data and super-imposed on the existing plot. This helps to visualize the ability of the model to capture effects of specific inputs.

Post-processing starts with extracting data corresponding to specific inputs by

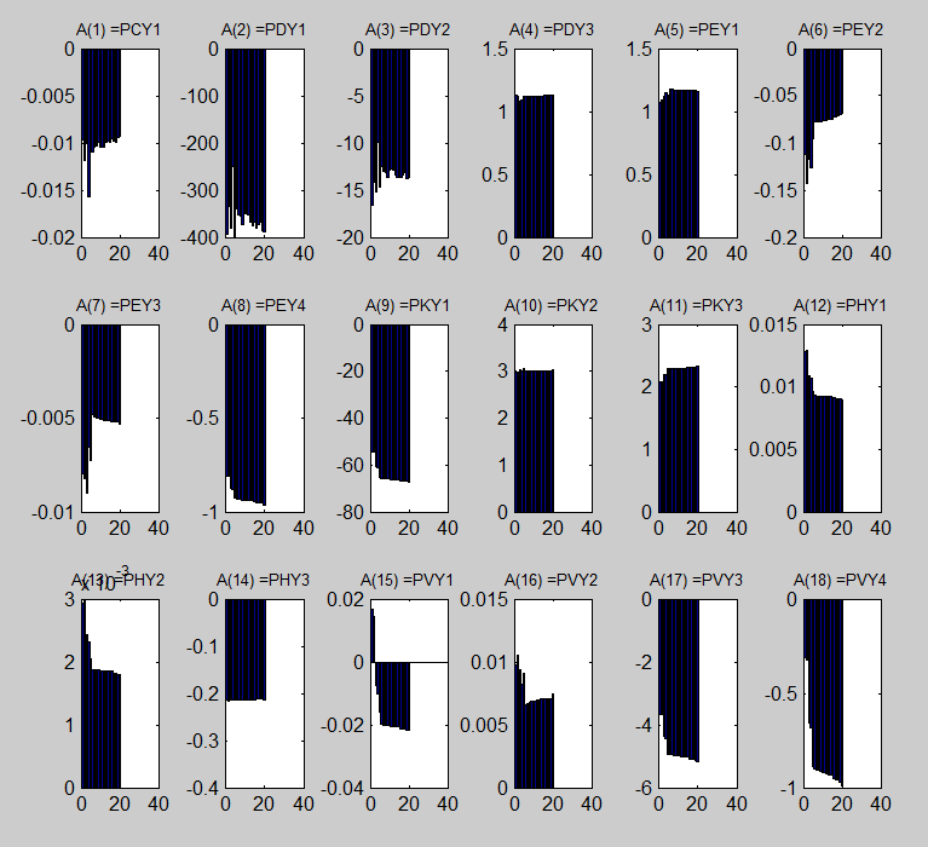


Figure 3.3: Fy fitting routine.

using logical indexing. For cubic smoothing spline, the MATLAB function *csaps* is used. It is the same function used for cubic smoothing spline in preprocessing Section 3.2.2. The only difference being that here it is applied to three dimensional data and generates a surface fit. The input to *csaps* are the data sites of slip angle and vertical load at which the lateral force is to be evaluated. For *csaps* to run, there is a need for all inputs matrices to have the same length. Hence, lateral load matrix dimensions should be changed. The number of rows should be the same as number of unique loads and number of columns must be equal to number of unique slip angles. Similar surface fit plots can be obtained for aligning moment, overturning moment, pneumatic scrub and pneumatic trail.

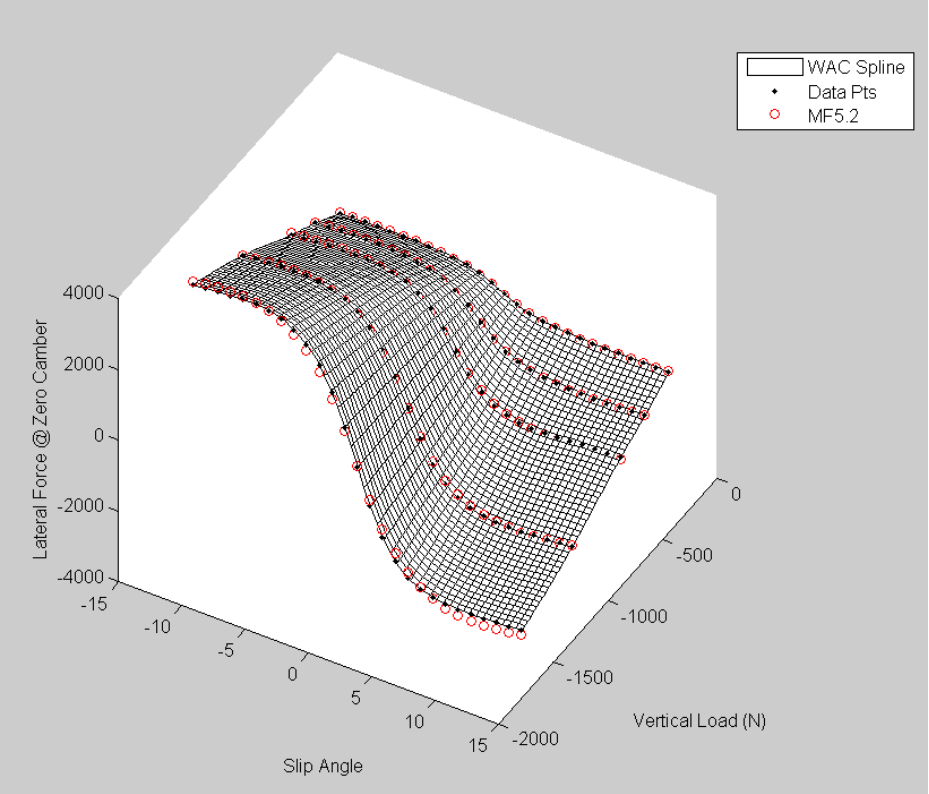


Figure 3.4: Comparing the fit obtained by MF5.2 and by cubic smoothing spline to the data.

3.5 Conclusion

This chapter laid the foundation for the underlying code of this thesis. It started with basic importing and storing the raw data in usable form for the code. The second part of preprocessing consisted of extracting sweep start and end points and dealing with outliers and other errors in the raw data. The final part of preprocessing involved compressing the data with the help of a cubic smoothing spline. The MF5.2 fitting was explained with respect to initial guesses of the coefficients followed by the fitting routine for lateral force. The last step in fitting involved a plot to see the change in coefficient values with every iteration. The final section of this chapter described the surface plot technique of visualizing the quality of fit. It can be applied to investigate a number of tire characteristics. In the next chapter, the code will be further analyzed. It will also cover the modifications required to work with different

data sets and different versions of the Pacejka model.

CHAPTER 4: CODE ANALYSIS FOR CONVENTIONAL BEHAVIOR

4.1 Introduction

Here, the steps involved in preprocessing and fitting are analyzed in detail with a view to improve the modeling process for tires exhibiting conventional behavior. The preprocessing analysis is split between effect of normal load correction, removing aligning moment outliers and cubic smoothing spline fit and tensioning in Sections 4.2, 4.3 and 4.4 respectively. The fitting routine is sensitive to initial guesses of the coefficients. The sensitivity is quantified in Section 4.5 and improved performance is obtained for modeling conventional phenomena by defining a default set of initial guesses. The regression algorithm is changed to analyze effect of using a different one in 4.6. Final analysis of the fitting routine with respect to bootstrapping, reproducibility and number of iterations is included in 4.7.

4.2 Effect of Normal Load Fluctuations

The first step in processing raw data is correcting for normal load control fluctuations. A possible cause of these fluctuations is the normal load controller performance of the *Calspan* TIRF test machine. To analyze the effect of the normalization procedure, a code is written to plot lateral force (Fy) and aligning moment (Mz), before and after the correction is applied. The result is shown in Figure 4.1. The upper half of the figure shows a plot of lateral force vs. slip angle and the lower plot shows aligning moment vs. slip angle. The data used in the plot corresponds to test conditions of inclination angle (IA) = 0°, Fz = 350 lbs and 12 psi pressure. For the lateral force (Fy) vs. slip angle (SA) plot, it is observed that chatter in the data at high values of slip angle (SA) is reduced. Also, at low SA, the hysteresis effect is clearly seen. The

effect of normal load correction is even less apparent on the Mz vs. SA plot. This could mean that the contribution of normal load control error is less in overall error in aligning moment. Similar plots are observed for different load and inclination combinations. Hence, adjusting for mean Fz, although useful, is not sufficient to smooth the data. The same holds true for errors in other input controllers. Without available information such as accuracy and resolution, the effect of output measurement sensors cannot be evaluated.

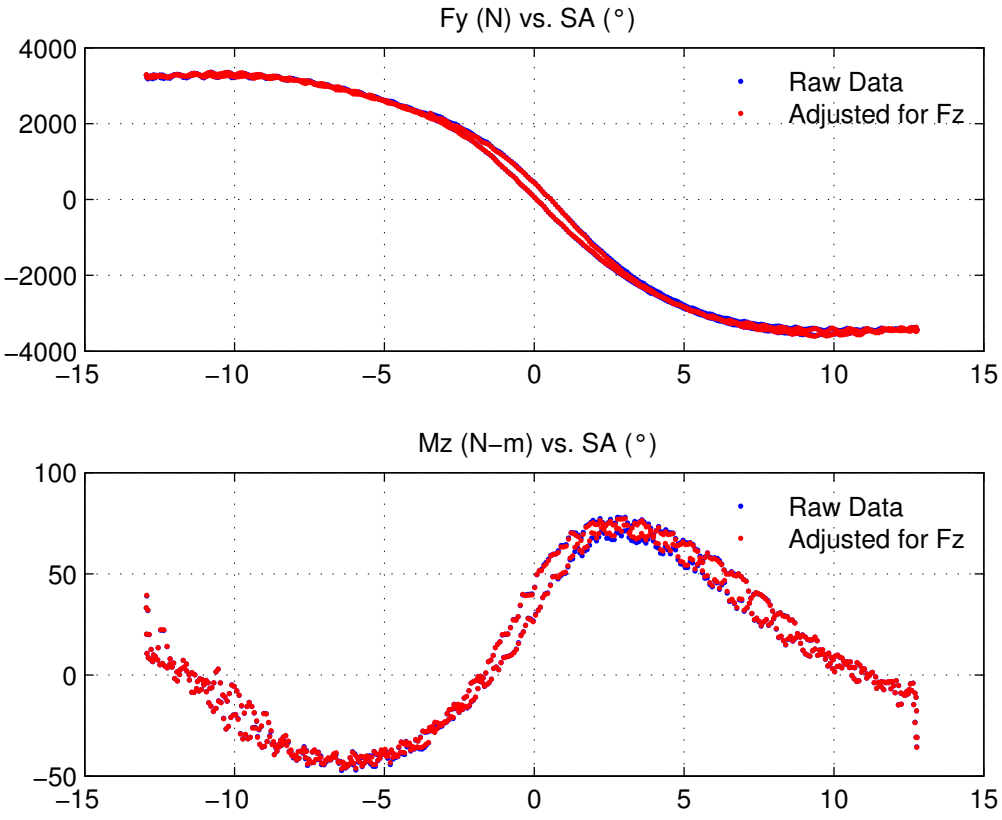


Figure 4.1: Effect of Fz correction.

4.3 Removing Aligning Moment Outliers

Part of the code in Section 3.2.2 also dealt with elimination of sudden peaks in self aligning moment (Mz). These peaks are attributed to the inertia of the test machine when the tire changes direction after reaching maximum slip angle. It can

be seen in the lower plot of Figure 4.1. The steady-state Pacejka model is not able to identify and capture these effects. It can throw the fitting routine off and lead to a reduced quality fit. It also has an affect on cubic smoothing spline fit accuracy. Figure 4.2 shows the cubic polynomial fit applied to raw Mz data. It is important to note the difference between cubic smoothing spline fit and cubic polynomial fit. In case of a cubic polynomial fit, it uses the same polynomial of degree 3 to fit the curve between data points. Cubic smoothing spline, on the other hand, involves a second order derivative of the function integrated over the entire data and weighted by a tension parameter to improve the continuity of the fit. Thus, cubic smoothing spline is preferred for smoothing the entire span of the data which is highly non-linear. But, a polynomial fit is sufficient to fit Mz for a given operating condition at a time. The effect of degree of the polynomial is evaluated ahead. If the difference between a point on this curve and the raw data is greater than 7 N-m, it is treated as an outlier. The threshold value is decided from experience. It can be different for different tires and requires analysis before setting. The outliers are replaced by value of corresponding points on the curve. They are represented by black dots. The red points depict the data after this processing step.

However, a cubic polynomial fit might not always lead to a smooth transition when raw data is substituted by cubic fit approximation. It has a different effect for different tires. Figures 4.3 and 4.4 are used for comparison with a quadratic polynomial fit. Using a cubic fit, the resulting curve is not smooth. This leads to inaccuracies in assumption of steady-state Mz values and outlier selection. This error can also propagate in succeeding processing steps. When analyzed for different operating conditions and different tires, the quadratic polynomial fit is better suited for steady-state Mz approximation. Cubic polynomials should be used only when there is significant improvement in the fit. Accordingly, comparison plots should be generated before selecting degree of the polynomial. The degree of polynomial fit can

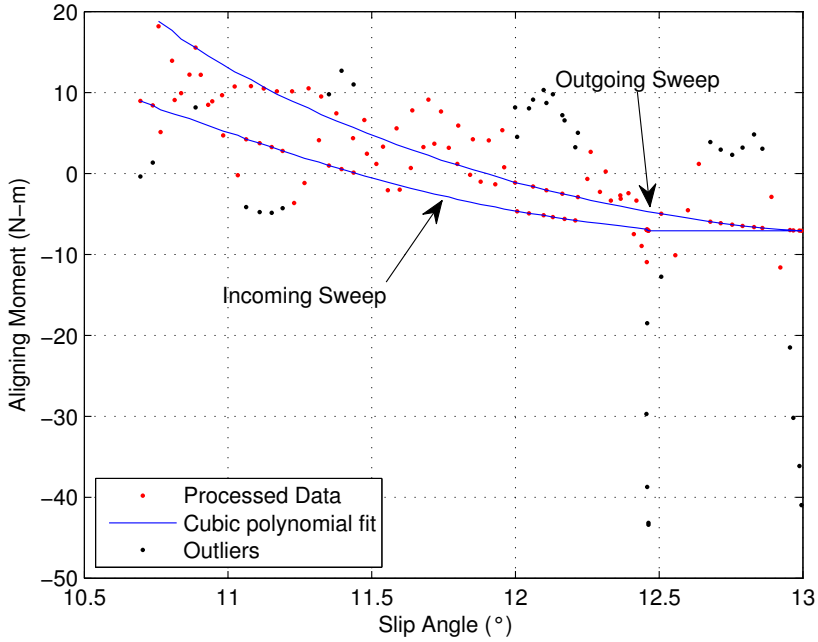


Figure 4.2: Processing of Mz outliers by cubic polynomial fit - plot of Mz vs. slip angle.

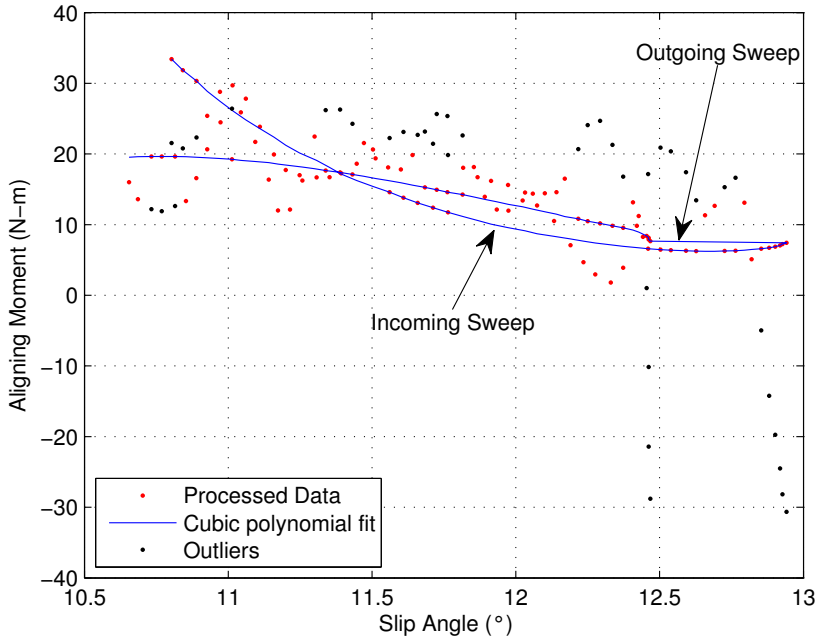


Figure 4.3: Failure of cubic polynomial fit assumption for treating Mz outliers.

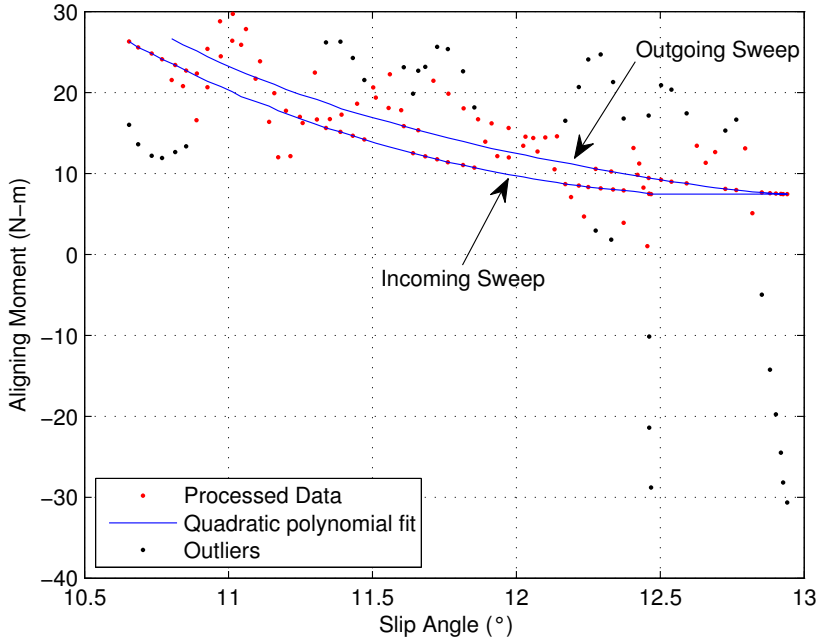


Figure 4.4: Better results obtained from quadratic polynomial fit assumption for treating Mz outliers.

have a knock-on effect on the accuracy of the smoothing spline fit that is investigated in the next section.

4.4 Spline Fit and Tension

A single test sweep for side slip angle (or longitudinal slip) comprises of a up-sweep and a down-sweep. Fluctuations are observed between up-sweep and down-sweep values. An example was seen in Figure 4.1. Fluctuations are also contributed by input controllers and output measurement sensors. There is a need to average these values to form a smooth curve. In order to facilitate fitting performance with respect to computational time, there is a need to capture the tire characteristics with reduced number of data points. We have already seen a 3D implementation of the cubic smoothing function in Section 3.4. A 2D implementation is used to capture the characteristic curves with a reduced data set. It truncates the data from sixty thousand points to a thousand points. The reduced data set is still large enough for a fitting routine to perform accurately. As described in the previous section, the cubic

smoothing method uses a tension parameter 'P' to weigh the effect of second-order derivative of the function. Its value is between 0 and 1. At P=0, *csaps* performs line fitting using the method of least-squares. When P=1, *csaps* uses natural cubic spline interpolation to perform fitting. The code in Chapter 3 used a default P value of 0.1. Figure 4.5 shows the effect of changing P on pure cornering data.

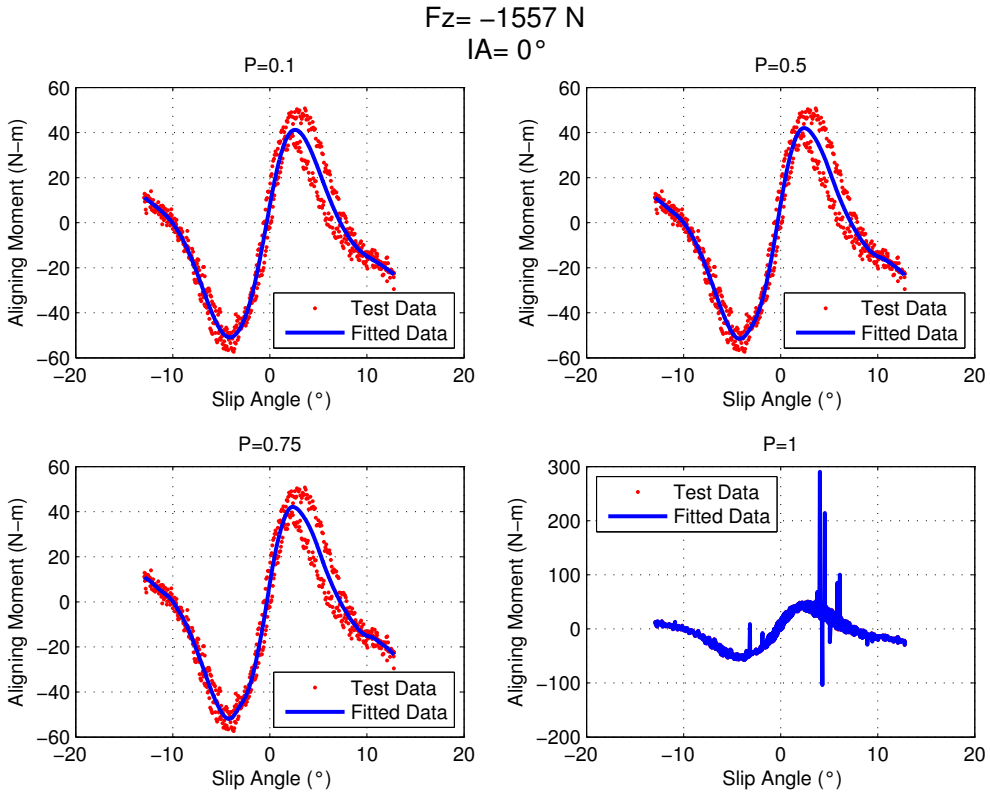


Figure 4.5: Analysis of *csaps* fit for P = 0.1, 0.5, 0.75 and 1. Tire operating conditions: Fz=350 lbs, IA=0°

The curve used for analysis is aligning moment vs. slip angle. It is observed that *csaps* is able to capture the aligning moment stiffness with equal accuracy for P=0.1 to P=0.75. It, however, fails drastically at P=1. This is a result of the cubic spline interpolation being employed. With change in P, the curve changes towards the end of slip angle range. Although, it is generally able to capture the average values throughout the sweep. It should be noted that the peak values obtained from the spline fit should not be used for limit calculations. They should instead be plucked

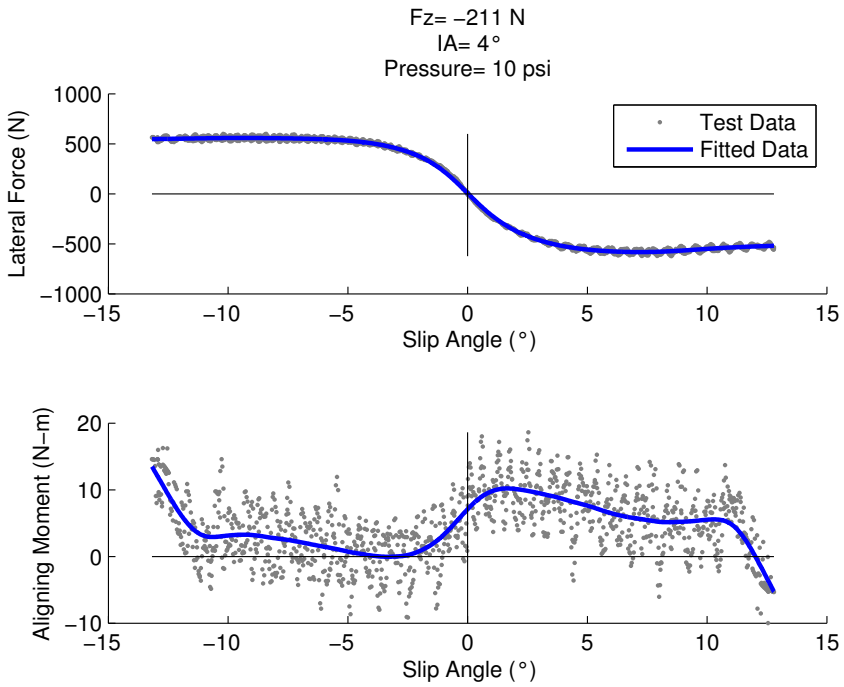


Figure 4.6: Verifying P=0.1 for lateral force and aligning moment. Tire operating conditions: Fz=50 lbs, IA=4°, Pressure=10 psi.

from the raw data.

Before diving into the fitting routine analysis, it is worthwhile to investigate if this tension value is suitable for other characteristics and operating conditions of the tire. Figure 4.6 shows a tire in a pure cornering test under low load, high inclination and low pressure. P-value of 0.1 captures the lateral force characteristic accurately. It is also fairly able to capture the aligning moment curve. The definition of *csaps* by MATLAB specifies that the transition region from least-squares straight line algorithm to cubic spline interpolation is very small and greatly dependent on the data. An example of this can be seen in Figure 4.7 where the tire is under a drive/brake test. The spline is unable to accurately capture the underlying curve until P is specified to the fourth decimal place. In some conditions, P=9.9999e-01 conforms even better with the data. Figure 4.8 shows the cubic smoothing spline for lateral force and longitudinal force under combined slip operation of the tire.

This completes the analysis of the data processing and smoothing part of the

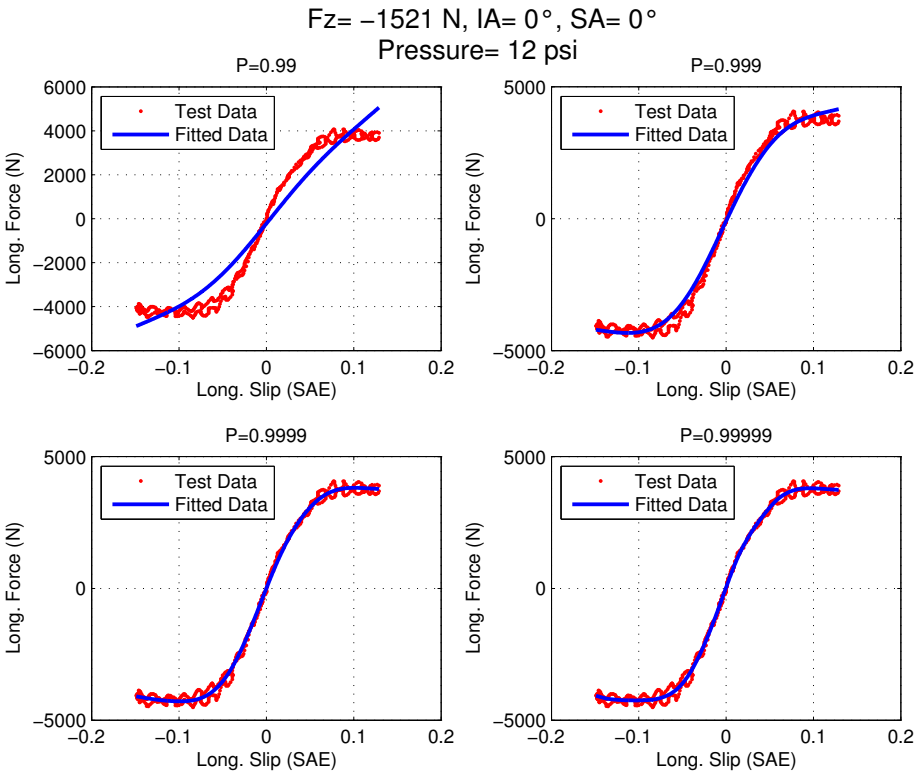


Figure 4.7: Small transition region of *csaps* as seen in longitudinal force characteristics. Tire operating conditions: Fz=350 lbs, IA=0°, SA=0°, Pressure=12 psi.

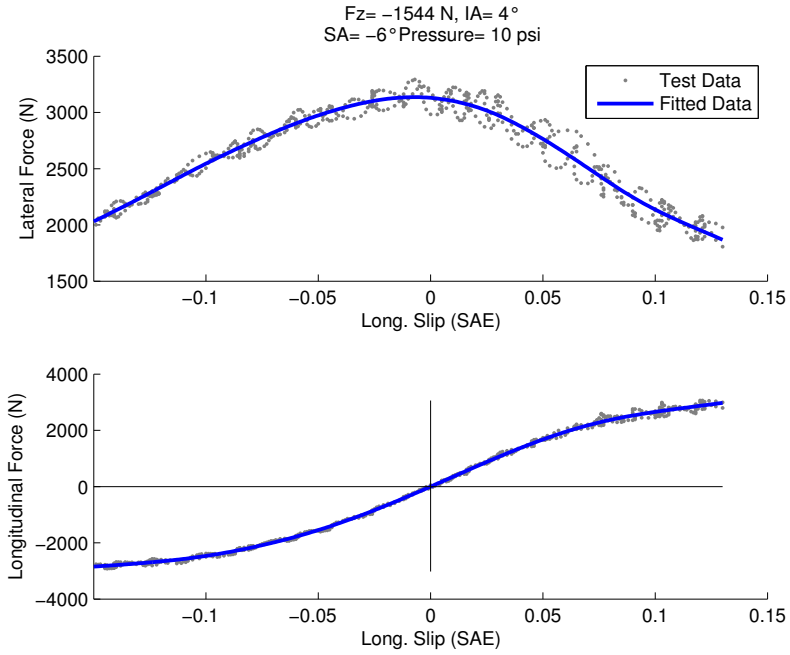


Figure 4.8: Verifying P=0.9999 for lateral force and longitudinal force. Tire operating conditions: Fz=350 lbs, IA=4°, SA=-6°, Pressure=10 psi.

code. The most crucial step in data smoothing is defining the tensioning parameter of the cubic smoothing spline such that it can accurately mimic the tire characteristics, across all operating conditions. Having established the characteristic curves for pure and combined slip operations, the next step is to investigate specific aspects of the fitting routine. We start with the initial guesses of coefficients in the next section.

4.5 Sensitivity to Starting Coefficients

The effect of initial guesses of the coefficients will be examined on the basis of computation time of the fitting routine, quality of fit and reproducibility of the results. Modeling of pure slip lateral force is analyzed as generating its better fit improves fit for other characteristics such as aligning moment and combined slip operation. Table 3.1 in Section 3.3 showed the starting values of the coefficients used in the default program. These were suggested by TNO for the MF5.2 version of the Pacejka model. The eighteen lateral force coefficients will be referred to as 'TNO18' from here onwards. Apart from the MF5.2 (2009) model, we will also investigate the effect of starting coefficients on the 2002 version defined in [7] and the 2012 version (also known as MF6.1.2) defined in [2]. Modifications are made in the code to use the other two models with different data sets. The modified code is included in the Appendix. The data used is for a similar test protocol obtained from different tires to account for tire-specific deviations in fitting.

MATLAB functions *tic* and *toc* are used to time the optimization of coefficients from their starting values. Residual is the difference between raw data value and that obtained by fitting. Norm of the residual is the square root of sum of the square of the residuals. It is often used as a measure of the goodness of fit [12]. As described in Section 3.3, the norm is an output of the least-square curve fit function in MATLAB and will be referred to as $RESNORM$. A visual comparison of fit quality is done by generating plots similar to the one depicted in Figure 3.4. For the 2009 version and TNO18 coefficients, the time taken for optimization is around 200 seconds. The

$RESNORM$ value is 1.2e+07 after the first iteration and at the end of twenty, it decreases to 2.2e+06. A lower $RESNORM$ value is indicative of a better fit.

Information regarding starting coefficients may not always be available. The performance of any optimizing technique depends hugely on them. Poor starting coefficients can lead to early convergence of the regression algorithm to a local minimum. Appendix 3 of [2] suggests default starting coefficients in absence of related data. It was decided to use these values for all versions of the Pacejka model to investigate the effect on fitting. This also helps to test the regression algorithm for local minima. The suggested default values are shown in Table 4.1. The remaining coefficients are set to zero. The four lateral force coefficients will be referred to as 'PAC4' hereafter. With the PAC4 coefficients, drastic decrease in computation time was observed for the 2009 model. The routine took only 7 seconds to generate the best fit coefficients. While the absolute values for computation time can change with load on the processor, they were compared at the same time. The norm of the residual was also lower at 1.8e+06. By using PAC4 starting coefficient values, a better fit was obtained at a much quicker rate. The process was repeated to ascertain the reproducibility of results. The same output in terms of processing time, $RESNORM$ and best-fit coefficient values was observed.

Table 4.1: Default value of the starting coefficients as suggested in Appendix 3 of [2].

Coefficient	Value	Coefficient	Value
PKY1	-50	PKY2	2
QSY7, QSY8	1	PCX1	1.6
PCY1	1.3	PDY2	-0.05

Similar results were obtained for the 2002 version of the Pacejka model. The optimization time was about 190 seconds and a $RESNORM$ value of 2.8e+06 was obtained using the TNO18 starting coefficients. Using PAC4 coefficients, the time was reduced to 23 seconds and the $RESNORM$ to 1.9e+06. The variation in coefficient

values with every iteration is also less for the latter case. This suggests that the regression algorithm is able to optimize the coefficients in fewer iterations. Reduced number of iterations of the fitting routine can lead to further reduction in computation time. The effect of number of iterations is covered in further detail in the next section.

As described in Section 2.3, the 2012 version of the Pacejka model has additional coefficients to model lateral force variation with pressure. Data from round 6 of the TTC is used which has slip angle sweeps for multiple cambers, pressures and normal loads. The tires used for fitting will be termed as 'Tire A', 'Tire B' and 'Tire C' due to restrictions in revealing tire data. The sensitivity of the fitting routine to starting coefficients can be highlighted by the fact that the regression algorithm fails to generate a good fit using TNO18 coefficients (The additional coefficients appearing in the 2012 version for which the TNO18 do not provide a value are set to zero). This was found to be true for all tires. Using PAC4 coefficients, for tires B and C, a $RESNORM$ of the order of e+06 was obtained which is indicative of a good fit. The computation time was also lower compared to tire A. For Tire A, however, they led to a $RESNORM$ of 2.5e+08. The computation time was also comparatively longer. The inability of the regression algorithm to generate a good fit can be attributed to the fact that for Tire A, peak lateral force decreases with increase in camber. A split-optimization method is proposed in Section 5.2 to improve the fit. The results of sensitivity of the fitting routine to the starting coefficients are shown in Table 4.2. It was decided to use PAC4 coefficient values as initial guesses as they led to a better performance of the fitting routine. The different values of the coefficients also increase the confidence in the algorithm to attain global minima.

4.6 Effect of Algorithm Used

It was decided to investigate the effect of algorithm used for non-linear least square curve fitting. The default algorithm for $lsqcurvefit$ is called "Trust-Region-Reflective". An additional algorithm called "Levenberg-Marquardt"(lm) is also avail-

Table 4.2: Results of the starting coefficient analysis

Version	Coefficient Set	*RESNORM*	Time (s)
2002	TNO18	2.2e+06	200
	PAC4	1.8e+06	7
2009(MF5.2)	TNO18	2.8e+06	190
	PAC4	1.9e+06	23
2012	TNO18	4.8e+08	18
	PAC4	4.3e+06	16

able to use. Using the *lm* algorithm, a degraded fit quality and increase in computation time was observed. This was true for all versions of the model and all tires. This observation is in accordance with [13] and should be referred to for more details regarding the two algorithms. It is recommended to use the default algorithm when using $lsqcurvefit$.

4.7 Bootstrapping Procedure, Reproducibility and Number of Iterations

Bootstrapping in the code essentially involves using the optimized coefficients at the end of one iteration, subtracting a small random value from it and then supplying them as input for the next iteration. The random value is of the order of e-15. Its order of magnitude is very low because the MATLAB random number generator *rand* is multiplied by *eps*. *eps* returns the distance between consecutive numbers of the same data type. It is also a measure of machine precision. In our case, it returns 2e-52 which corresponds to the 'double' data type in MATLAB. We have already seen the sensitivity of fitting routine to starting coefficients. The desired effect of bootstrapping is to ensure $lsqcurvefit$ does not optimize to a local minimum.

However, the actual effect of bootstrapping is questionable. The value supplied to the next iteration of the routine is almost equal to that obtained after optimization during current iteration. We need to analyze if this method of bootstrapping is really capable of causing $lsqcurvefit$ to optimize to another minimum. The analysis was done by running the fitting routine at three different conditions: 1) As is (boot-

strapping), 2) Without subtracting the small random number (optimized coefficients test) and 3) With same starting coefficients for every iteration (reproducibility test). The results from first and second conditions will reveal the real effect of using the small random number. Condition three will serve as a test of reproducibility of the $lsqcurvefit$ algorithm.

2012 version of the Pacejka model is used for all analysis. For a given set of data, we will look at the trend in residual values and coefficient values with and without the '$-1 * eps * rand$' contribution. Both Fy and Mz residuals and coefficients are investigated to eliminate possible bias due to equation definition. The Fy residuals changed with iteration for both with and without subtracting the small random number. When starting coefficients are equal to the best-fit coefficients from the preceding iteration, change in residuals is of the order of e-02. The contribution of the small random number to this change is of the order of e-06. The residuals remain constant when the starting coefficients are held constant. Hence, for Fy residuals, re-supplying optimized coefficients has a greater impact on $lsqcurvefit$ than re-supplying after subtracting the small random value from them. Also, the $lsqcurvefit$ algorithm is able to reproduce the same values every iteration. The same holds true for coefficient values. Similar observations are made when Mz is analyzed. Thus, bootstrapping by using '$-1 * eps * rand$' is not sufficient to test the fitter for local minimum. We will eliminate this factor from the code. There is a need to have starting values far away from the current minimum to understand if it is indeed a local minimum. Hence, the real test of local vs. global minimum was already done when the fitter was supplied with set A and set B coefficients for the different versions of the Pacejka model. The fact that it resulted in lower norm of the residuals and faster convergence highlights that TNO18 coefficients could have potentially led to a local minimum.

Since, we assessed the reproducibility of $lsqcurvefit$, there is also a need to examine the effect of number of iterations of the fitting routine. Data was gathered for

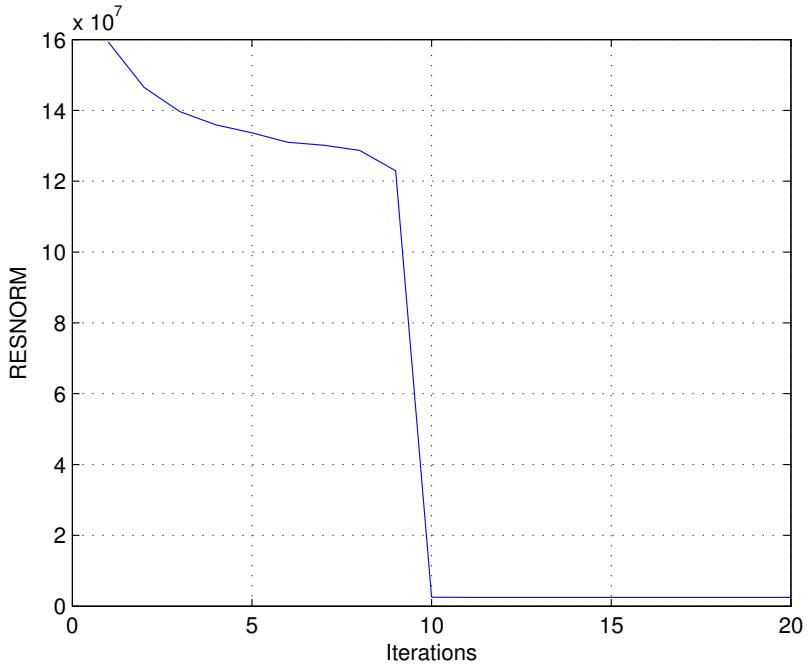


Figure 4.9: Change in value of norm of the residuals with iterations.

the *RESNORM* and final value of coefficients for different number of iterations. A plot similar to one shown in Figure 4.9 can be used to visualize the change in their values with iteration. It may be different for different versions of the model and the data under consideration. The number of iterations should be low enough to save computation time but high enough to ensure repeatability of norm of the residuals and coefficient values.

4.8 Conclusions of Code Analysis

For the preprocessing part, it was seen that the effect of applying normal load correction is minimal to overall smoothing of the data. A better method can be developed to correct for fluctuations arising due to input controllers and output measurement sensors if data related to their accuracy, resolution and logging rate is available. Following on to the aligning moment outlier evaluation, a quadratic polynomial fit should be the default fit of choice. A cubic polynomial fit can be used when it results in significantly better results. The most important step in smoothing the data is applying

the cubic smoothing spline with a defined tension. The tension varies with characteristic and hence, analysis is required for different operating conditions before selecting a value.

For the fitting process, based on all the observations, it can be inferred that initial guesses of the coefficients have a huge impact on performance of the fitting routine and the resulting quality of the fit. This inference stands true despite the underlying modeling equations. Starting coefficients suitable for one set of Pacejka model equations might not yield the desired accuracy of fit for other set of model equations. In absence of available data, the value of the starting coefficients should be set according to Appendix 3 of [2]. This works best for the MF5.2 and the 2002 versions of the Pacejka model. It also works well for the 2012 version. However, the lack of initial guesses hinders the ability of the fitting algorithm to generate an accurate fit when phenomenon such as that for Tire A is observed. Analysis was also done to evaluate the effect of algorithm used in least square curve fitting in MATLAB. Best performance is obtained for when $lsqcurvefit$ uses its default "Trust-Region-Reflective" algorithm. The effect of bootstrapping as defined in the default code was examined. It is found that contribution of the small random value subtraction is insignificant to the ability of the fitting routine to generate best-fit coefficient or to test the fitting algorithm for a local minima. Finally, the number of iterations can be decided by plotting against trends of coefficient and residual values.

CHAPTER 5: MODELING UNCONVENTIONAL BEHAVIOR

5.1 Introduction

Several methods for modeling conventional and unconventional tire behavior were tried in the investigation of an optimized approach. The analysis done in the previous chapter dealt with improving the fitting process for modeling conventional tire behavior. The unconventional nature of the camber curve observed for Tire A is modeled better using the split-optimization method. Different methods of applying this technique were analyzed. For capturing unconventional effects (Tires A and C), fits were improved by first optimizing for camber and normal load effects. This was followed by tire-specific optimization of the remaining coefficients to generate the best fit possible. An improvement over the default regression technique was observed for both the unconventional tires in question. With a limitation on the method, the split-optimization is found to generate an accurate fit even for conventional tires. The talking points for analysis are the change in coefficient values, norm of the residuals and finally visual analysis of the fit. An improved method of modeling unconventional pressure effects is proposed by applying a modification to the model equations. The modification showed positive results for the limited data.

5.2 The Split-Optimization Method

In the analysis of sensitivity to starting coefficients in Section 4.5, the optimization routine failed to generate a good fit for one particular tire, Tire A. The plot in Figure 5.1 shows its variation in lateral force with increasing camber. It is seen that the behavior is contrary to that used for defining the model equations as described in Section 2.3. The effect of camber on another tire, Tire B conforms with that used

for defining the model equations as seen in Figure 5.2. Thus, the fitter is able to generate a good fit for Tire B. The same phenomena as Tire A is observed for Tire C. However, the difference in peak lateral force values is small. The plot is shown in Figure 5.3. This explains why the fitter is able to optimize better even with Tire C showing similar characteristics as Tire A.

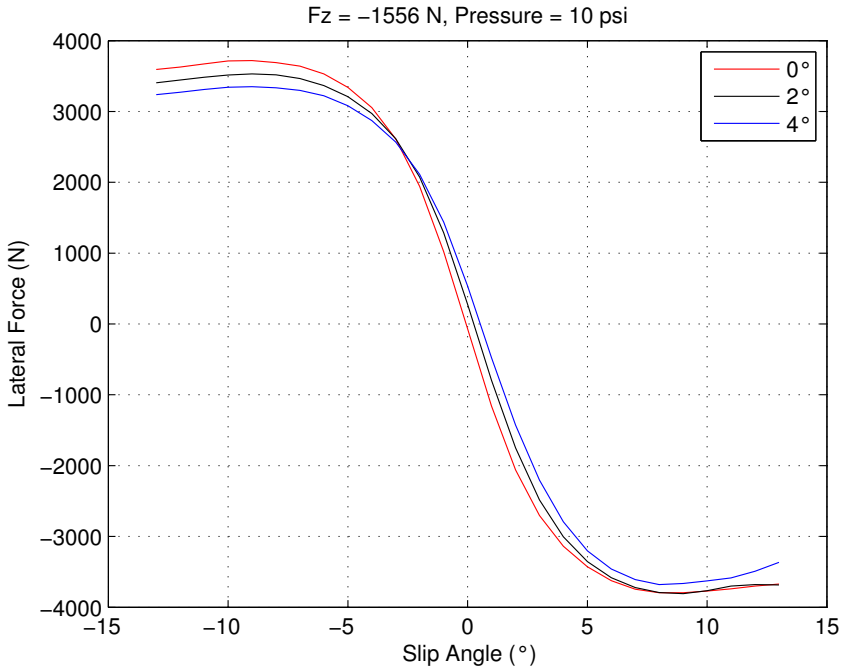


Figure 5.1: Effect of camber on lateral force generated by Tire A. Tire operating conditions: Fz=350 lbs, Pressure=10 psi.

A split-optimization method is proposed to tackle this problem. The method involves keeping one of the inputs, either camber or pressure, constant to generate best-fit coefficients. We do not consider keeping the load constant as it shows expected behavior. The best-fit coefficients can then be used directly, as fixed coefficients, or indirectly, as initial guesses, to assist fitting using data comprising of all the inputs. We will investigate the effect of keeping pressure and camber constant. Figure 5.4 depicts the different ways to implement this technique. Analysis is done to evaluate the most suitable one. Only Tire A is used for analysis.

Data corresponding to a constant camber value will be referred to as fixed camber

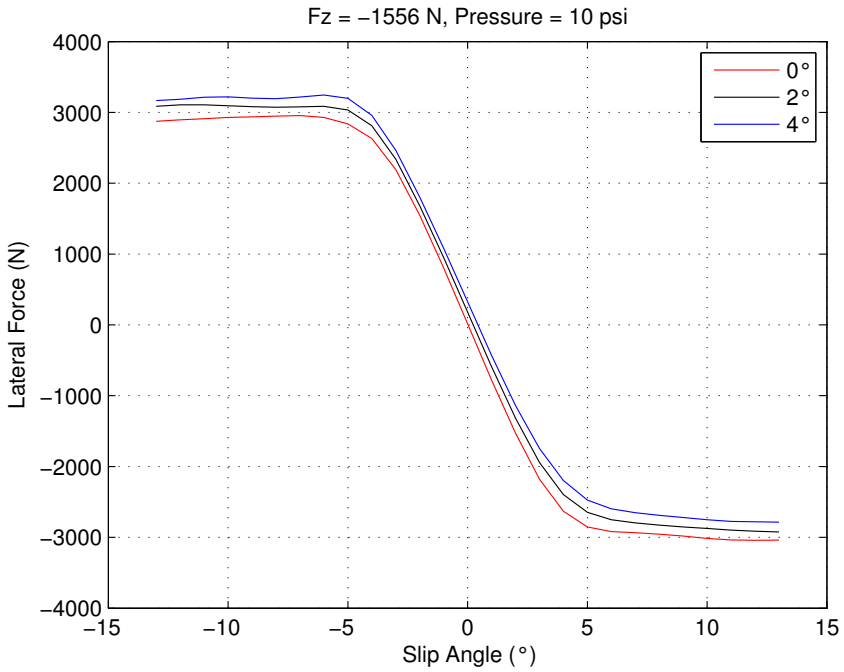


Figure 5.2: Effect of camber on lateral force generated by Tire B. Tire operating conditions: Fz=350 lbs, Pressure=10 psi.

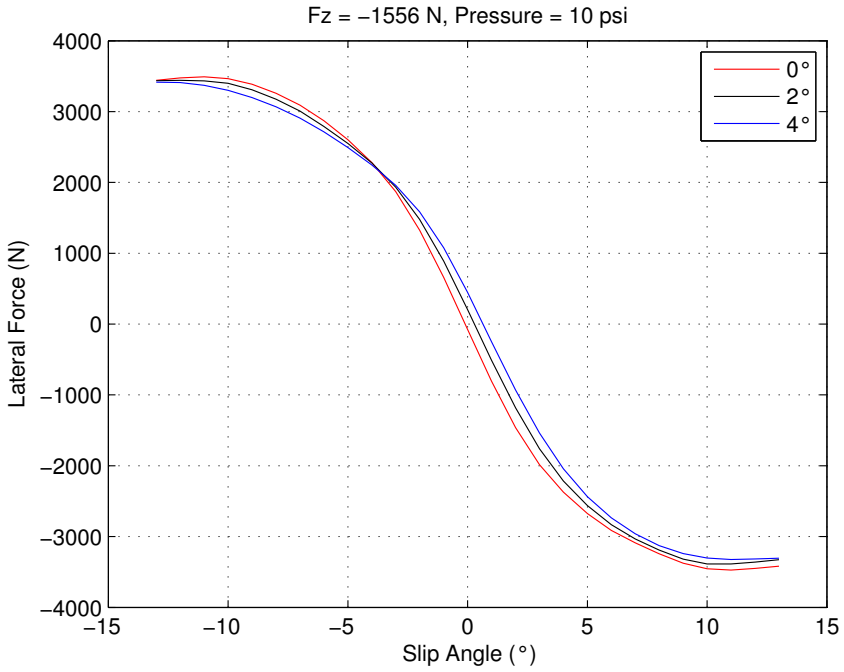


Figure 5.3: Effect of camber on lateral force generated by Tire C. Tire operating conditions: Fz=350 lbs, Pressure=10 psi.

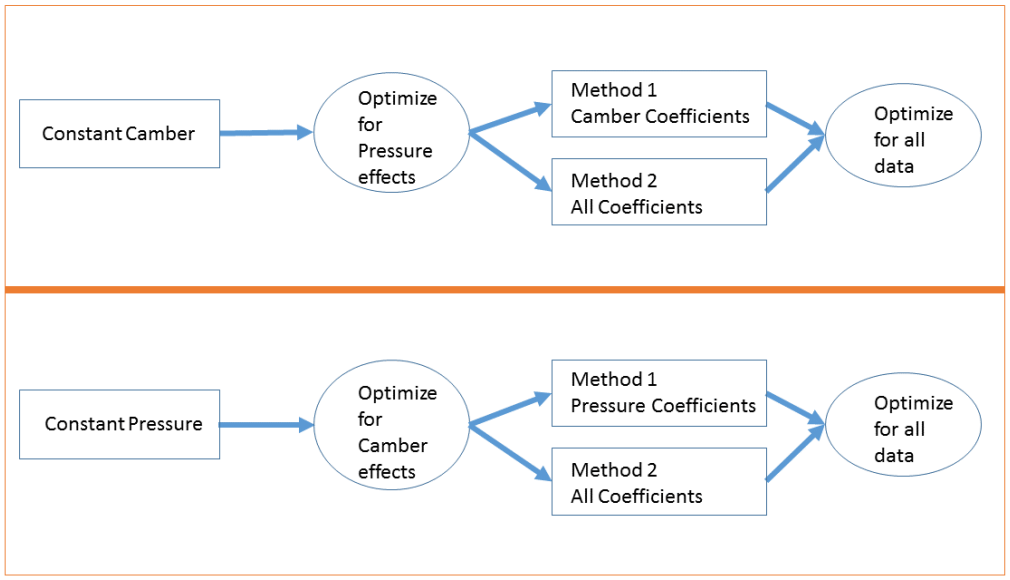


Figure 5.4: Different combinations to apply the split optimization technique.

data. It is extracted by using logical indexing in MATLAB to obtain location of sites corresponding to a desired camber value. In our case, that value is 0°. In the code, data corresponding to 0° camber is stored after completion of processing steps and before fitting begins. The remainder of the process remains same. A bar plot of change in individual coefficient value with iteration is shown in Figure 5.5. It is seen that all coefficients in the model responsible for camber effects remain at zero and are not used by $lsqcurvefit$ for optimizing the fit. The $RESNORM$ value is lower at 2.1e+06 as compared to fitting using all the available test data. It results in an improved fit for modeling effect of pressure on lateral force.

Fixed pressure data is the set of data corresponding to a given value of pressure. Fitting using this data set shows analogous results to that using constant camber data. The points corresponding to 10 psi pressure are extracted from the main data and supplied to the fitting routine. The bar plot for variation in coefficient values is shown in Figure 5.6 below. As expected, the coefficients used for modeling effect of change in inflation pressure remain zero. Critically, the $RESNORM$ value is lower,

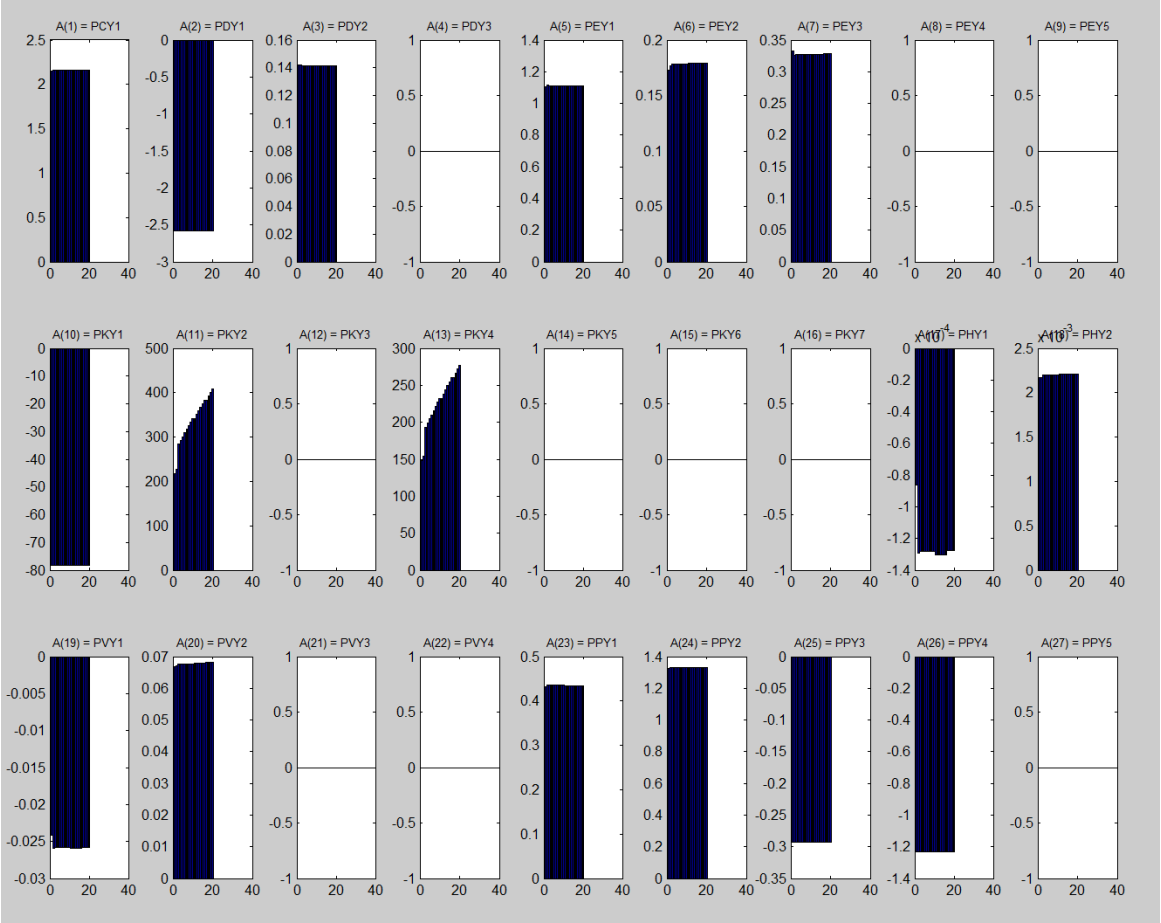


Figure 5.5: Pacejka 2012 coefficients optimization for data pertaining to 0° inclination.

of the order of e+05, resulting in a better fit compared to using camber as the fixed input.

In modeling lateral force, cornering stiffness plays a major role. It varies mainly with normal load and inclination of the tire. The coefficients *PKY1, PKY2, PKY3, PKY4, PKY5, PKY6* and $PKY7$ collectively model cornering stiffness variation with variation in load and inclination. The coefficients $PPY1, PPY2$ and $PPY5$ capture cornering stiffness variation with pressure. In constant pressure fitting, the 'K' coefficients are optimized together. In constant camber fitting, only $PKY1, PKY2$ and $PKY4$ undergo optimization together. The latter hinders the ability of the model to capture cornering stiffness deviations with camber which is critical for Tire A. This leads to the observed poor fit as compared to the former. The other explanation may

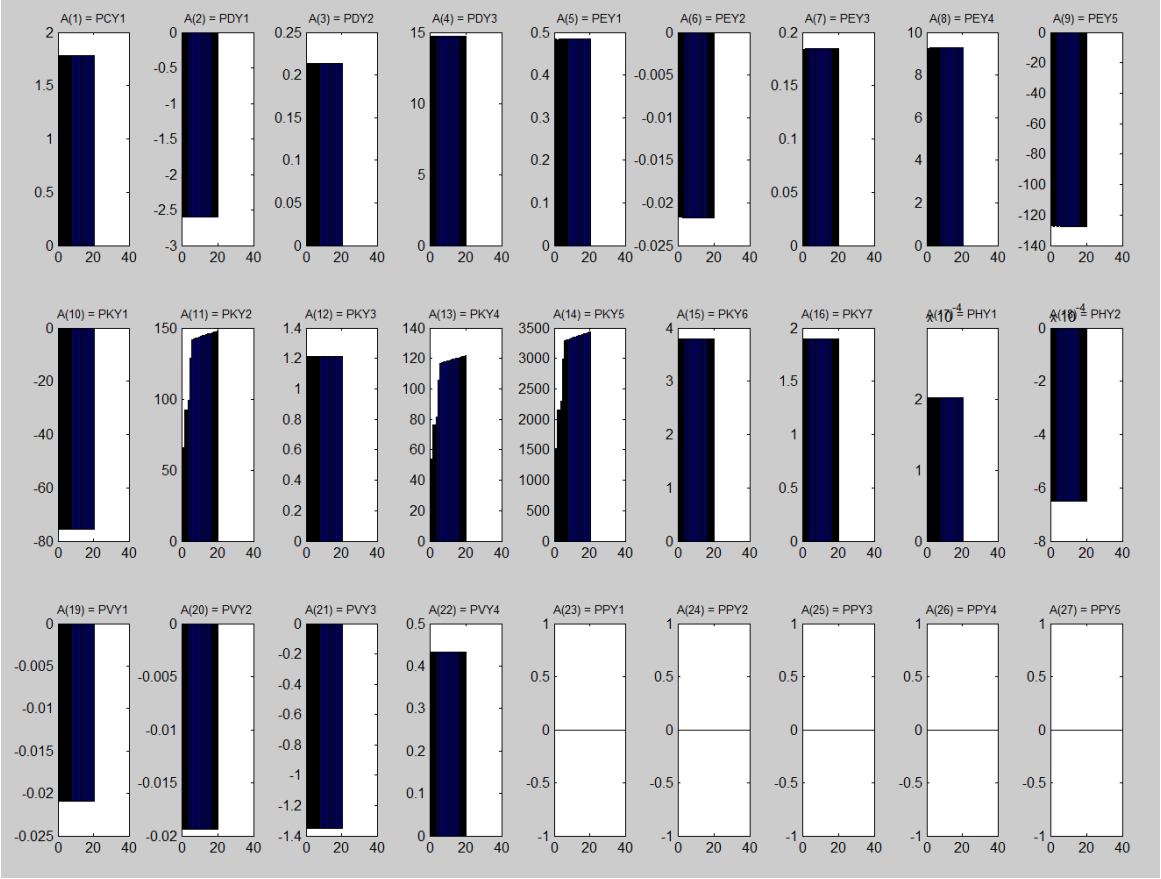


Figure 5.6: Pacejka 2012 coefficients optimization for data pertaining to 10 psi inflation pressure.

be the definition of pressure factors in the model equations and their relatively smaller effect on modeling cornering stiffness. Thus, in fitting for one fixed input, keeping the pressure constant is preferable.

Fitting using a fixed input requires the use of two fitting routines. One routine optimizes the coefficients using data corresponding to the fixed input. The second routine is then used to improve the fit over the entire data. Having obtained best-fit coefficients for truncated data sets, there are now two ways to run the second routine. They are: 1. Optimize only fixed input coefficients, 2. Optimize all coefficients together. The assumption of method 1 is that the non-zero coefficients are able to accurately capture all phenomena related to the varying inputs in the first routine. Thus, in the second routine, the zero-valued coefficients will only model the

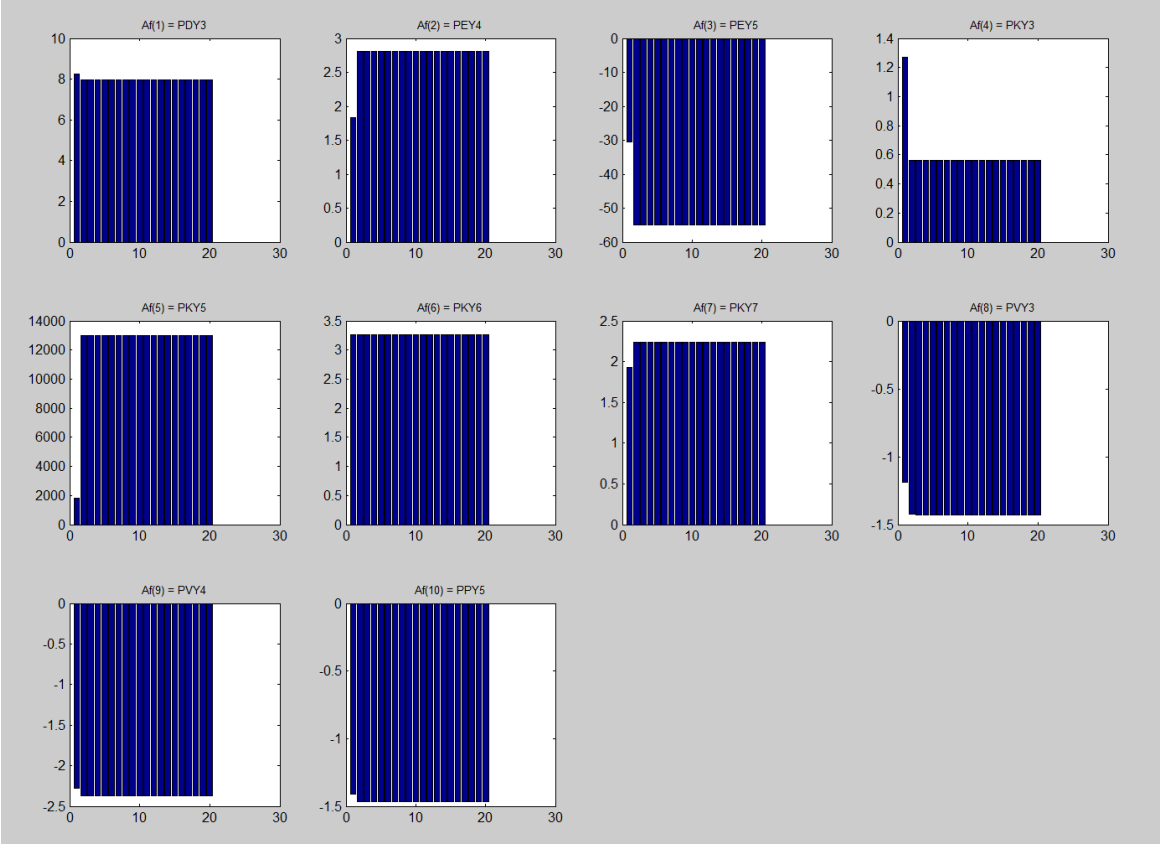


Figure 5.7: Optimization obtained from running the second routine for camber coefficients. Data containing all inputs is used.

effects of the input that was held constant. The second method assumes that there is inter-dependence between coefficients which the first routine fails to capture. This is particularly true when keeping camber constant as seen ahead.

Using method 1 for fixed camber data, all coefficients except the zero-valued camber coefficients are assumed to capture their corresponding effects accurately. The code now passes only the zero-valued coefficients to the fitter. Modification is required to the definition of the model such that it optimizes only the zero-valued coefficients. The zero-valued coefficients vary as shown in Figure 5.7. The *RESNORM* value for fitting just the camber coefficients is 6.4e+07 which is indicative of a good enough fit. Despite the apparent overall improvement in the fit, the camber coefficients do not attain a stable value. This is confirmed when the second method is applied.

$PKY1, PKY2$ and $PKY4$ are the only camber coefficients that are optimized in both fitting routines. Their values from the first fit (fixed camber fit) serve as initial guesses for the second routine (full data fit). At the end of the second routine, their values change drastically along with other K-coefficient values. This suggests that the coefficients obtained from the first fit either correspond to a local minimum or are insufficient to provide a better fit without the other K-coefficients.

This confirms our previous suggestion that it is better to keep pressure constant when performing fixed input fitting. We have already seen that it leads to a better fit from the first routine ($RESNORM$ is lower). Thus, running the second routine with method 1 leads to a better fit. Using method 2 for the second routine, a similar $RESNORM$ value is obtained to when camber was held constant. So, method 2 is unsuitable in either case. However, the deviation in K-coefficient values is less and the fit is stable.

To analyze the deficiency in model equations, we assess the tabulated coefficient values in Figure 5.8. In Section 2.3 that includes the sine definition of the *Magic Formula*, we understand that the coefficient E_y controls the curvature at the peak of the lateral force curve and $K_{y\alpha}$ models the cornering stiffness. When the default full optimization method is used, there is a gain in E-coefficient values and K-coefficient values remain low. Thus, the majority of modeling for pressure and camber effects is accomplished by adjusting the curve at its peak. However, when a fixed input is used, it is observed that K-coefficients gain value and E-coefficient values are smaller. Thus, with a fixed input, the fit is optimized by improving the cornering stiffness modeling and/or by being less dependent on adjusting the peak curvature. In this way, the split-optimization technique improves the output curve by causing the initial slope of the curve to change than to change the curvature at the peak. For Tire A, it was also observed that the variation of cornering stiffness with pressure does not conform with that used for defining the pressure factors in the model equations. It is dealt with

Fixed Camber	
Coefficients	Routine 1
PCY1	2.157201
PDY1	-2.5788
PDY2	0.14119
PDY3	0
PEY1	1.109799
PEY2	0.17887
PEY3	**0.328449**
PEY4	**0**
PEY5	**0**
PKY1	**-78.2655**
PKY2	**408.6377**
PKY3	**0**
PKY4	**277.516**
PKY5	**0**
PKY6	0
PKY7	0
PHY1	-0.00013
PHY2	0.002214
PVY1	-0.02576
PVY2	0.068207
PVY3	0
PVY4	0
PPY1	0.434669
PPY2	1.331099
PPY3	-0.29219
PPY4	-1.2316
PPY5	0
RESNORM	2.13E+06

Fixed Pressure	
Coefficients	Routine 1
PCY1	1.7839
PDY1	-2.5993
PDY2	0.21394
PDY3	14.7344
PEY1	0.48366
PEY2	-0.02171
PEY3	**0.18453**
PEY4	**9.266899**
PEY5	**-127.349**
PKY1	**-75.6603**
PKY2	**147.9047**
PKY3	**1.213801**
PKY4	**121.8247**
PKY5	**3431.753**
PKY6	3.7893
PKY7	1.9009
PHY1	0.000202
PHY2	-0.00065
PVY1	-0.02089
PVY2	-0.01939
PVY3	-1.3505
PVY4	0.43358
PPY1	0
PPY2	0
PPY3	0
PPY4	0
PPY5	0
RESNORM	6.39E+05

Full Optimization	
Coefficients	Value
PCY1	1.506
PDY1	-1.9644
PDY2	-0.59725
PDY3	1.1148
PEY1	-0.00386
PEY2	-0.00774
PEY3	**-56.8745**
PEY4	**-2466.21**
PEY5	**39082.98**
PKY1	**-46.891**
PKY2	**2.2258**
PKY3	**0.17778**
PKY4	**1.7487**
PKY5	**-0.76369**
PKY6	0.20405
PKY7	0.17569
PHY1	-0.00263
PHY2	-0.00772
PVY1	-0.03726
PVY2	-0.16116
PVY3	-0.22124
PVY4	0.8702
PPY1	-0.01963
PPY2	0.066958
PPY3	-0.04225
PPY4	-0.16185
PPY5	-2.8915
RESNORM	2.02E+08

Figure 5.8: Tabulated values of the coefficients using different optimization techniques.

in Section 5.4. Consequently, an argument can be made that the cornering stiffness equation ($K_{y\alpha}$) definition fails when the tire shows unconventional behavior with both camber and pressure. It follows from [8] that the curvature and shape factors must be made inflation pressure dependent.

To gain further insight into the quality of fit, we will generate comparison plots. For the default full optimization method, the plot obtained is as shown in Figure 5.9. For the split-optimization method, starting with fixed camber data set, a 3D plot of SA vs. IA vs. Fy is shown in Figure 5.10. The fit obtained is for zero-valued coefficients optimized using method 1. To analyze its effect on other characteristics, a 3D plot of SA vs. P vs. Fy is created. The plot overlays fits from before and after the camber coefficients are optimized. It is shown in Figure 5.11. The difference is insignificant. Thus, optimizing for the best-fit camber coefficients does not have any effect on fit related to other inputs. However, the underlying fit fails to accurately

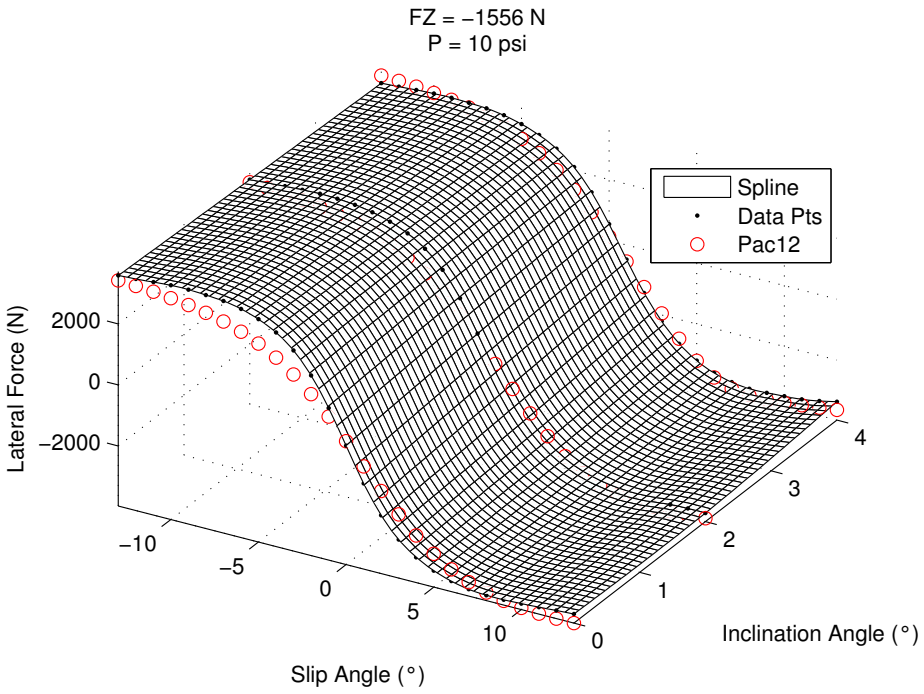


Figure 5.9: Fit obtained using the default full optimization method. Tire operating conditions: Fz=350 lbs, Pressure=10 psi.

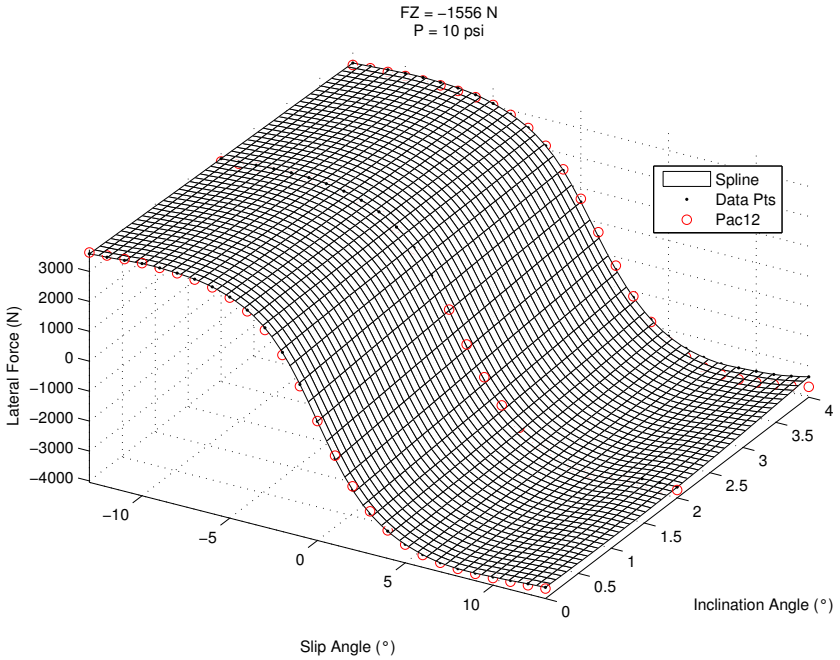


Figure 5.10: Visualizing the fit for optimized camber coefficients using full set of data. First routine runs on fixed camber data and Method 1 is used for the second routine. Tire operating conditions: Fz=350 lbs, Pressure=10 psi.

capture both inclination and pressure effects.

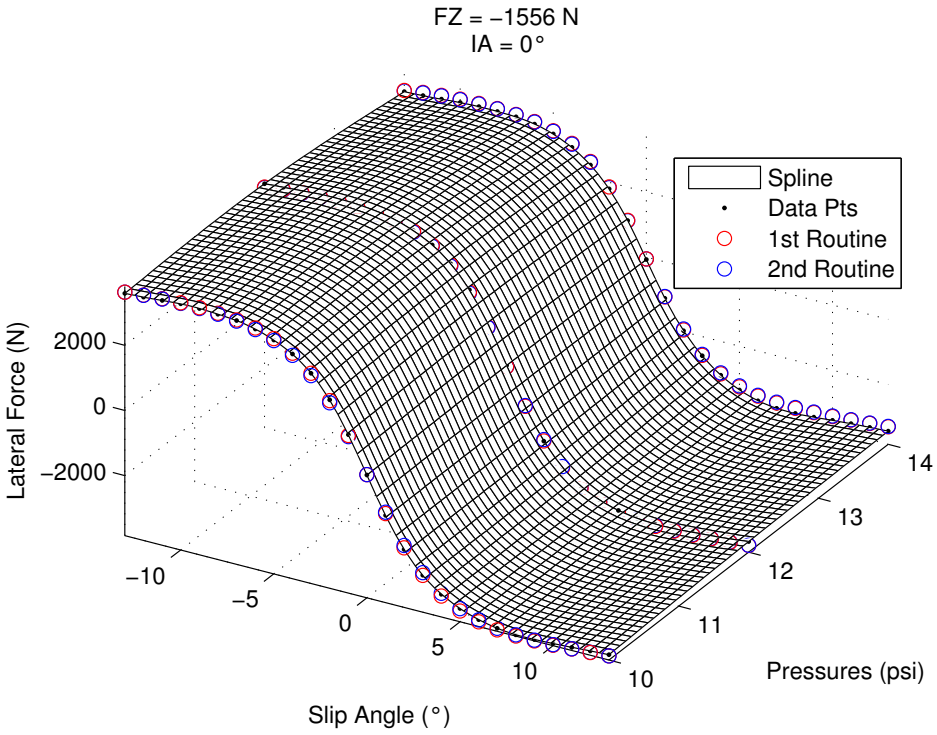


Figure 5.11: Visualizing the effect of optimizing camber coefficients on ability to model pressure characteristics. First routine runs on fixed camber data and Method 1 is used for optimization. Tire operating conditions: Fz=350 lbs, IA=0°

The second method involves re-supplying all coefficients to the fitter. A fit obtained by deriving best-fit coefficients using this method is worse than that obtained by method 1. Figure 5.12 shows a plot similar to the one in Figure 5.11. The fit generated by method 2 is depicted by the blue dots. The red dots depict the fit obtained from the first routine. On comparison with the Figure 5.11, a deteriorated fit quality is clearly visible as it tries to model camber effects.

The plots obtained for fixed pressure data and method 2 are analogous as the fit quality obtained is similar, if not worse. Another validation for using pressure as a fixed input instead of camber is shown in Figure 5.13. The underlying fit quality before optimizing the pressure coefficients is better as compared to Figure 5.10. Optimizing the pressure coefficients has less negative impact on the ability of the model to capture

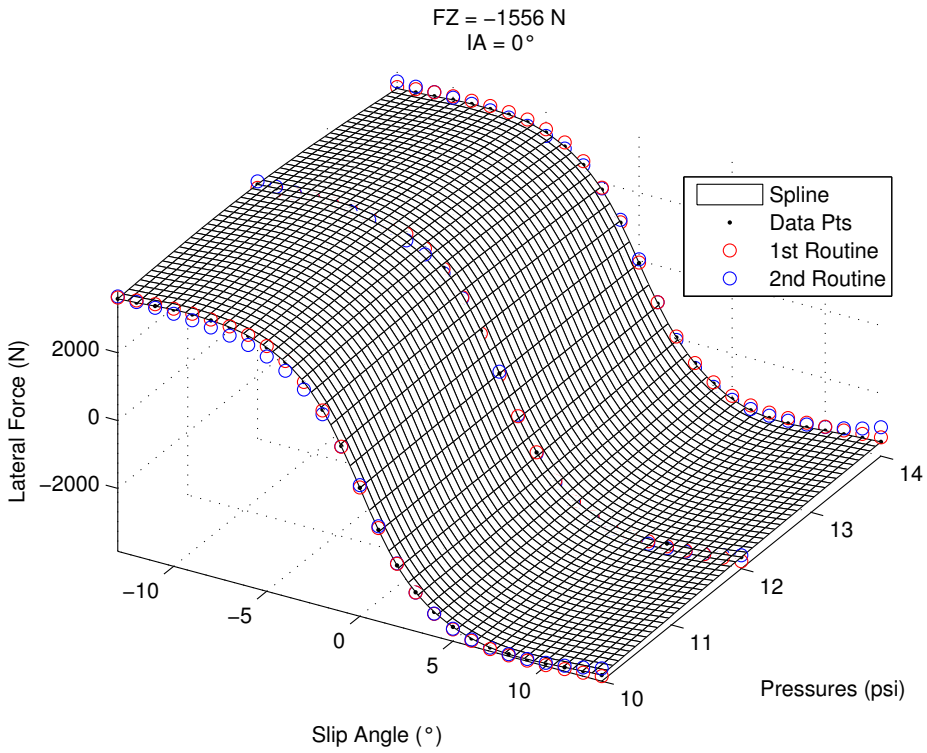


Figure 5.12: Visualizing the effect of optimizing camber coefficients on ability to model pressure characteristics. First routine runs on fixed camber data and Method 2 is used for optimization. Tire operating conditions: Fz=350 lbs, IA=0°

inclination effects.

5.3 Limitation of Split-Optimization

5.3.1 Conventional Behavior

Fixing pressure for the first routine works well for all the tires. The main limitation of this technique is the determination of coefficients optimized in the second routine. Between Tires A, B and C, Tire B shows behavior that is in accordance with the assumed behavior in defining the equations for modeling camber effects. It generates a good fit using the default optimization technique. The $RESNORM$ value is 4.37e+06. Using the split-optimization technique and optimizing all the coefficients in the second routine, a similar value of $RESNORM$ is observed. However, if only the pressure coefficients are optimized, a $RESNORM$ of the order of e+07 is observed. The comparison plots are shown in Figure 5.14a and 5.14b respectively. The poor fit

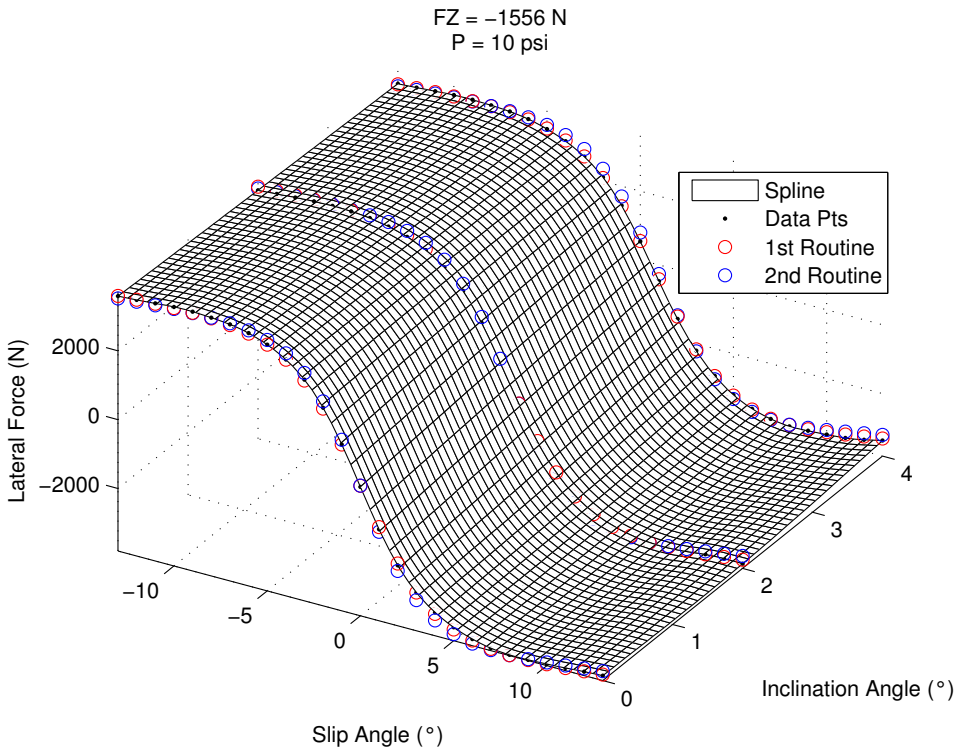
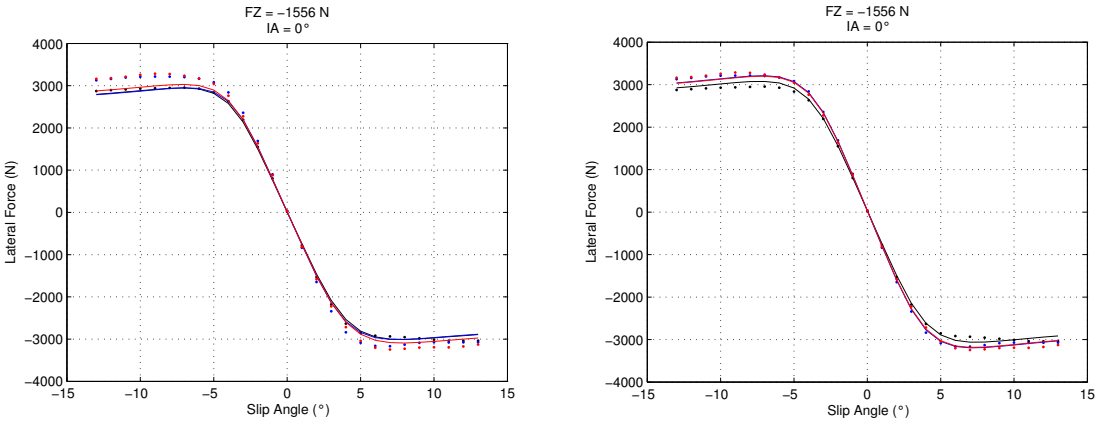


Figure 5.13: Visualizing the effect of optimizing pressure coefficients on ability to model inclination characteristics. First routine runs on fixed pressure data and Method 1 is used for optimization. Tire operating conditions: Fz=350 lbs, P=10 psi

quality with method 1 of split-optimization is seen for a conventional tire. This was found to be true for all tires exhibiting conventional behavior. Thus, for such tires, the split optimization technique works best when the first routine serves as an estimator of good starting coefficients (method 2).

5.3.2 Unconventional Behavior

The method used for the second routine is tire specific in case of unconventional Tires A and C. For Tire A, the second routine resulted in a better fit when only pressure coefficients were optimized. However, for Tire C, the second routine generated a better fit when all the coefficients were optimized. This can be attributed to the fact that the difference in peak lateral force values is less for Tire C and hence, the peak curvature coefficient (E_y) is able to capture the fit once K-coefficients are optimized



(a) Only Pressure Coefficients Optimized (b) All Optimized

Figure 5.14: Limitation of split-optimization to model conventional behavior.

Table 5.1: Comparison of *RESNORM* values for the tires using different methods of optimization.

TIRE	*RESNORM*		
	Full	Split - M1	Split-M2
TIRE A	2.02e+08	6.4e+07	5.4e+07
TIRE B	4.37e+06	1.3e+07	4.83e+06
TIRE C	1.65e+08	1.11e+07	2.83e+06

in the first routine. As a final comparison, the *RESNORM* values for all three tires based on the method used are shown in Table 5.1.

5.4 Modified Model Equation for Pressure Effects

Figures 5.15 and 5.16 show the variation in lateral force characteristics with pressure for three normal loads corresponding to two different tires tested. Figure 5.15 is obtained when Tire B is tested and Figure 5.16 corresponds to Tire A. The phenomenon on the basis of which the $K_{y\alpha}$ equation was modified for pressure is stated in Section 2.3. It is visible in case of Tire B but is reversed for Tire A. For Tire B, at 350 lbs load, greater cornering stiffness is observed when the pressure is 14 psi. At 50 lbs, higher pressure leads to reduced cornering stiffness. Tire A exhibits similar behavior for 50 lbs load. However, for Fz=350 lbs, high pressure does not result in greater

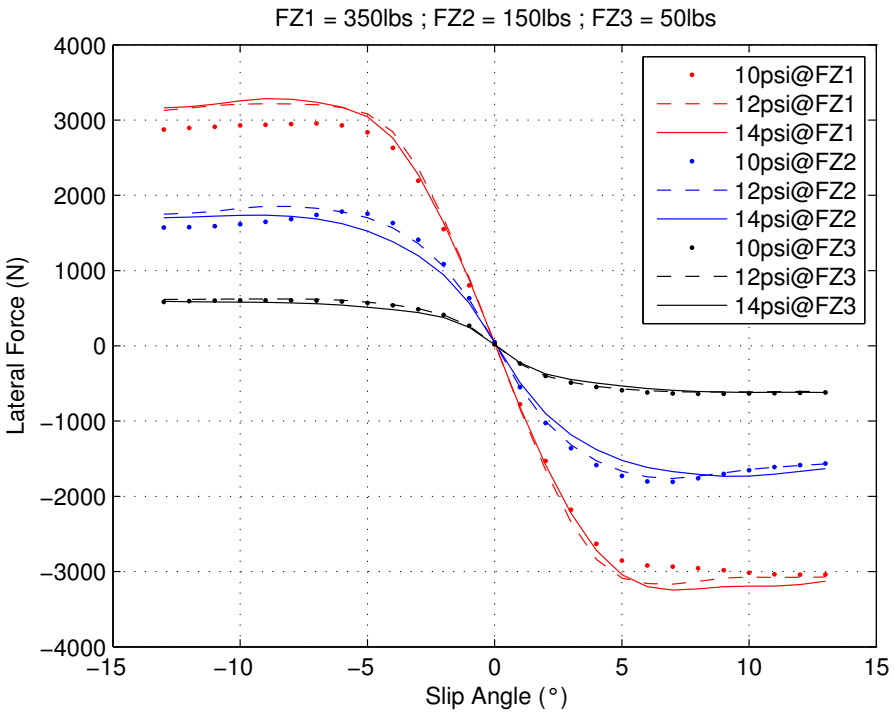


Figure 5.15: Lateral force characteristics for Tire B at three normal loads and three pressures for each load. Tire operating conditions: IA=0°.

cornering stiffness. To verify that this not an error due to the smoothing spline fit, a plot of raw data and spline fit is generated for three different pressures at 350 lbs load. It is shown in Figure 5.17. The raw data shows the same phenomenon. Also, Figure 5.1 in Section 4.5 showed how peak lateral force decreased with increasing camber. A split optimization methodology was developed to tackle the camber anomaly. We will now examine the ability of the model to handle the pressure anomaly.

MATLAB functions $fnder$ and $diff$ are used to obtain cornering stiffness from the data and the fit respectively. We have already seen a 3D plot obtained using *csaps* in Section 4.5. $fnder$ calculates the derivative of the function obtained using *csaps* and stores the resulting surface function in an object. The plot obtained for Tire A is shown in Figure 5.18. It helps visualize the effect of inflation pressure on cornering stiffness for the highest test load of 350 lbs. To calculate cornering stiffness from the fit, we use $diff$ on the lateral force values obtained from the model equations and

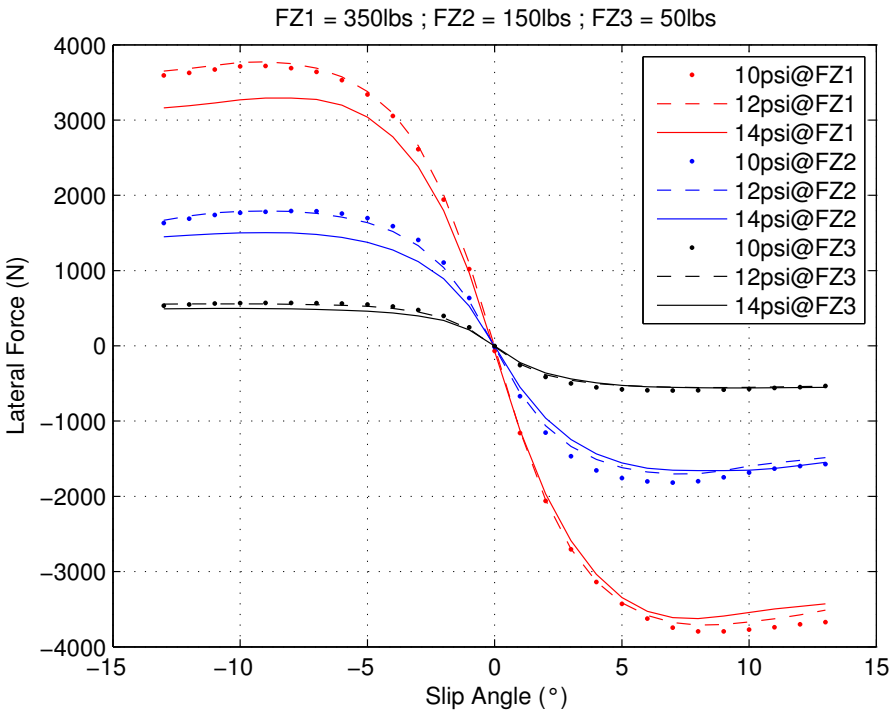


Figure 5.16: Lateral force characteristics for Tire A at three normal loads and three pressures for each load. Tire operating conditions: IA=0°.

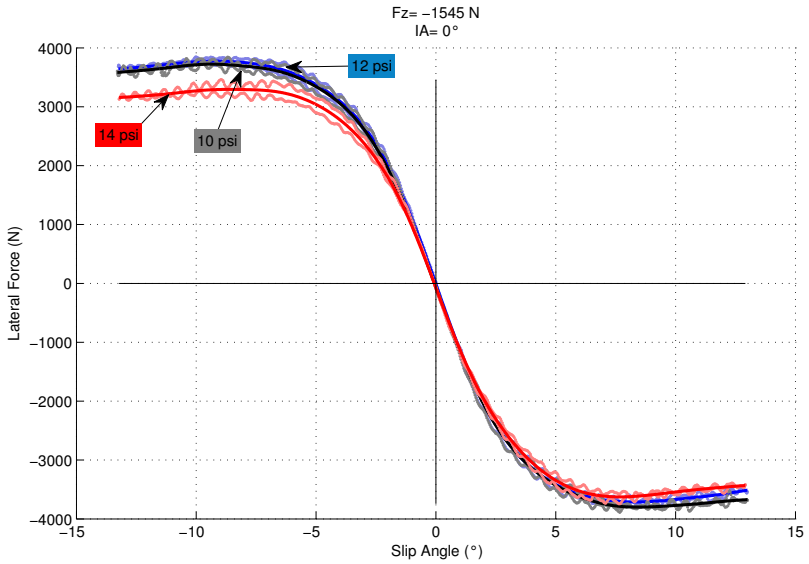


Figure 5.17: Verification of accuracy of spline fit by comparison with raw data for Tire A. Tire operating conditions: Fz=350 lbs, IA=0°.

divide it by the slip angle. However, for this analysis we will use 2D plots of Fy vs. SA for different pressures to judge the prediction of cornering stiffness variation and

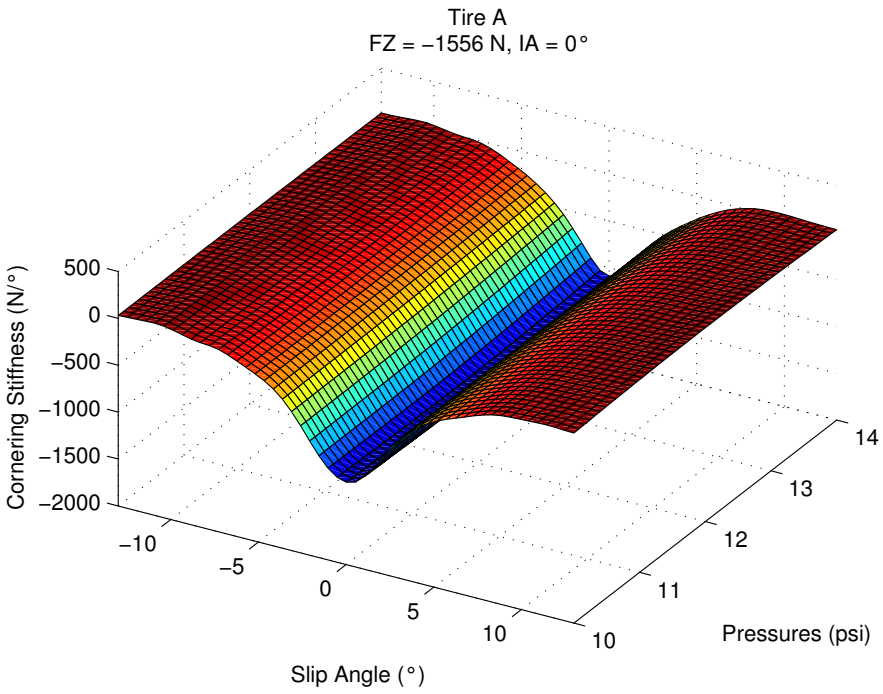
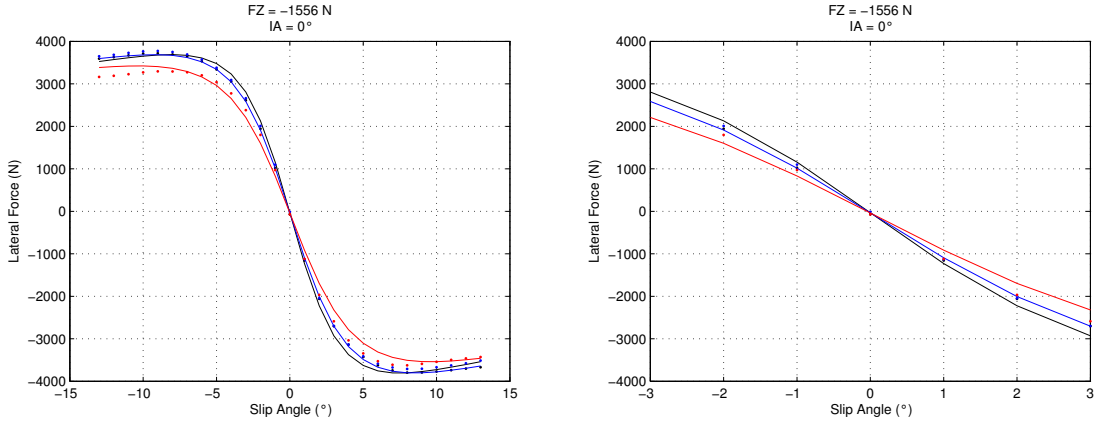


Figure 5.18: Cornering Stiffness surface fit using *fnder*. Tire operating conditions: Fz=350 lbs, IA=0°.



(a) -15° <SA<+15°

(b) -3° <SA<+3°

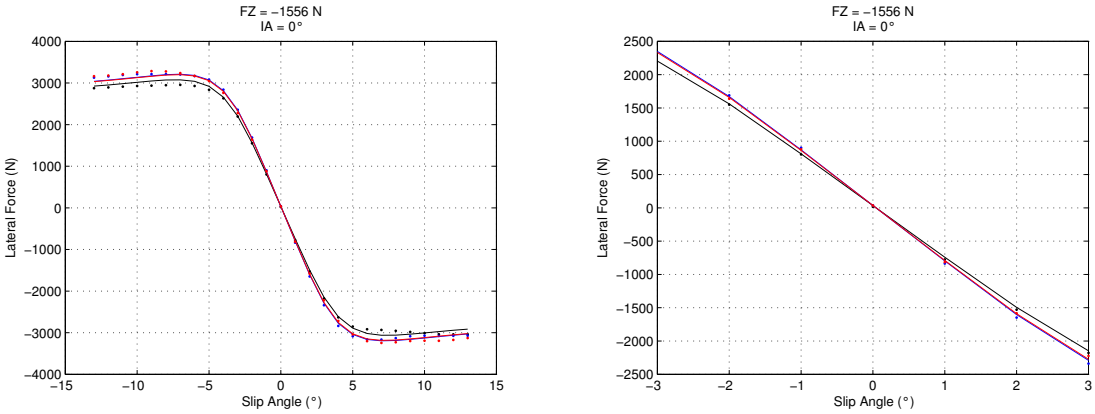
Figure 5.19: Lateral force characteristics for Tire A obtained using the linear definition.

overall quality of the fit simultaneously.

$$
\begin{aligned}
K_{y\alpha} =& p_{Ky1} F'_{z0} (1 + p_{py1} dp_i)(1 - p_{Ky3}|\gamma^*|) \\
& . \sin \left[p_{Ky4} \arctan \left\{ \frac{F_z / F'_{z0}}{(p_{Ky2} + p_{Ky5}\gamma^{*2})(1 + p_{py2} dp_i)} \right\} \right] . \zeta_3 \lambda_{Ky\alpha}
\end{aligned} \tag{5.1}
$$

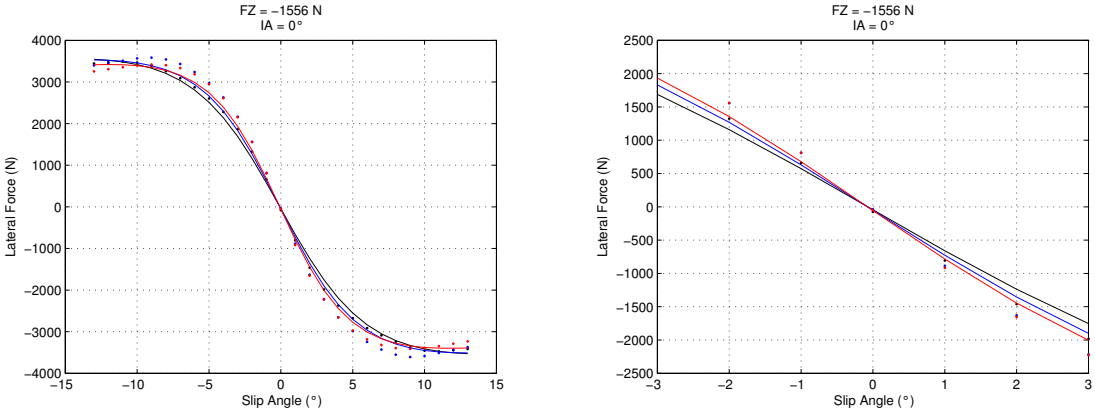
We begin by using the linear relationship seen in Equation 2.17 and re-stated in Equation 5.1 to model change in cornering stiffness with pressure, as we have done in all the fits using the 2012 version so far. The fits obtained for Tire A and Tire B in modeling lateral force are depicted in Figures 5.19a and 5.20a respectively. The plots are zoomed in Figures 5.19b and 5.20b to observe the fit for cornering stiffness. The dots represent data and the curves represent the prediction of the model. The red curve corresponds to 14 psi pressure, the blue curve 12 psi and the black curve represents 10 psi. This will hold true for all figures appearing further in this analysis. It is seen that the model is able to capture the reversed effect of pressure observed in case of Tire A. However, the curves are not close to the actual data. The overall fit, especially, for 14 psi pressure is not accurate as compared to the 10 and 12 psi curves. Tire B shows little variance in cornering stiffness with pressure. Hence, it is not suitable to comment on fit quality. Nevertheless, the fit is able to capture the trend which conforms with that observed by Besselink et al. However, their assumption that unlike peak lateral friction coefficient, cornering stiffness is not tire specific is untrue. Due to the anomaly observed for Tire A, the cornering stiffness equation isn't satisfactorily able to model the variation of cornering stiffness with pressure.

Another tire 'C' is analyzed since Tire B showed little variance of cornering stiffness with pressure. The corresponding plots are included in Figures 5.21a and 5.21b. The cornering stiffness variation is clearly visible and is captured by the model even though the values predicted by the fit do not match the original values. The same is true for the overall fit. The model specially fails to capture the curvature at high values of slip angle. A possible explanation is similar Fy values obtained when the tire is inflated



(a) -15° <SA<+15° (b) -3° <SA<+3°

Figure 5.20: Lateral force characteristics for Tire B obtained using the linear definition.



(a) -15° <SA<+15° (b) -3° <SA<+3°

Figure 5.21: Lateral force characteristics for Tire C obtained using the linear definition.

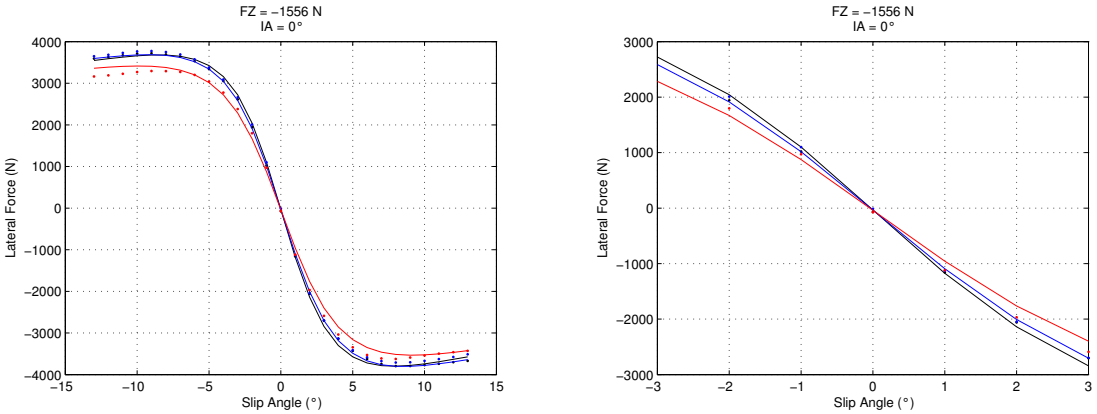
to 12 psi and 14 psi in the low slip angle range, and equal values at higher slip angles for when the tire is at 10 psi and 12 psi.

There is a need to modify the $K_{y\alpha}$ equation, appearing in the 2012 version of the Pacejka model, to be able to capture unexpected variation of cornering stiffness with pressure as seen for Tire A. A new $K_{y\alpha}$ equation is proposed which employs a quadratic polynomial in the denominator of the *arctan* fraction to model variation with inflation pressure. It is as shown in Equation 5.2. Consequently, an additional

coefficient $PPY6$ is defined and its starting value is kept at 0. The fit is now analyzed using this modified form of the 2012 Pacejka model.

$$\begin{aligned} K_{y\alpha} =& p_{Ky1} F'_{z0} (1 + p_{py1} dp_i)(1 - p_{Ky3}|\gamma^*|) \\ & . \sin \left[p_{Ky4} \arctan \left\{ \frac{F_z / F'_{z0}}{(p_{Ky2} + p_{Ky5}\gamma^{*2})(1 + p_{py2} dp_i + p_{py6} dp_i^2)} \right\} \right] . \zeta_3 \lambda_{Ky\alpha} \end{aligned} \tag{5.2}$$

The camber coefficients remain unaffected due to the fact that they are optimized separately from the pressure coefficients. Analysis is done by visual inspection of the fit. The normal load and inclination values are maintained to make the preceding plots comparable. The fit is generated using the new definition of $K_{y\alpha}$. The plots for Tire A are shown in Figures 5.22a and 5.22b.



(a) -15° <SA<+15°

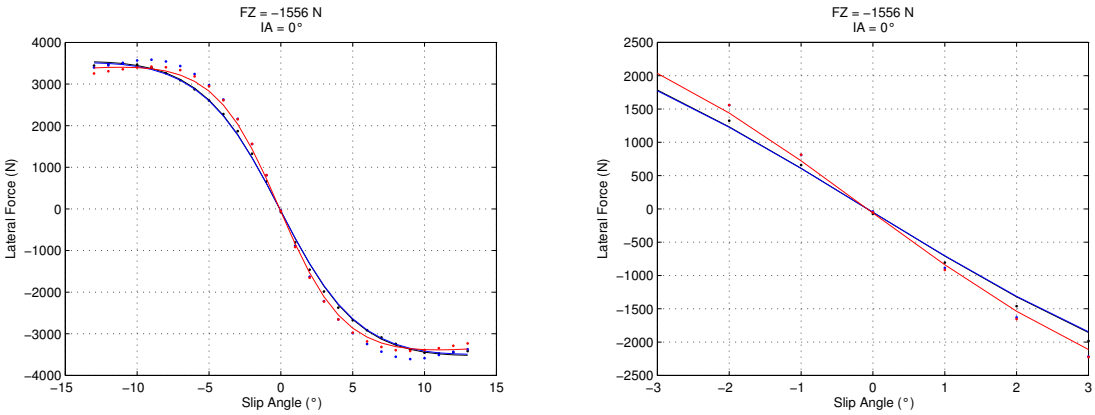
(b) -3° <SA<+3°

Figure 5.22: Lateral force characteristics for Tire A obtained using the quadratic definition.

Primary comparison is made between the zoomed in plots of Tire A to assess the effect on ability to model change in cornering stiffness. The resultant curves not only match the trend but are closer to the data sites. This leads to increased accuracy when the model is relied upon for cornering stiffness calculations. In terms of overall quality of the fit in modeling lateral force variation with pressure, improved curvature is observed near the peaks. As in the case with cornering stiffness, the

curve is able to follow data sites more closely. It has no negative effect on the fit for camber characteristics. If anything, the camber fit improves marginally. The additional pressure coefficient leads to an insignificant increase in computation time.

Similar observations are made for Tire C. The corresponding plots are shown in Figures 5.23a and 5.23b. The most accurate fit is obtained when the tire is at 14 psi inflation pressure. As described earlier, this can be attributed to the data sites intersecting at high slip angles for the tire at 10 psi and 12 psi. Although the overall fit is not accurate, it is still better when compared with the linear definition of $K_{y\alpha}$. It has no effect on fit quality of Tire B which shows the least cornering stiffness variance with pressure.



(a) -15° <SA<+15°

(b) -3° <SA<+3°

Figure 5.23: Lateral force characteristics for Tire C obtained using the quadratic definition.

The assumption that at high loads, cornering stiffness increases with increase in pressure does not hold true for all tires as observed for Tire A. It is a tire dependent property like peak lateral friction coefficient. Consequently, the assumption that a linear polynomial is able to model this characteristic is false. Tire A also exhibits anomaly with respect to camber. A split optimization technique was used to generate a better fit for camber. For the pressure anomaly, a better fit can be obtained by using a quadratic polynomial to model variation in cornering stiffness. Using these techniques,

the Pacejka model can be successfully applied in case of observed unconventional tire behavior with inputs.

The code required to model steady-state lateral force and aligning moment, under pure slip conditions, using the modified 2012 version of the Pacejka model is included in the Appendix. It is defined to run with the data obtained from round 6 of testing but can be easily changed to use data from other rounds. The code for other characteristics can then be developed using definitions of the equations from [2].

CHAPTER 6: CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusion

The aim of this thesis was to provide an introduction to tire modeling using the Pacejka model and a method to enhance its capability in handling conventional and unconventional tire behavior. The research began with the code obtained from the FSAE TTC website. The code was developed by Bill Cobb for the steady-state, pure side-slip MF5.2(2009) version of the Pacejka model. Analysis was done to investigate specific aspects of this code. The input controllers and output measurement sensors introduce fluctuations in the raw data. The preprocessing analysis included effect of averaging Normal Load values for a sweep and normalizing the error. It was found that this has little effect on smoothing the data. With availability of controller and sensor performance parameters such as accuracy, resolution and logging rates, this process can be greatly improved. Then, processing of Aligning Moment outliers present in the data due to the inertia of the machine when the tire changes direction at maximum slip angle was investigated for degree of polynomial of the fit. It is dependent on the tire under investigation and can be decided by generating comparison plots as shown in its analysis. In cases where it cannot be done for whatever reason, a quadratic polynomial fit is a reliable option. The final step in preprocessing analysis consisted of effect of tension of the cubic smoothing spline on its ability to capture different tire characteristics accurately. Again, the final answer is tire and characteristic specific. It is the most crucial step as the cubic smoothing spline does majority of the smoothing and hence, time and effort must be invested to ensure the selected tension of the fit is suitable to capture required tire characteristics.

The fitting routine was examined with respect to its sensitivity to initial guesses of the coefficients. To aid investigation in case of failure of the fitting routine, it was decided to use the 2002 and 2012 versions of the Pacejka model as its literature was readily available. Based on the observations, best performance of the fitter was obtained when initial guesses of the coefficients are set as per Appendix 3 of [2]. However, the 2012 Pacejka model failed to capture unconventional camber and pressure effects pertaining to Tire A and Tire C. The effect of different algorithms available to be used in the non-linear least squares curve fitting routine was examined. The default algorithm "Trust-Region-Reflective" was found to give the best results. Before dealing with the pressure anomaly, the impact of bootstrapping was evaluated. It was found that subtracting the small random number doesn't really have an impact on the fitter to deflect to another minimum. Hence, it was omitted from the code. The non-linear least square curve fit function was found to exhibit good reproducibility. The effect of number of iterations is case specific and should be decided by observing trends of coefficients and norm of the residuals. The final number of iterations should be low enough to save computation time but high enough to ensure repeatability of the fitting routine.

To deal with unconventional camber effects, a split optimization technique was employed. It involved keeping one input constant while optimizing coefficients for the others. This is followed by tire-specific optimization of the remaining coefficients over the complete data. Analysis was done to evaluate selection of fixed input. Keeping the pressure constant resulted in a better overall fit quality at the end of all optimizations. For Tire A, all combinations resulted in a similar value of norm of the residual. In such a scenario, visual analysis of the fit is required. It was found that for Tire A, it was better to optimize only the camber coefficients in the second routine. Tire C norm of the residuals showed a clear advantage of optimizing all coefficients together. Further explanation was provided based on the effect on coefficient values and visual

analysis of the fit. The split-optimization philosophy was evaluated for conventional tires and it was found that the fit obtained was as accurate as that using the default full optimization method provided the coefficient values obtained from the first routine act as starting coefficients to the second routine. The time taken by to run both the routines was less than that by default full optimization routine for Tires A and C. Thus, split-optimization proved reliable for all data.

As per [8], the effect of pressure on cornering stiffness is tire independent. This was found to be untrue as with the case of Tire A, the opposite phenomena was observed. Hence, the assumption that a linear equation is sufficient to capture variation of cornering stiffness with pressure failed. Consequently, a modification to the 2012 Pacejka model was proposed which implemented a quadratic equation in the arctan fraction of the cornering stiffness equation. A visibly better fit was obtained for both cornering stiffness effect and overall fit for lateral force in pure cornering conditions. The quadratic definition also improved the fit for tires exhibiting conventional characteristics with inflation pressure. It had no negative effect on the ability of fit to capture camber characteristics. Due to the limited availability of corresponding data, this modification could not be evaluated for more tires. However, the findings within the limited data sets are encouraging enough to expect high fidelity from the modified model.

6.2 Future Scope

The work presented in this thesis can serve as a platform to dive into advanced topics in empirical and semi-empirical tire modeling. For instance, the code provided in the Appendix can be extended to model transient response of the tire using the SWIFT model defined in [2]. With availability of performance paramters of the controllers and sensors of the test machine, frequency analysis can be done to examine impact of each sensor/controller on the fluctuations observed in the data. Extending from findings of this research, the modified version of 2012 Pacejka model can be ap-

plied to non-FSAE tires exhibiting unconventional phenomena. It will be interesting to asses its performance at higher values of Normal Load. With available information on starting coefficients, comparisons can be made between fits generated using the split optimization method and the normal method used otherwise. There is also potential to modify equations that model camber and pressure effects to be able to capture unconventional characteristics in a qualitatively better manner.

BIBLIOGRAPHY

[1] Erdogan, G. "Tire Modeling - Lateral and Longitudinal Forces" (2009). (http://www.menet.umn.edu/ gurkan/Tire%20Modeling%20%20Lecture.pdf)

[2] Pacejka, H. B., and Besselink, I. J. M. Tire and vehicle dynamics. Elsevier (2012).

[3] Fiala, E. "Lateral forces on rolling pneumatic tires." Zeitschrift VDI 96, no. 29 (1954): 973-979.

[4] Milliken, W. F., and Milliken, D. L. Race car vehicle dynamics. Vol. 400. Warrendale: Society of Automotive Engineers (1995).

[5] Radt Jr, H. S., and Milliken W. F. "Non-dimensionalizing tyre data for vehicle simulation." In Road Vehicle Handling, I Mech E Conference Publications 1983-5. Sponsored by Automobile Division of the Institution of Mechanical Engineers under patronage of Federation Internationale des Societies d'Ingenieurs des Techniques de l'Automobile (FISITA) he, no. C133/83. 1983.

[6] Bayle, P., Forissier, J. F. and Lafon, S. "A new tyre model for vehicle dynamics simulations." Automotive Technology International 93 (1993): 193-198.

[7] Pacejka, Hans. Tire and vehicle dynamics. Elsevier (2002).

[8] Schmeitz, A. J. C.,Besselink, I. J. M., De Hoogh, J., and Nijmeijer, H. "Extending the Magic Formula and SWIFT tyre models for inflation pressure changes." VDI BERICHTE 1912 (2005): 201.

[9] Cobb, W. "Pacejka MF5.2 fitting of TTC Data" (2008). (http://sae.wsu.edu/ttc/viewtopic.php?f=17&t=25)

[10] Cobb, W. "Matlab Processing of FSAE TTC Tire Test Data" (2009). (http://sae.wsu.edu/ttc/viewtopic.php?f=17&t=23)

[11] van Oosten, J. JM., and Bakker,E. "Determination of magic tyre model parameters." Vehicle System Dynamics 21, no. S1 (1992): 19-29.

[12] Betzler, K. "Fitting in Matlab." Fachbereich Physik, University at Osnabruck.

[13] Berghen, F. V. "Levenberg-Marquardt algorithms vs Trust Region algorithms." IRIDIA, Universit Libre de Bruxelles (2004).

APPENDIX: MATLAB CODE FOR THE MODIFIED PACEJKA MODEL

```
% Adapted from code provided by Bill Cobb (william.a.cobb@gm.com).


clc
clear all;
global FZ0 pi0 R0

global PCY1 PDY1 PDY2 PDY3 ...
    PEY1 PEY2 PEY3 PEY4 PEY5 ...
    PKY1 PKY2 PKY3 PKY4 PKY5 PKY6 PKY7...
    PHY1 PHY2  ...
    PVY1 PVY2 PVY3 PVY4...
    PPY1 PPY2 PPY3 PPY4 PPY5;

global LFZO LCX LMUX LEX LKX  LHX LVX LCY LMUY LEY LKYA LKYG LHY LVY
    LGAY LTR LRES LGAZ LXAL LYKA LVYKA LS LSGKP LSGAL LGYR LDMUY
    LDMUX EPSK LKZG

clc
%% Importing

%%TTC3
% [filename pathname]= uigetfile('*.dat','Enter TIRF Test File',
'Enter path here')
%
% t=importdata([pathname filename]);
% names  = t.textdata{2}
% nchans = size(t.data,2)    % how much we got?
%
% t.data(1:1750,:)=[];    % toss out the 1st 1750 pts
%(different load/warmup)
% for n=1:nchans          % demultiplex
%     [name,names]=strtok(names);


%     eval([upper(name) '= t.data(:,' num2str(n) ');']);
% end
%-----------------------------------------------
%%TTC6
[filename pathname]= uigetfile('*.mat','Enter TIRF Test File',
'Enter path here')
```

```
t=importdata([pathname filename]);

%dynamically create and assign variables from fieldnames
fnames  = fieldnames(t);

for i=5:length(fnames)
    %eval(['t.' fnames{i} '(1:1317)=[];']) %delete warm-up points
    %no warmup points for run 12.
    eval([fnames{i} '=t.' fnames{i} ';'])
end

%% Pre-processing and Visualising

m=1:length(SA);     % point counter
%sp=spline(m,SA+3.5); % TTC3
sp = spline(m,SA); %TTC6 sweeps start at 0 SA
z=fnzeros(sp);      % location of zero crossings
z=round(z(1,:));   % no dups and integer indices
z = [1 z 61323]; %additional reqd for B1654run12

figure('Name','Locations of Test Slip Sweep','NumberTitle','Off')
plot(m,SA,'r')
hold on
plot(z,zeros(length(z)),'ko')
% % xlim([0 3200])
% deleting the error points due to spline
%Technique for combined slip data.
%Resort to drop kick if this doesn't work
% j=0;
% for i=2:length(z)-1
%     if((z(i)-z(i-1)<=150))
%         if((z(i+1)-z(i)<=150))
%             j=j+1;
%             temp(1,j)=i;
%         end
%     end
% end
% z(temp)=[];

%Technique for TTC6 pure cornering
z(2:2:length(z))=[];
plot(z,zeros(length(z)),'bo')
line([0 m(end)], [0 0],'color','k')
xlabel('Point Count')
```

```
ylabel('Slip Angle')
legend('Test Data','Computed Slip Points of Interest'),legend Boxoff
clear fmdata fmdatafull
q=0;



%% Smoothening and Spline Fitting
%for n=1:3:length(z)-1 % TTC3
for n=1:1:length(z)-1   %TTC6 Data files dont have shutoff.
%     %TTC3
%     sa=SA(z(n):z(n+2));
%     fz=FZ(z(n):z(n+2));
%     fy=FY(z(n):z(n+2));
%     mz=MZ(z(n):z(n+2));
%     mx=MX(z(n):z(n+2));
%     rl=RL(z(n):z(n+2));
%     pi=P(z(n):z(n+2)).*0.145;%storing in psi
%     ia=IA(z(n):z(n+2));

    %TTC6
    sa=SA(z(n):z(n+1));
    fz=FZ(z(n):z(n+1));
    fy=FY(z(n):z(n+1));
    mz=MZ(z(n):z(n+1));
    ia=round((IA(z(n):z(n+1))));
    pi=P(z(n):z(n+1)).*0.145;%storing in psi
    et=ET(z(n):z(n+1));
    %%
%  %--- plot to see the effect of mean FZ
%     figure('Name','Reduced error by mean FZ - A comparison',
'NumberTitle','off')
%     subplot(2,1,1)
%     plot(sa,fy,'k')
%     title('SA vs. Fy')
%     hold on
%     subplot(2,1,2)
%     plot(sa,mz,'k')
%     title('SA vs. Mz')
%     hold on
    %%
 %------ reduce FZ error by normalizing with Fz:
    fy= round(mean(fz))*ones(length(fz),1).*fy./fz;
    mz= round(mean(fz))*ones(length(fz),1).*mz./fz;
```

```
   % mx= round(mean(fz))*ones(length(fz),1).*mx./fz;
    fz= round(mean(fz)).*ones(length(fz),1);
    pi= round(mean(pi)).*ones(length(fz),1);
%     mz_org_rng=mz;
    %%
%  %--- plot to see the effect of mean FZ
% %figure('Name','Reduced error by mean FZ - A comparison',
'NumberTitle','off')
%     subplot(2,1,1)
%     plot(sa,fy,'r-')
%     subplot(2,1,2)
%     plot(sa,mz,'r-')
%     hold off
    %%

    [tmp,imn]=min(sa);
    [tmp,imx]=max(sa);
    p=1:length(sa);
    rng=imx-50:imx+50;
    pp=polyfit(p(rng),mz(rng)',2);
    mzf=polyval(pp,p(rng));

ind=find(abs(mzf-mz(rng)') > 7);
    mz(rng(ind))=mzf(ind);
    %%
% %trial for excludedata
% outliers = excludedata(p(rng),mzf,'indices',ind);
% %comments : excludedata makes a logical array with 1 at outliers and 0
% %at other points. On comparing with ind statement above, the result is
% %the same.
%
%     %visualizing the outliers
%     figure('Name','Polyfit to max MZ for outliers','NumberTitle','off')
%     plot(sa(rng),mz(rng),'r.')
%     xlabel('Slip Angle ()')
%     ylabel('Aligning Moment (N-m)')
%     hold on
%     plot (sa(rng),mzf)
%     hold on
%
%     mzmx_outliers=zeros(1,length(rng));
%     mzmx_outliers(ind)=mz_org_rng(rng(ind));
%     mzmx_outliers(mzmx_outliers==0)=NaN;
%     plot(sa(rng),mzmx_outliers,'k.')
%     legend('Processed Data','Quadratic polynomial fit',
```

```
'Outliers','Location','southwest')
%     grid on
%     hold off

    %%
    rng=imn-50:imn+50;
    pp=polyfit(p(rng),mz(rng)',2);
    mzf=polyval(pp,p(rng));

    ind=find(abs(mzf-mz(rng)') > 7);
    mz(rng(ind))=mzf(ind);

%      %visualizing the outliers
%     figure('Name','Polyfit to min MZ for outliers','NumberTitle','off')
%     plot(sa(rng),mz(rng),'ro')
%     xlabel('SA')
%     ylabel('MZ')
%     hold on
%     plot (sa(rng),mzf)
%     hold on
%
%     mzmn_outliers=zeros(1,length(rng));
%     mzmn_outliers(ind)=mz_org_rng(rng(ind));
%     mzmn_outliers(mzmn_outliers==0)=NaN;
%     plot(sa(rng),mzmn_outliers,'k*')
%     hold off

%%

sp_fy=csaps(sa,fy,.1);
sp_mz=csaps(sa,mz,.1);
%sp_mx=csaps(sa,mx,.1);
%sp_rl=csaps(sa,rl,.1);

% %%  Analysing the effect of changing P
% figure('Name','Effect of Changing P in csaps','NumberTitle','off')
% subplot(2,2,1)
% plot(sa,mz,'r.')
% grid on
% hold on
% sp_mz=csaps(sa,mz,.1);
% fnplt(sp_mz)
% title('P=0.1')
% xlabel('Slip Angle')
% ylabel('Aligning Moment')
```

```
% legend('Test Data','Fitted Data')
% subplot(2,2,2)
% plot(sa,mz,'r.')
% grid on
% hold on
% sp_mz=csaps(sa,mz,.5);
% fnplt(sp_mz)
% title('P=0.5')
% xlabel('Slip Angle')
% ylabel('Aligning Moment')
% legend('Test Data','Fitted Data')
% subplot(2,2,3)
% plot(sa,mz,'r.')
% grid on
% hold on
% sp_mz=csaps(sa,mz,.75);
% fnplt(sp_mz)
% title('P=0.75')
% xlabel('Slip Angle')
% ylabel('Aligning Moment')
% legend('Test Data','Fitted Data')
% subplot(2,2,4)
% plot(sa,mz,'r.')
% grid on
% hold on
% sp_mz=csaps(sa,mz,1);
% fnplt(sp_mz)
% title('P=1')
% xlabel('Slip Angle')
% ylabel('Aligning Moment')
% legend('Test Data','Fitted Data')
% hold off


    for sl=floor(min(sa)):1:ceil(max(sa))
        q=q+1;
        fmdata(q,1)=sl;
        %mean loads differ with sweep. Setting them to unique values
        if(fz(end)<-1500)
            fmdata(q,2)=-1556;

        else if((-1500<fz(end))&&(fz(end)<-1000))
            fmdata(q,2)=-1112;

            else if((-1000<fz(end))&& (fz(end)<-500))
```

```
            fmdata(q,2)=-667;

                else if((-500<fz(end))&&(fz(end)<-250))
            fmdata(q,2)=-444;

                    else if((-250<fz(end))&&(fz(end)<-150))
            fmdata(q,2)=-222;
                        end
                    end
                end
            end
        end
%       fmdata(q,3)=round(mean(fz)); %TTC3
        fmdata(q,3)=round(mean(ia));
        fmdata(q,4)=round(mean(pi));
        fmdata(q,5)=fnval(sp_fy,sl);
        fmdata(q,6)=fnval(sp_mz,sl);
    end
end
inx_slip14=find(abs(fmdata(:,1))==14);
fmdata(inx_slip14,:)=[];%removing slip points to satisfy reshape

%Restricted set fitting - Pressure
fmdatafull = fmdata;
fmdatafull = sortrows(fmdatafull,[4,1,2,3]);
inx_p10 = (fmdatafull(:,4)==10);
fmdata = fmdatafull(inx_p10,:);
%Restricted set fitting

fmdata  = sortrows(fmdata,[4,1,2,3]);

t.SA= fmdata(:,1);
t.P= fmdata(:,4);
t.FZ= fmdata(:,2);
t.FY= fmdata(:,5);
t.MZ= fmdata(:,6);
% t.MX= fmdata(:,6);
t.IA= fmdata(:,3);
% %-------------------------------------------------------------------------
%% Initializing the MF parameters

FZ0=  abs(mean(t.FZ)) % = FNOMIN = 'nominal wheel load'
pi0= mean(t.P)% = 'nominal tire pressure'
R0  = .240
```

```
AMU=10;
LFZO                    = 0.100000E+01          ;
LCX                     = 0.100000E+01          ;
LMUX                    = 0.100000E+01          ;
LEX                     = 0.100000E+01          ;
LKX                     = 0.100000E+01          ;
LHX                     = 0.100000E+01          ;
LVX                     = 0.100000E+01          ;
LCY                     = 0.100000E+01          ;
LMUY                    = 0.100000E+01          ;
LDMUY = AMU*LMUY/(1+(AMU-1)*LMUY);
LDMUX = AMU*LMUX/(1+(AMU-1)*LMUX);
LEY                     = 0.100000E+01          ;
LKYA                    = 0.100000E+01          ;
LKYG                    = 0.100000E+01          ;
LHY                     = 0.100000E+01          ;
LVY                     = 0.100000E+01          ;
LGAY                    = 0.100000E+01          ;
LTR                     = 0.100000E+01          ;
LRES                    = 0.100000E+01          ;
LGAZ                    = 0.100000E+01          ;
LXAL                    = 0.100000E+01          ;
LYKA                    = 0.100000E+01          ;
LVYKA                   = 0.100000E+01          ;
LS                      = 0.100000E+01          ;
LSGKP                   = 0.100000E+01          ;
LSGAL                   = 0.100000E+01          ;
LGYR                    = 0.100000E+01          ;
EPSK                    = 0.100000E+01          ;
LKZG                    = 0.100000E+01          ;

%[LATERAL_COEFFICIENTS]
%As per Appendix 3 of Tyre and VD - 3rd Edition, pg. 615
PCY1 =  1.3;
PDY1 = 0;
PDY2 = -0.05;
PDY3 = 0;
PEY1 = 0;
PEY2 = 0;
PEY3 = 0;
PEY4 = 0;
PEY5 = 0;
PKY1 = -50;
PKY2 = 2;
```

```
PKY3 = 0;
PKY4 = 0;%0
PKY5 = 0;
PKY6 = 0;
PKY7 = 0;
PHY1 = 0;
PHY2 = 0;
PVY1 = 0;
PVY2 = 0;
PVY3 = 0;
PVY4 = 0;
PPY1 = 0;
PPY2 = 0;
PPY3 = 0;
PPY4 = 0;
PPY5 = 0;

%% Fitting FY for pure side slip
%Restricted set fitting

INPUT = [t.SA,t.FZ,t.P,t.IA]; % slip, vert, P,incl
clear AA RESNORM
A_str ={'PCY1' 'PDY1' 'PDY2' 'PDY3' 'PEY1' 'PEY2' 'PEY3' 'PEY4' 'PEY5'
    'PKY1' 'PKY2' 'PKY3' 'PKY4' 'PKY5' 'PKY6' 'PKY7' 'PHY1' 'PHY2' ...
    'PVY1' 'PVY2' 'PVY3' 'PVY4'...
    'PPY1' 'PPY2' 'PPY3' 'PPY4' 'PPY5'};
A_old =[PCY1 PDY1 PDY2 PDY3 PEY1 PEY2 PEY3 PEY4 PEY5
    PKY1 PKY2 PKY3 PKY4 PKY5 PKY6 PKY7 PHY1 PHY2 ...
    PVY1 PVY2 PVY3 PVY4...
    PPY1 PPY2 PPY3 PPY4 PPY5];

options =optimset('MaxFunEvals',20000,'MaxIter',20000,'Display','final',
    'TolX',1e-7,'TolFun',1e-7);%,'Algorithm','levenberg-marquardt');

fig1=figure('Name',['Pacejka_12  FY Fitting Results'],
'Position',[2 2 1600 1180],'NumberTitle','off');
for k=1:20
    [A,RESNORM(k),RESIDUAL(:,k),EXITFLAG] =
     lsqcurvefit('Pac12_Fy',A_old,INPUT,t.FY,[],[],options);
        AA(:,k)=A;
        for n=1:27
        subplot(3,9,n)
  bar([AA(n,:)],'group')
        title(['A(' num2str(n) ') = ' A_str{n}],'FontSize',8)
        end
```

```
    for n=1:27  % update A coefficients to newest values
    %   disp(['A_old(' num2str(n) ') = ' num2str(A_old(n)) ';'
     'A(' num2str(n) ') = ' num2str(A(n)) ';'])
        eval(['A_old(' num2str(n) ') = ' num2str(A(n)) ';' ])
    end
    set(fig1,'Name',[filename '    Restricted Pressure Fit: Pac12 - Fy
    Iteration: ' num2str(k)  '    RESNORM: ' num2str(RESNORM(k))])
    drawnow
end

%% Comparison Plots for SA vs. IA vs. FY

pressures   = unique(round(fmdata(:,4)))'
npressures  = length(pressures)

slips   =  unique(round(fmdata(:,1)))'
nslips  = length(slips)

cambers = unique(fmdata(:,3))'
ncambers = length(cambers)

FZreqd = -1556;%Load-->__N
inxP    = find((fmdata(:,2)==FZreqd));
fmdataP = fmdata(inxP,:); %fmdata related to one load
fmdataP = sortrows(fmdataP,[3,1]);


% 2-D plot for effect of IA on Fy - Data
d=ncambers;
l=length(fmdataP);
figure('Name',['Effect of IA on Fy - data - ' filename ],
'NumberTitle','Off')
plot(fmdataP(1:l/d,1),fmdataP(1:l/d,5),'r')
hold on
plot(fmdataP((l/d+1):(2*l/d),1),fmdataP((l/d+1):(2*l/d),5),'k')
plot(fmdataP((2*l/d+1):(3*l/d),1),fmdataP((2*l/d+1):(3*l/d),5),'b')
legend('0','2','4')
grid on
hold off

%% 3-D plot for SA vs. IA vs. FY - Data, Spline and Fit

fmdataP = sortrows(fmdataP,[1,2]); %sort for reshape
fy0     = reshape(fmdataP(:,5),ncambers,nslips)';
```

```
pace12fy=Pac12_Fy(A,INPUT);
figure('Name','Restricted Pressure Fit : Pac12 - SA vs IA vs Fy',
'NumberTitle','off');%'MenuBar','none'
hold on
fnplt(csaps({slips,cambers},fy0,0.9999))
plot3(INPUT(inxP,1),INPUT(inxP,4),t.FY(inxP),'k.')
plot3(INPUT(inxP,1),INPUT(inxP,4),pace12fy(inxP),'ro');
view(30,45)
title({['FZ = ' num2str(FZreqd) ' N'];
['P = ' num2str(pressures) ' psi']})
xlabel('Slip Angle ()')
ylabel('Inclination Angle ()')
zlabel('Lateral Force (N)')
legend('WAC Spline','Data Pts','Pac12')
colormap(white)
legend('Spline','Data Pts','Pac12-P Restricted')
hold off
%% 3-D Comparison Plots for SA vs. FZ vs. FY
preqd = 16;% Pressure-->__psi
inx0    = find(fmdata(:,2) == preqd);
fmdata0 = fmdata(inx0,:);
loads   = mean(reshape(fmdata0(:,3),[],nslips),2)'
nloads  = length(loads)
fy0     = reshape(fmdata0(:,4),nloads,nslips)';

figure('Name','A Fitting Comparison for Pac-12 :FY',
'NumberTitle','off');%'MenuBar','none'
hold on
fnplt(csaps({slips,loads},fy0,.9999))
plot3(INPUT(inx0,1),INPUT(inx0,2),t.FY(inx0),'k.')
plot3(INPUT(inx0,1),INPUT(inx0,2),pace12fy(inx0),'ro')
view(30,45)
title(['P = ' num2str(preqd) 'psi'])
xlabel('Slip Angle')
ylabel('Vertical Load (N)')
zlabel('Lateral Force')
legend('WAC Spline','Data Pts','Pac12')
colormap(white)

%Update to best-fit obtained from restricted data fitting
PCY1    = A(1);
PDY1    = A(2);
PDY2    = A(3);
```

```
PDY3    = A(4);
PEY1    = A(5);
PEY2    = A(6);
PEY3    = A(7);
PEY4    = A(8);
PEY5    = A(9);
PKY1    = A(10);
PKY2    = A(11);
PKY3    = A(12);
PKY4    = A(13);
PKY5    = A(14);
PKY6    = A(15);
PKY7    = A(16);
PHY1    = A(17);
PHY2    = A(18);
PVY1    = A(19);
PVY2    = A(20);
PVY3    = A(21);
PVY4    = A(22);
PPY1    = A(23);
PPY2    = A(24);
PPY3    = A(25);
PPY4    = A(26);
PPY5    = A(27);

%% Fitting FY for pure side slip


%Full set fit.
%Update t to have full data
t.SA= fmdatafull(:,1);
t.P= fmdatafull(:,4);
t.FZ= fmdatafull(:,2);
t.FY= fmdatafull(:,5);
t.MZ= fmdatafull(:,6);
% t.MX= fmdata(:,6);
t.IA= fmdatafull(:,3);

FZ0=  abs(mean(t.FZ)) % = FNOMIN = 'nominal wheel load'
pi0= mean(t.P)% = 'nominal tire pressure'

INPUT = [t.SA,t.FZ,t.P,t.IA]; % slip, vert, P,incl
clear AAf RESNORMf

%METHOD 1
```

```
Af_str ={'PPY1' 'PPY2' 'PPY3' 'PPY4' 'PPY5'};
Af_old =[PPY1 PPY2 PPY3 PPY4 PPY5];

%METHOD 2
% Af_str ={'PCY1' 'PDY1' 'PDY2' 'PDY3' 'PEY1' 'PEY2' 'PEY3' 'PEY4'
%   'PEY5' 'PKY1' 'PKY2' 'PKY3' 'PKY4' 'PKY5' 'PKY6' 'PKY7' 'PHY1'
%    'PHY2' 'PVY1' 'PVY2' 'PVY3' 'PVY4'...
%     'PPY1' 'PPY2' 'PPY3' 'PPY4' 'PPY5'};
% Af_old =[PCY1 PDY1 PDY2 PDY3 PEY1 PEY2 PEY3 PEY4 PEY5
%     PKY1 PKY2 PKY3 PKY4 PKY5 PKY6 PKY7 PHY1 PHY2 ...
%     PVY1 PVY2 PVY3 PVY4...
%     PPY1 PPY2 PPY3 PPY4 PPY5];

options =optimset('MaxFunEvals',20000,'MaxIter',20000,'Display','final',
    'TolX',1e-7,'TolFun',1e-7);%,'Algorithm','levenberg-marquardt');

fig1f=figure('Name',['Pacejka_12  FY Fitting Results'],
'Position',[2 2 1600 1180],'NumberTitle','off');
for k=1:20
    [Af,RESNORMf(k),RESIDUALf(:,k),EXITFLAGf] =
     lsqcurvefit('Pac12_Fy_Prestricted',Af_old,INPUT,t.FY,[],[],options);
        AAf(:,k)=Af;
        for n=1:5
        subplot(2,3,n)
        bar([AAf(n,:)],'group')
        title(['Af(' num2str(n) ') = ' Af_str{n}],'FontSize',8)
        end

    for n=1:5  % update A coefficients to newest values
      disp(['Af_old(' num2str(n) ') = ' num2str(Af_old(n)) ';
          ' 'Af(' num2str(n) ') = ' num2str(Af(n)) ';'])
      eval(['Af_old(' num2str(n) ') = ' num2str(Af(n)) ';' ])
    end
    set(fig1f,'Name',[filename '
    Only Pressure Coeffs Optimized : Pac12 - Fy      Iteration:
    ' num2str(k)  ' RESNORMf: ' num2str(RESNORMf(k)) ])
    drawnow
end

%% Comparison Plots for SA vs. P vs. FY
pressuresf   = unique(round(fmdatafull(:,4)))'
npressuresf  = length(pressuresf)

slipsf   =  unique(round(fmdatafull(:,1)))'
nslipsf  = length(slipsf)
```

```
cambersf = unique(fmdatafull(:,3))'
ncambersf = length(cambersf)

FZreqd = -1556;%Load-->__N
Creqd = 0;
inxPf    = find((fmdatafull(:,2)==FZreqd) & (fmdatafull(:,3)==Creqd));
fmdataPf = fmdatafull(inxPf,:); %fmdata related to one load and camber

% 2-D plot for effect of P on Fy - Data
d=npressures;
l=length(fmdataP);
figure('Name',['Effect of P on Fy - data - ' filename ],
'NumberTitle','Off')
plot(fmdataP(1:l/d,1),fmdataP(1:l/d,4),'r')
hold on
plot(fmdataP((l/d+1):(2*l/d),1),fmdataP((l/d+1):(2*l/d),4),'k')
plot(fmdataP((2*l/d+1):(3*l/d),1),fmdataP((2*l/d+1):(3*l/d),4),'b')
legend('10psi','14psi','16psi')
grid on
hold off

% 3-D plot for SA vs. P vs. FY - Data, Spline and Fit

fmdataPf = sortrows(fmdataPf,[1,2]); %sort for reshape
pace12fyf=Pac12_Fy_Prestricted(Af,INPUT);
figure('Name','only Pressure coeffs optimized : Pac12 - Fy',
'NumberTitle','off');%'MenuBar','none'
hold on
fnplt(csaps({slipsf,pressuresf},fy0f,0.9999))
plot3(INPUT(inxPf,1),INPUT(inxPf,3),t.FY(inxPf),'k.')
plot3(INPUT(inxPf,1),INPUT(inxPf,3),pace12fyf(inxPf),'ro')
view(30,45)
title({['FZ = ' num2str(FZreqd) ' N'];['IA = ' num2str(Creqd) '']})
xlabel('Slip Angle ()')
ylabel('Pressures (psi)')
zlabel('Lateral Force (N)')
legend('WAC Spline','Data Pts','Pac12')
colormap(white)

%% 3-D Comparison Plots for SA vs. IA vs. FY

Preqd = 10;
inxPrf = find((fmdatafull(:,2)==FZreqd) & (fmdatafull(:,4)==Preqd));
fmdataPrf = fmdatafull(inxPrf,:);
```

```
fmdataPrf = sortrows(fmdataPrf,[1,2]); %sort for reshape

fy0Prf     = reshape(fmdataPrf(:,5),ncambersf,nslipsf)';

pace12fyPrf=Pac12_Fy_Prestricted(Af,INPUT);
figure('Name','Only Pressure Coeffs Optimized :
Pac12 - SA vs IA vs FY','NumberTitle','off');%'MenuBar','none'
hold on
fnplt(csaps({slipsf,cambersf},fy0Prf,0.9999))
plot3(INPUT(inxPrf,1),INPUT(inxPrf,4),t.FY(inxPrf),'k.')
plot3(INPUT(inxPrf,1),INPUT(inxPrf,4),pace12fyf(inxPrf),'ro')
view(30,45)
title({['FZ = ' num2str(FZreqd) ' N'];['P = ' num2str(Preqd) ' psi']})
xlabel('Slip Angle ()')
ylabel('Inclination Angle ()')
zlabel('Lateral Force (N)')
legend('Spline','Data Pts','Pac12 - ZeroValued')
colormap(white)
hold off

%update all to best-fit coefficients
PCY1    = A(1);
PDY1    = A(2);
PDY2    = A(3);
PDY3    = A(4);
PEY1    = A(5);
PEY2    = A(6);
PEY3    = A(7);
PEY4    = A(8);
PEY5    = A(9);
PKY1    = A(10);
PKY2    = A(11);
PKY3    = A(12);
PKY4    = A(13);
PKY5    = A(14);
PKY6    = A(15);
PKY7    = A(16);
PHY1    = A(17);
PHY2    = A(18);
PVY1    = A(19);
PVY2    = A(20);
PVY3    = A(21);
PVY4    = A(22);
PPY1    = A(23);
PPY2    = A(24);
```

```
PPY3    = A(25);
PPY4    = A(26);
PPY5    = A(27);

function FY = Pac12_Fy(A,X)

global FZ0 pi0 LFZ0 LCY LMUY LEY ...
    LKYG LHY LVY LGAY KY LKYA LDMUY EPSK

ALPHA  =  X(:,1)*pi/180;
FZ     =  abs(X(:,2));
P  =  X(:,3);
% GAMMA = zeros(length(P),1);
GAMMA  =  X(:,4)*pi/180;
GAMMAY = GAMMA .* LGAY; %31 (%48 lgay=lg
FZ0PR  = FZ0  .*  LFZ0; %15
DFZ    = (FZ-FZ0PR) ./ FZ0PR; %14,  (%30)
dpi    = (P - pi0)./pi0;

PCY1    = A(1);
PDY1    = A(2);
PDY2    = A(3);
PDY3    = A(4);
PEY1    = A(5);
PEY2    = A(6);
PEY3    = A(7);
PEY4    = A(8);
PEY5    = A(9);
PKY1    = A(10);
PKY2    = A(11);
PKY3    = A(12);
PKY4    = A(13);
PKY5    = A(14);
PKY6    = A(15);
PKY7    = A(16);
PHY1    = A(17);
PHY2    = A(18);
PVY1    = A(19);
PVY2    = A(20);
PVY3    = A(21);
PVY4    = A(22);
PPY1    = A(23);
PPY2    = A(24);
PPY3    = A(25);
PPY4    = A(26);
```

```
PPY5    = A(27);

%-- lateral force (pure side slip)
%Pg. 179 - Tyres and VD - Pacejka - 3rd Edition
SVYG    = FZ.*(PVY3+PVY4.*DFZ).*GAMMAY.*LKYG.*LDMUY;%4.E28
KYG0    = FZ.*(PKY6 + PKY7.*DFZ).*(1.0 + PPY5.*dpi).*LKYG;%4.E30
CY      = PCY1 .* LCY;  %4.E21
MUY     = (PDY1+PDY2 .* DFZ) .*(1.0 + PPY3.*dpi + PPY4.*dpi.^2).*
 (1.0-PDY3 .* GAMMAY.^2) .* LMUY; %4.E23.
DY      = MUY .* FZ; %4.E22 Turn slip is neglected.
KYA0    = PKY1 .* FZ0PR .*(1.0 + PPY1.*dpi).*
sin(PKY4.* atan(FZ ./ ((PKY2+PKY5.*GAMMAY.^2).*
(1+PPY2.*dpi).* FZ0PR))) .* LKYA; %4.E25
KY      = KYA0.*(1.0-PKY3 .* abs(GAMMAY)); %4.E25
SHY     = (PHY1+PHY2 .* DFZ) .* LHY +
(KYG0.*GAMMAY - SVYG)./(KY+EPSK); %4.E27
ALPHAY  = ALPHA+SHY;  %4.E20
BY      = KY ./ (CY .* DY);  %4.E26
EY      = (PEY1+PEY2 .* DFZ) .* (1.0 + PEY5.*GAMMAY.^2 -
(PEY3+PEY4 .* GAMMAY) .* sign(ALPHAY)) .* LEY; %4.E24
SVY     = FZ .* (PVY1+PVY2 .* DFZ) .* LVY.* LDMUY + SVYG; %4.E29
FY0     = DY .* sin(CY .* atan(BY .* ALPHAY-EY .*
 (BY .* ALPHAY-atan(BY .* ALPHAY))))+SVY; %4.E19
FY      = FY0;
end


function FY = Pac12_Fy_Prestricted(A,X)

global FZ0 pi0 LFZ0 LCY LMUY LEY ...
    LKYG LHY LVY LGAY KY LKYA LDMUY EPSK

global PCY1 PDY1 PDY2 PDY3...
    PEY1 PEY2 PEY3 PEY4 PEY5...
    PKY1 PKY2 PKY3 PKY4 PKY5 PKY6 PKY7...
    PHY1 PHY2  ...
    PVY1 PVY2 PVY3 PVY4;


ALPHA  =  X(:,1)*pi/180;
FZ     =  abs(X(:,2));
P  =  X(:,3);
% GAMMA = zeros(length(P),1);
GAMMA  =  X(:,4)*pi/180;
GAMMAY = GAMMA .* LGAY; %31 (%48 lgay=lg
```

```
FZ0PR  = FZ0  .*  LFZ0; %15
DFZ    = (FZ-FZ0PR) ./ FZ0PR; %14,  (%30)
dpi    = (P - pi0)./pi0;

PPY1  = A(1);
PPY2  = A(2);
PPY3  = A(3);
PPY4  = A(4);
PPY5  = A(5);
PPY6  = A(6);

%-- lateral force (pure side slip)
%Pg. 179 - Tyres and VD - Pacejka - 3rd Edition
SVYG  = FZ.*(PVY3+PVY4.*DFZ).*GAMMAY.*LKYG.*LDMUY;%4.E28
KYG0  = FZ.*(PKY6 + PKY7.*DFZ).*(1.0 + PPY5.*dpi).*LKYG;%4.E30
CY    = PCY1 .* LCY;  %4.E21
MUY   = (PDY1+PDY2 .* DFZ) .*(1.0 + PPY3.*dpi + PPY4.*dpi.^2).*

 (1.0-PDY3 .* GAMMAY.^2) .* LMUY; %4.E23.
 %We neglect LMUV as not dealing with wet road.
DY    = MUY .* FZ; %4.E22 Turn slip is neglected.
KYA0  = PKY1 .* FZ0PR .*(1.0 + PPY1.*dpi).*
sin(PKY4.* atan(FZ .* FZ0PR./ ((PKY2+PKY5.*GAMMAY.^2).*
(1+PPY2.*dpi+PPY6.*dpi.^2)))) .* LKYA; %4.E25
KY    = KYA0.*(1.0-PKY3 .* abs(GAMMAY)); %4.E25
SHY   = (PHY1+PHY2 .* DFZ) .* LHY +
(KYG0.*GAMMAY - SVYG)./(KY+EPSK); %4.E27
ALPHAY= ALPHA+SHY;  %4.E20
BY    = KY ./ (CY .* DY);  %4.E26
EY    = (PEY1+PEY2 .* DFZ) .* (1.0 + PEY5.*GAMMAY.^2 -
(PEY3+PEY4 .* GAMMAY) .* sign(ALPHAY)) .* LEY; %4.E24
SVY   = FZ .* (PVY1+PVY2 .* DFZ) .* LVY.* LDMUY + SVYG; %4.E29
FY0   = DY .* sin(CY .* atan(BY .* ALPHAY-EY .*
(BY .* ALPHAY-atan(BY .* ALPHAY))))+SVY; %4.E19
FY    = FY0;
end
```