

# Nonlinear optimization of a new polynomial tyre model

Alberto López · José Luis Olazagoitia ·  
Cristina Moriano · Antonio Ortiz

Received: 28 January 2014 / Accepted: 5 August 2014 / Published online: 22 August 2014  
© Springer Science+Business Media Dordrecht 2014

**Abstract** Tyre behaviour is strongly nonlinear. This article presents the validation of a new polynomial tyre model with real test data, analysing the convergence properties during the optimization process to calculate the values of the parameters. A multivariate model with 13 parameters is shown, including normal load and camber angle. The article reviews the methods of getting polynomial approximations of the magic formula tyre model used to develop the new polynomial model, the numerical optimization methods which calculate the parameters of the model from real test data, and it explains how the terms of the Jacobian matrix are modified when we impose constraints to the curve; this can be useful to improve the adjustment in some areas of the curve. The convergence properties are shown both for the magic formula tyre model and for this polynomial tyre model. The proposed model presents a fast convergence both in one and in three variables. This is an additional advantage to its excellent analytical properties, and the model is very easy to compute and can be easily derived and integrated. It is very well adapted for real-time computing.

**Keywords** Polynomial tyre model · Tyre model coefficients · Nonlinear optimization

## 1 Introduction

Tyre models are important to evaluate the behaviour of such an important component of a car. Those models calculate forces in the tyre-ground contact. Probably, the most accurate and widely used by the community of automotive engineers is the so-called Pacejka's magic formula tyre model [7, 8] and [9], (see 2.1).

Due to the nonlinear behaviour of the tyre, the optimization procedure required to calculate the parameters of those models for an optimum adjustment to test data is not a simple problem, because the forces in the contact depend on slip, slip angle, normal load and camber angle; thus, the mathematical problem is a nonlinear multivariate optimization problem. The complexity of the mathematical formulation of the model can influence both the easiness of computing the model and the convergence properties during the nonlinear optimization process. The Pacejka's magic formula tyre model uses a complex nested inverse tangent function.

The authors of this paper have been looking for a simpler expression quicker and easier to process both during the optimization and during the direct computing of the model, more suitable for real-time applications.

---

A. López (✉) · J. L. Olazagoitia · C. Moriano  
Industrial Engineering Department, Universidad Antonio  
de Nebrija, C/Pirineos 55, 28040 Madrid, Spain  
e-mail: alopezro@nebrija.es

A. Ortiz  
Department of Mechanical Engineering, Universidad de Málaga,  
C/ Doctor Ortiz Ramos s/n, Escuela de Ingenierías, Campus de  
Teatinos, 29071 Málaga, Spain

The new polynomial model presented and validated in this article is obtained from the magic formula expression, by using theory of approximation, expanding the magic formula in series of Jacobi orthogonal polynomials. In the following section, we summarize how that expansion was obtained.

This article validates the new model with real test data and analyses the convergence properties of the model during the optimization process, to calculate the values of the parameters. A multivariate model is proposed including the influence of camber angle and normal load.

This work is integrated in a more general line of research, whose goal is to obtain fast computing solutions of the vehicle nonlinear equations, expanding them in orthogonal polynomial series (Chebyshev and Jacobi polynomials). The application is saving computing time in pre-collision situations for active safety devices (see the two PhD Thesis of the authors [1–3] and the papers of Amirouche [4] and Ferrara [5,6]).

## 2 Theoretical background

### 2.1 The magic formula tyre model

The well-known tyre model proposed by Bakker, Nyborg and Pacejka [7,8] and [9] is a semi-empirical tyre model based on the “magic” formula:

$$Y = D \cdot \sin[C \cdot \arctan(BX - E \cdot [BX - \arctan(BX)])]$$

This model is widely used and accepted by the community of automotive engineers and is also considered the most accurate. For that reason, we use it as the reference in this paper.

The shape of the curve is controlled by four parameters:  $B$ ,  $C$ ,  $D$  and  $E$ . The equation can calculate the following:

- Lateral forces in a tyre,  $F_y$ , as a function of the slip angle of the tyre,  $\alpha$  (in degrees)
- Braking force,  $F_x$ , as a function of longitudinal slip  $K$  (%).
- Self-aligning torque,  $M_z$ , as a function of the slip angle  $\alpha$ .

Figure 3 shows the aspect of this magic formula model in the case of a longitudinal force.

$B$ ,  $C$ ,  $D$  and  $E$  are constants that describe the inclination of the curve at the origin ( $BCD$ ), the peak

value ( $D$ ), the curvature ( $E$ ) and the basic form ( $C$ ) for each case (lateral, braking or self-aligning torque). In addition, the curve can have vertical ( $Sv$ ) or horizontal ( $Sh$ ) shifts at the origin. The full expression is as follows:

$$Y = D \cdot \sin[C \cdot \arctan(B(X + Sh) - E \cdot [B(X + Sh) - \arctan(B(X + Sh))]) + Sv$$

Coefficients  $B$ ,  $D$  and  $E$  are functions of the vertical load in the tyre,  $F_z$ :

$$d = a_1 \cdot F_z^2 + a_2 \cdot F_z; \quad B = BCD / (C \cdot d);$$

$$E = a_6 \cdot F_z^2 + a_7 \cdot F_z + a_8;$$

$$BCD_1 = \frac{a_3 \cdot F_z^2 + a_4 \cdot F_z}{e^{a_5 \cdot F_z}};$$

$$BCD_2 = a_3 \cdot \sin(a_4 (\arctan(a_5 \cdot F_z)));$$

$BCD_1$  is valid for the longitudinal force and the self-aligning torque with  $C = 1.65$  and  $C = 2.4$ , respectively.

$BCD_2$  is valid for the lateral force with  $C = 1.3$ .

The Camber angle  $\gamma$  in the wheel modifies the shifts  $Sh$  and  $Sv$  and the stiffness  $BCD$ :

$$\Delta S_h = a_9 \cdot \gamma; \quad \Delta S_v = (a_{10} \cdot F_z^2 + a_{11} \cdot F_z) \cdot \gamma;$$

$$\Delta B = -a_{12} \cdot |\gamma| \cdot B; \quad E_1 = \frac{E_0}{(1 - a_{13} \cdot |\gamma|)}$$

$E_1$  is the  $E$  value modified by the camber angle in the self-aligning torque calculation.

In the next two sections, we explain how to obtain a polynomial approximations to this magic formula model.

### 2.2 Approximation of a function in Chebyshev series

The Chebyshev polynomials, see [10], of the first kind are defined by  $T_n(x) = \cos[n \arccos(x)]$  and are orthogonal regarding the function  $w(x) = (1 - x^2)^{-1/2}$  in the interval  $[-1, 1]$ .

To work in different  $[a, b]$  intervals, shifted polynomials with the following change must be used:

$$t = 1/2[(b - a)x + a + b].$$

Their general expression [11] is the following:

$$T_n(x) = \frac{n}{2} \sum_{m=0}^{\lfloor n/2 \rfloor} (-1)^m \frac{(n - m - 1)!}{m!(n - 2m)!} (2x)^{n-2m};$$

$$n = 1, 2, 3 \dots; T_0(x) = 1$$

$[n/2]$  is the highest whole number  $\leq n/2$ . They fulfil the following recursive property:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x); \quad n = 1, 2, \dots$$

Chebyshev polynomials can be computed and manipulated using the MAPLE Orthopoly library. The expansion of a function in Chebyshev series (ACh) has the following form:

$$f(x) = \sum_{n=0}^{\infty} 'a_n T_n(x),$$

The single comma in the summation indicates that the first term must be divided by 2.

This expansion usually converges faster than the power series and the coefficients get the value:

$$a_n = \frac{1}{r_n} \int_{-1}^1 w(x) \cdot f(x) T_n(x) dx$$

Where  $w(x)$  is the weight function  $w(x) = (1 - x^2)^{-1/2}$ . If we truncate the series in degree  $N$ , we get an approximation to the function, the more accurate the higher  $N$  is. Due to properties of Chebyshev polynomials, truncating in  $N - 1$  is the best  $N - 1$  degree polynomial approximation to the development at  $N$  degree.  $r_n$  is the norm of the function ( $\pi/2$  for Chebyshev polynomials).

The coefficients  $a_n$  can be assessed with the direct integration in some functions, but, in general, this is not possible and the previous integral must be approximated by some other quadrature formula. This research work has been implemented in MAPLE, which uses quadrature algorithms, which first analyse the singularities and then use Clenshaw–Curtis quadrature [12, 13]; if the result is not satisfactory, Newton-Cotes adaptive formulae are used. All this is carried out at the Chebpade function from MAPLE Numapprox library of approximation of functions.

Chebyshev-Padé functions obtain good approximations, but not those of minimum–maximum error (known as minimax). To find the latter, the Remez algorithm [14] is used, which fine tunes the result by numeric iterations and converges to an improved minimax approximation.

The Remez algorithm produces optimal results at the approximation. This method allows the calculation of minimum error of any given function  $f(t)$  weighted with any weight term  $w(t)$ . If  $w(t) = 1/|f(t)|$  is used, the minimum relative error is obtained. These methods are described in any good book on the approximation theory [15].

In MAPLE, the Remez algorithm is implemented by the minimax function included in the Numapprox library of approximation of functions.

Next, we introduce Jacobi polynomials because they introduce flexibility in the approximation.

### 2.3 Expansion in series of Jacobi polynomials

Within the families of classic orthogonal polynomials generated from the Sturm-Liouville differential equation, from which Chebyshev polynomials also derive, we consider now the Jacobi polynomials, see [16]. Jacobi polynomials can also be computed and manipulated using the MAPLE Orthopoly library. The expansion of a function in series of Jacobi polynomials uses a Jacobi weight function this time. The integral must be programmed, and a library for expansions of functions in Jacobi series is not available in MAPLE.

$$f(x) \approx \sum_{k=0}^n a_k \cdot J_k(x);$$

$$a_n = \frac{1}{r_n} \int_{-1}^1 w(x) \cdot f(x) J_n(x) dx$$

The Jacobi weight function in this type of orthogonal polynomials is the following:

$$w(x) = \frac{(1 - x)^\delta}{(1 + x)^\gamma}$$

This function is controlled by two parameters  $\delta$  and  $\gamma$  that allow choosing the area of a best approximation at the orthogonality interval. In practice, this is very interesting as it will allow us to improve the adjustment of the error at any area of the longitudinal force, lateral force or self-aligning torque curves, depending on the application in which the approximation is used, for instance, looking either for a more reduced error in slip values close to zero or in values close to the maximum stress or in the maximum slip point (100%), (see Fig. 3).

The norm  $r_n$  in Jacobi polynomials is not constant, but it is also a function of  $\delta$ ,  $\gamma$  and the degree of the  $n$  polynomial.

$$r_n = \frac{2^{\delta+\gamma+1} \Gamma(n + \delta + 1) \cdot \Gamma(n + \gamma + 1)}{n! \cdot (2n + \delta + \gamma + 1) \cdot \Gamma(n + \delta + \gamma + 1)}$$

The recurrence relation seen for the Chebyshev polynomials now takes a more general expression in the case of Jacobi polynomials:

$$J_{n+1}^{(\delta,\gamma)}(x) = (a_n + b_b) \cdot J_n^{(\delta,\gamma)}(x) - c_n \cdot J_{n-1}^{(\delta,\gamma)}(x);$$

$$n = 1, 2, \dots$$

Where the recurrence coefficients are now:

$$a_n = \frac{(2n + 1 + \delta + \gamma)(2n + 2 + \delta + \gamma)}{2(n + 1)(n + 1 + \delta + \gamma)}$$

$$b_n = \frac{(\delta^2 - \gamma^2)(2n + 1 + \delta + \gamma)}{2(n + 1)(2n + \delta + \gamma)(n + 1 + \delta + \gamma)}$$

$$c_n = \frac{(n + \delta)(n + \gamma)(2n + 2 + \delta + \gamma)}{(n + 1)(n + 1 + \delta + \gamma)(2n + \delta + \gamma)}$$

### 3 The new polynomial tyre model

#### 3.1 General description

As a result of the expansion of the magic formula in series of Jacobi polynomials, the authors obtained a very simple mathematical expression to calculate longitudinal and lateral forces in a tyre [17]:

$$F = A_0 + A_1 \cdot \frac{x}{x + b} + A_2 \cdot \left(\frac{x}{x + b}\right)^2 + A_3 \cdot \left(\frac{x}{x + b}\right)^3; \tag{1}$$

A simple degree  $N = 3$  polynomial in an easy rational function  $x/(x+b)$ .

- $F$ : can be lateral ( $F_y$ ) or longitudinal ( $F_x$ ) force, the expression is valid for both. For self-aligning torque, a degree four polynomial should be used to obtain good accuracy.
- $x$ : can be longitudinal slip ( $s$ ) or slip angle ( $\alpha$ ) according to what force we are considering.
- $A_i$  and  $b$  are the basic parameters of the model. Usual values of  $b$  are between 3 and 8; values around five are very common.

This model shows excellent coincidence with the original magic formula (the maximum difference is lower than 1% with  $N = 3$ ), both for  $F_x$  and  $F_y$ . Self-aligning torque requires a degree four polynomial. The model has excellent analytical properties; it is possible to obtain the position of extreme points, asymptotes, analytic derivatives and integrals of this expression in an easy manner (the last is not possible in the original magic formula). Finally, the main advantage

is the facility of processing (test showed processing time 20 times faster than the magic formula tyre model). Obviously, the inverse tangent nested functions of the magic formula are very inefficient in terms of computation.

We have to calculate the term  $v = \frac{x}{x+b}$  only once, including it in the polynomial, for more efficiency, the Horner polynomial form can be used:

$$F = A_0 + A_1 \cdot v + A_2 \cdot v^2 + A_3 \cdot v^3$$

$$= A_0 + (A_1 + (A_2 + A_3 \cdot v) \cdot v) \cdot v$$

The work [17] was based in previous papers of the authors, [18,19].

In [17], we had published our theoretical polynomial formula, comparing it with the mathematic expression of the magic formula, but without any validation with real test data.

But in the present paper, we tackle the problem of nonlinear optimization, that is, how to obtain the parameters of our model from test data and analysing the convergence of our model comparing it with the speed of convergence of the MF Tyre model (this had not been analysed in [17]).

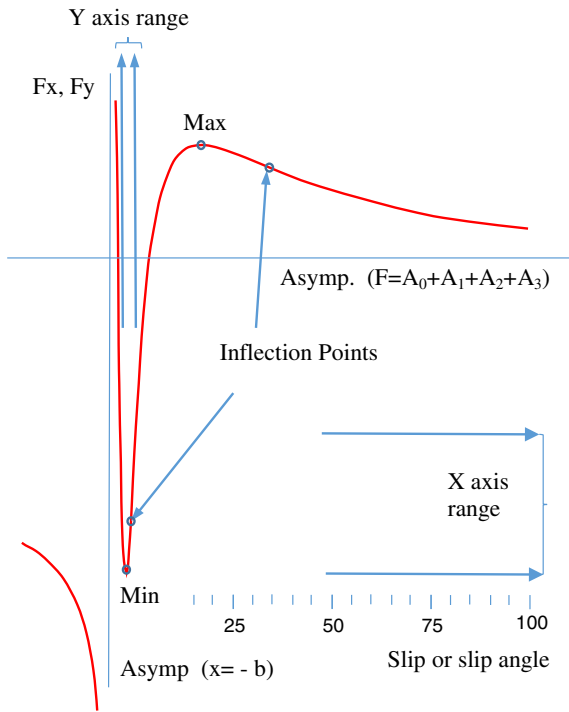
The approximate function proposed by the authors (1) presents a typical aspect as the one seen in Fig. 1

In the horizontal axis, the graph represents the longitudinal slip or the lateral slip. The vertical axis shows the longitudinal force  $F_x$  or the lateral force  $F_y$ . The model is valid for both, with different values of the parameters obviously.

The different curve branches are shown in Fig. 1. Obviously, and regarding the tyre model, only the branch from the minimum point to the right is used.

In this useful area, the curve shows two local ends at the interval 0–100. A typical maximum around  $x = 15$  (for the longitudinal force) and a minimum close to the origin. Depending on the values of the coefficients, this minimum point could be in any of the four quadrants. As it is a polynomial of a rational function, this function changes very quickly near the minimum; therefore, we must be very careful in the process of approximation to test data in order to keep the curve on the right of the minimum value. We will see how to achieve this in Sect. 5.

Depending on the coefficients, the position of both inflection points allows a very flexible adaptation to the curvature not only at the ascending branch on the right of the minimum, but also at the horizontal area on the right of the maximum.



**Fig. 1** Curve of the proposed polynomial model. Lateral or longitudinal force versus slip angle or slip

Obviously, the use of symmetry will allow symmetric or asymmetric branches describing equal or different behaviours in traction or braking or in asymmetric lateral behaviour on the right or left.

Both vertical and horizontal shifts of the magic formula can evidently be applied in a natural form (already integrated in the equation itself), but in a more flexible way as an inflection point can be kept in the upwards section of every branch when working with two equations, one for each side of the symmetry, in case the tyre’s behaviour requires it.

Let us see now the mathematical analysis of the curve.

3.2 Function derivatives

$$F = A_0 + A_1 \cdot u + A_2 \cdot u^2 + A_3 \cdot u^3$$

where

$$u = \frac{x}{x + b}; u' = \frac{b}{(x + b)^2}; u'' = \frac{-2 \cdot b}{(x + b)^3}$$

$$F' = (A_1 + 2 \cdot A_2 \cdot u + 3 \cdot A_3 \cdot u^2) \cdot u'$$

$$F'' = (2 \cdot A_2 \cdot u' + 6 \cdot A_3 \cdot u \cdot u') \cdot u' + (A_1 + 2 \cdot A_2 \cdot u + 3 \cdot A_3 \cdot u^2) \cdot u''$$

$$F_0 = A_0; F'_0 = \frac{A_1}{b}; F''_0 = \frac{2}{b^2}(A_2 - A_1)u_0;$$

$$u'_0 = \frac{1}{b}; u''_0 = -\frac{2}{b^2}$$

3.3 Maximum and minimum values

The position of the extrema in function of the coefficients is easily calculated as follows:

$$u_{max} = \frac{-A_2 \pm \sqrt{A_2^2 - 3A_1A_3}}{3A_3}; x_{max} = \frac{b \cdot u_{max}}{(1 - u_{max})}$$

$$F_{max} = A_0 + A_1 \cdot u_{max} + A_2 \cdot u_{max}^2 + A_3 \cdot u_{max}^3$$

The positive value of the root corresponds to the local minimum close to  $x = 0$  and the negative one to the maximum close to  $x = 15$ .

3.4 Inflection points

The position of the inflection points in function of the polynomial coefficients is the following:

$$F'' = 0 = (2 \cdot A_2 + 6 \cdot A_3 \cdot u) \cdot u'^2 + (A_1 + 2 \cdot A_2 \cdot u + 3 \cdot A_3 \cdot u^2) \cdot u'';$$

$$(2 \cdot A_2 + 6 \cdot A_3 \cdot u) \cdot \frac{b^2}{(x + b)^4}$$

$$= (A_1 + 2 \cdot A_2 \cdot u + 3 \cdot A_3 \cdot u^2) \cdot \frac{-2 \cdot b}{(x + b)^3}$$

$$R = -(A_1 + 2A_2 + 3A_3); S = b \cdot (3A_3 - A_2 - 2A_1);$$

$$T = b^2(A_2 - A_1); R \cdot x^2 + S \cdot x + T = 0$$

$$x_{inf} = \frac{-S \pm \sqrt{S^2 - 4 \cdot R \cdot T}}{2 \cdot R}$$

The negative root corresponds to an inflection point placed in the ascending section on the right of the minimum, and the positive one to the point on the right of the maximum.  $R, S$  and  $T$  are intermediate auxiliary variables used in order to simplify the expressions, but without any conceptual interest.

3.5 Asymptotes

The curve represents a vertical asymptote in  $x = -b$  and an horizontal asymptote on the right of the origin in  $x = A_0 + A_1 + A_2 + A_3$

### 3.6 Symmetries and shifts

The symmetric curve in the second quadrant, which will be called  $F2$ , is obtained by simply changing the sign of parameter  $b$ , making  $u = x/(x - b)$ . The symmetric curve in the third quadrant is  $-F2$ , and the symmetric function in the fourth quadrant is  $-F$ . If the behaviour of the tyre is symmetric, the same equation (with the same coefficients) can be used; if it is asymmetric, coefficients can be changed.

The application of shifts  $S_x$  and  $S_y$  is also very easy:

$$F_{shifted} = F(x + S_x) + S_y$$

## 4 Getting coefficients from tests

### 4.1 Introduction

In [17], this polynomial formulation was achieved from the magic formula and the approximation theory implemented on symbolic calculation programs, in particular MAPLE. In this paper, we validate the model with real test data, using a nonlinear optimization method, in a multivariate domain, taking into account not only the slip or slip angle, but also camber angle and normal load too.

At this point, we review the different methods of optimization present in the bibliography and we explain the application to both the proposed new polynomial model and the magic formula tyre model.

The main methods of nonlinear optimization used in the approximation of tyre models with test data can be classified as follows:

- Newton’s methods
  - Newton’s method
  - Gauss–Newton method and the Marquardt–Levenberg variant
  - Quasi-Newton methods
- SQP methods
- Iterative methods from the simplex method (Nelder–Mead)
- Genetic algorithms

We describe now the functioning of those methods. All of them have been applied to the estimation of the parameters of the presented polynomial tyre model with good convergence results. They have been applied to the magic formula tyre model too.

### 4.2 The Newton’s method

The basic tenet of the Newton method for nonlinear least-squares optimization is the following [20]:

If we have a set of  $m$  test points  $(x_i, y_i)$ , where, in general,  $y_i$  is the longitudinal or lateral force related to slip or lateral slip, which we also denote  $x_i$ , as the  $m = 55$  test points that can be seen in the Fig. 3 of Sect. 7.

The differences between the value predicted by our model and that presented in the test make-up a residue vector  $r$  (with a size of 55 in the proposed example) where every residue has the form:

$$r_i = y_i - F_i = y_i - (A_0 + A_1 \cdot u_i + A_2 \cdot u_i^2 + A_3 \cdot u_i^3);$$

being  $u_i = x_i/(x_i + b)$ , with  $i = 1 \dots m$ ; (55 points in our example).

If  $\beta$  ( $\beta_1 \dots \beta_5$ ) is the vector of the parameters of the model,  $\beta_j$ , where  $(j = 1, \dots, n)$  in this case  $n = 5$ , being in our particular model ( $\beta_1 = A_0, \beta_2 = A_1, \dots, \beta_5 = b$ ), the sum of the quadratic deviations will be a function of  $\beta$ :

$$S(\beta) = \sum_{i=1}^m r_i^2(\beta) \tag{2}$$

The Newton’s method starts from the Taylor series expansion of the function, and for simplicity, we assume that the function depends only on a unique parameter  $\beta$  at every point  $i$ :

$$(X_i, \beta_1) \approx F(x_i, \beta_1^k) + F'(x_i, \beta_1^k) \Delta\beta + \frac{1}{2} F''(x_i, \beta_1^k) \cdot (\Delta\beta)^2 + \dots$$

Being  $\Delta\beta = (\beta_1 - \beta_1^k)$

The Newton’s method establish that the function reaches its extrema when its derivative with respect to  $\Delta\beta = 0$ , that means:

$$F'(x_i, \beta_1^k) + F'(x_i, \beta_1^k) \cdot \Delta\beta = 0$$

Being as, in our model, the vector  $\beta$  contains now several parameters, the previous expression becomes:

$$G + H \cdot \Delta\beta = 0$$

From the previous equation, we can obtain the step of the parameter’s vector in every iteration, the so-called Newton’s step.

$$\Delta\beta = -H^{-1} \cdot G \tag{3}$$

and calculate the value of  $\beta$  in the next iteration,

$$\beta^{(s+1)} = \beta^s + \Delta\beta$$

$G$  is the gradient vector of  $S(\beta)$ , whose terms are as follows:

$$G_j = 2 \sum_{i=1}^m r_i \frac{\partial r_i}{\partial \beta_j}$$

$H$  is the Hessian matrix of  $S$ , obtained by differentiating the terms of the gradient:

$$H_{jk} = 2 \sum_{i=1}^m \left( \frac{\partial r_i}{\partial \beta_j} \frac{\partial r_i}{\partial \beta_k} + r_i \frac{\partial^2 r_i}{\partial \beta_j \partial \beta_k} \right)$$

This method and all its derived methods are iterative, and they need an initial value of the parameters' vector. The quality of the final result will depend notably on the goodness of this initial value.

### 4.3 Gauss–Newton and Marquardt–Levenberg methods

The Gauss–Newton method approximates the Hessian matrix, neglecting the second term of the previous equation as follows.

$$H_{jk} \approx 2 \sum_{i=1}^m J_{ij} \cdot J_{ik}; \quad J_{ij} = \frac{\partial r_i}{\partial \beta_j}$$

$J_{ij}$  is the Jacobian matrix which contains the partial derivatives of the vector of residuals with respect of the parameters of the model. If we write  $G$  and  $H$  in matrix notation, we obtain:

$$\begin{aligned} G &= 2 \cdot J_r^T \cdot r; \quad H \approx 2J_r^T \cdot J_r; \\ G + H \cdot \Delta\beta &= J_r^T \cdot r + J_r^T \cdot J_r \cdot \Delta\beta = 0; \\ J_r^T \cdot J_r \cdot \Delta\beta &= -J_r^T \cdot r \end{aligned}$$

In every iteration, the new value of the parameters' vector is the following:

$$\begin{aligned} \Delta\beta &= -(J_r^T \cdot J_r)^{-1} J_r^T \cdot r; \\ \beta^{(s+1)} &= \beta^{(s)} - (J_r^T J_r)^{-1} J_r^T \cdot r \end{aligned} \tag{4}$$

In the most basic tyre polynomial model, which includes initially five parameters, the Jacobian matrix is a  $(Np \times 5)$  matrix, being  $Np$  the number of data points (see the data vector in the example of Sect. 5). For this tyre model, the rows in this Jacobian matrix have the following shape:

$$\begin{aligned} \frac{\partial r_i}{\partial A_0} &= -1; \quad \frac{\partial r_i}{\partial A_1} = -u_i; \quad \frac{\partial r_i}{\partial A_2} = -u_i^2; \quad \frac{\partial r_i}{\partial A_3} = -u_i^3; \\ \frac{\partial r_i}{\partial b} &= \frac{1}{(x_i + b)} \left[ A_1 \cdot u_i + 2 \cdot A_2 \cdot u_i^2 + 3 \cdot A_3 \cdot u_i^3 \right] \end{aligned}$$

The Jacobian matrix terms in the magic formula are the following:

$$\begin{aligned} \frac{\partial r_i}{\partial S_v} &= -1; \quad \frac{\partial r_i}{\partial d} = -\sin(V_i); \\ \frac{\partial r_i}{\partial C} &= [-d \cdot \cos(V_i)] \cdot \arctan(W_i) \\ \frac{\partial r_i}{\partial B} &= \frac{\partial r_i}{\partial V_i} \cdot \frac{\partial V_i}{\partial W_i} \cdot \frac{\partial W_i}{\partial B} = [-d \cdot \cos(V_i)] \cdot \left[ \frac{C}{1 + W_i^2} \right] \\ &\quad \cdot \left[ x_i + S_h - E \left( x_i + S_v - \frac{x_i + S_h}{1 + B^2 \cdot (x_i + S_h)^2} \right) \right] \\ \frac{\partial r_i}{\partial E} &= \frac{\partial r_i}{\partial V_i} \cdot \frac{\partial V_i}{\partial W_i} \cdot \frac{\partial W_i}{\partial E} [-d \cdot \cos(V_i)] \cdot \left[ \frac{C}{1 + W_i^2} \right] \\ &\quad \cdot [-B(x_i + S_h) + \arctan(B \cdot (x_i + S_h))]; \\ \frac{\partial r_i}{\partial S_h} &= \frac{\partial r_i}{\partial V_i} \cdot \frac{\partial V_i}{\partial W_i} \cdot \frac{\partial W_i}{\partial S_h} [-d \cdot \cos(V_i)] \cdot \left[ \frac{C}{1 + W_i^2} \right] \\ &\quad \cdot \left[ B - E \left( B - \frac{B}{1 + B^2 \cdot (x_i + S_h)^2} \right) \right]; \end{aligned}$$

Being:

$$\begin{aligned} W_i &= B(x_i + S_h) - E \cdot [B(x_i + S_h) \\ &\quad - \arctan(B(x_i + S_h))] \text{ and } V_i = C \cdot \arctan(W_i) \end{aligned}$$

If convergence problems appear, there are several methods which modify the Gauss–Newton method. The first and most simple method consists of reducing the length of the step of the parameters' vector  $\Delta\beta$ , by multiplying it by a constant  $\alpha$  lower than 1.

$$\beta^{(s+1)} = \beta^{(s)} - \alpha \cdot (J_r^T J_r)^{-1} J_r^T r$$

In this manner, we can solve situations, in which, the step of the parameters' vector  $\Delta\beta$  points to the right direction (which reduces the addition of quadratic deviations), but it is too long.

The second method is the so-called Marquardt–Levenberg method, [21], in which, the step  $\Delta\beta$  is modified by adding the term  $\lambda \cdot D$ , where  $D$  is a positive diagonal matrix and  $\lambda$  is the so-called Marquardt's parameter. It is also called the trust region method. The addition of this term rotates the vector  $\Delta\beta$  towards the maximum descending slope.

$$\beta^{(s+1)} = \beta^{(s)} - (J_r^T J_r + \lambda \cdot D)^{-1} J_r^T r$$

In point 7, we will see the convergence of the Gauss–Newton method in our tyre example, which is very fast. The convergence of our model is compared with the convergence of the magic formula tyre model.

#### 4.4 Quasi-Newton methods

According to [22], in order to estimate the parameters of the magic formula tyre model, the research team of the TNO (the research organization at the Netherlands), used the so-called quasi-Newton method [23] and [24], implemented by the E04FDF subroutine of the NAG (Numerical Analysis Group) [25] (see <http://www.nag.co.uk/>). This method is similar to the Gauss–Newton method, but its application is not specific for least-squares problems, but its application field is wider, actually it can be used to optimize any function.

The family of quasi-Newton methods avoids the inversion of the Hessian matrix  $\mathbf{H}$  in Eq. 2, by calculating directly the inverse of a pseudo-Hessian matrix  $\mathbf{B}$ , which is obtained by successive approximations of the gradient  $\mathbf{G}$  in a generalization of the secant method to the multivariate domain. In this way, these methods improve the computational efficiency of the whole calculation. In order to estimate the pseudo-Hessian  $\mathbf{B}$ , different iterative algorithms have been used and published along the history, DFP (Davidon–Fletcher–Powell), [26,27], BFGS (Broyden–Fletcher–Goldfarb–Shanno), see [20], SR1 (Symmetric Rank 1), see [28] and [20] and the class of Broyden methods, see [29,30] and [20]. All of them use the Sherman–Morrison formula to invert the pseudo-Hessian matrix  $\mathbf{B}$ , see [20].

#### 4.5 The sequential quadratic programming (SQP) method

SQP method [31] poses the general problem of nonlinear optimization for a given target function  $S(\boldsymbol{\beta})$  of a parameters' vector  $\boldsymbol{\beta}$ , but now with a set of constraint equations  $\mathbf{G}(\boldsymbol{\beta}) \geq 0$ , which can be both equality or inequality functions of the parameters' vector too.

$$\mathbf{G}(\boldsymbol{\beta}) = (\mathbf{G}_1(\boldsymbol{\beta}), \dots, \mathbf{G}_m(\boldsymbol{\beta}))$$

SQP is an iterative method, and it models the nonlinear problem for a given iteration by a quadratic programming (QP) sub-problem, solves that QP sub-problem

and then uses the solution to find a new parameters' vector  $\boldsymbol{\beta}^{(s+1)}$ .

To find the solution, SQP uses the Lagrangian function that combines the objective function  $S(\boldsymbol{\beta})$  and the constraints  $\mathbf{G}(\boldsymbol{\beta})$  properly. The Lagrangian function of our problem is the following:

$$\mathbf{L}(\boldsymbol{\beta}, \mathbf{u}) = S(\boldsymbol{\beta}) - \mathbf{u}^T \mathbf{G}(\boldsymbol{\beta})$$

where  $\mathbf{u}$  is the vector of Lagrange's multipliers of the nonlinear problem. SQP replaces the objective function  $S(\boldsymbol{\beta})$  by its local quadratic approximation, expanding it in a Taylor series, and the constraint functions  $\mathbf{G}(\boldsymbol{\beta})$  are replaced by their local linear approximations. This construction is done in such a way that the algorithm sequence converges to a local minimum. Modern optimization textbooks have chapters devoted to SQP methods, see [31].

#### 4.6 The Nelder–Mead method

This method was proposed by [32], see also [33,34] and [35], and it allows to minimize a target function in a multidimensional space. The method uses the “simplex” concept which uses elements of  $N + 1$  vertices, in a  $N$ -dimensional space. In a one-dimensional space, the simplex element is just a line. In a bi-dimensional one, the simplex element is a triangle, and in a tri-dimensional space, the element is a tetrahedron and so on.

The algorithm generates a new test position by extrapolating the behaviour of the target function in every vertex of the simplex. One of these vertices is replaced with a new point, and it progresses in this way. The easiest step is to replace the worst point with a new one obtained by reflecting it across the centroid of the  $N$  remaining points. If this new point is better than the best of all the current points, we can try to extend outwards the simplex element along this line. If the new point is not better than the previous one, it will probably be in a valley area and we should compress the simplex towards a better point. It is known that the method can converge at nonstationary points. This method is implemented in the *fminsearch* library of MATLAB.

#### 4.7 Genetic algorithms

Recently, the group of the Department of Mechanical Engineering of the University of Malaga (IMMA) has



developed a new optimization method to calculate the coefficients of the magic formula tyre model, see [36] and [37]. They propose the use of genetic algorithms, which work with high accuracy and efficiency, avoiding the use of initial values for the parameters. Genetic algorithms' techniques were presented initially by [38] and [39]. The IMMA applied them to tyre models first.

An interesting hybrid approach can be found in [40] that combines genetic algorithms with classic gradient search methods applied to multiparametric nonlinear systems.

Several authors who have worked in nonlinear multivariate optimization in tyre models agree that when using Newton, Gauss–Newton, quasi-Newton and Nelder–Mead methods, the selection of an adequate initial point of the parameters' vector is an important issue for the quality of the final solution, and minor variations in this initial point can produce different final results, see for example [22,36,37] and [20].

The reason for this is the nonlinear condition of the problem and the subsequent need for an iterative method with an initial point.

During this research work, we have programmed tyre models optimizations (by hand, writing the code in a low level software) using Newton, Gauss–Newton and Marquardt–Levenberg methods, both for the polynomial and for magic formula models, with and without constraints. We have also used the libraries of MAPLE and MATLAB which combine Nelder–Mead and Quasi-Newton methods with the previous ones, and the results are always coincident; we could observe many times this problem, in certain combination of the parameters, a slight change in the initial point can yield different results.

It is very difficult to give a general rule of when the optimization will not converge. A first factor is how big are the residues, that is, how far from the final curve is the initial values curve. But this is not the only factor, the shape of the initial values curve can also influence the possibility of convergence. In addition, the final result of the optimization can be apparently good, but actually, it could be a local minimum, and we could find a near combination of parameters with better optimization results (a lower sum of quadratic deviations) if we start with different initial values. Most optimization methods find local minima. Those problems are bigger when the number of parameters and the number of variables increases, because the number of possible combinations of data is bigger and we could find

minima very close. As usual, a good knowledge of the physical phenomenon (the tyre behaviour in this case) and the previous experience in optimization of similar tyres can help a lot.

Therefore, this is not a well-solved problem or at least not in a fully automatic way.

From this point of view, the genetic algorithms method, whose approach is probabilistic, nondeterministic and very different from the rest, is very interesting because it does not need an initial point.

Added to the previous methods, we can mention the work of [41] and [42] who estimates the values of the Pacejka's magic formula tyre model, using the so-called TS (two stage) technique and compares it with different methods of observation and parameters estimation based on Kalman filters, using data obtained along the life time of the vehicle.

If we want to obtain a fast convergence in the optimization algorithms, the initial point should not be far from the optimum. The first step in a nonlinear optimization process is the search of a reasonably good initial point. We can use the previous results of a different tyre under similar load conditions to obtain this initial point.

## 5 Optimization with constraints to the model coefficients

Sometimes, it may be necessary to obtain the optimum curve under certain constraints, because we are more confident in some points of the test than in others or because we want to equate the value of the curve, or the value of its derivatives, at both sides of the origin, or because we want to give a fixed value to the derivative at the origin. If we have equality constraints, a direct and effective technique is to include the constraints in the original equation, before starting the optimization process, as we show below, instead of formulating the constraint equations added, and using SQP algorithms, perhaps more adequate for inequality constraints.

If no constraints are imposed, the optimization algorithms will calculate the coefficients so that the value of the sum of the quadratic deviations is the minimum. As constraints are imposed, the quadratic deviation will be bigger and bigger, but the curve will comply those constraints. As our model has five basic coefficients, in theory, we could impose up to five constraints, although we must always keep in mind that the less constraints

we impose, the better the adjustment of the test data and the lesser the sum of the quadratic deviations will be.

Let us see now some typical constraint examples. All of them have been tested both with our polynomial tyre model and with the magic formula tyre model. We have obtained a fast convergence and moderate variations of the sum of quadratic deviations.

**R1. The curve passes exactly through point**  $(x_p, y_p)$ .

$$y_p = A_0 + A_1 \cdot u_p + A_2 \cdot u_p^2 + A_3 \cdot u_p^3;$$

being  $u_p = x_p/(x_p + b)$

The resulting equation is the following:

$$r_i = y_i - F_i = y_i - y_p + (A_1 \cdot u_p + A_2 \cdot u_p^2 + A_3 \cdot u_p^3) - (A_1 \cdot u_i + A_2 \cdot u_i^2 + A_3 \cdot u_i^3)$$

The terms of the Jacobian matrix for our tyre model are modified as follows:

$$\frac{\partial r_i}{\partial A_1} = u_p - u_i; \quad \frac{\partial r_i}{\partial A_2} = u_p^2 - u_i^2; \quad \frac{\partial r_i}{\partial A_3} = u_p^3 - u_i^3$$

$$\frac{\partial r_i}{\partial b} = \frac{1}{(x_i + b)} [A_1 \cdot u_i + 2 \cdot A_2 \cdot u_i^2 + 3 \cdot A_3 \cdot u_i^3]$$

$$- \frac{1}{(x_p + b)} [A_1 \cdot u_p + 2 \cdot A_2 \cdot u_p^2 + 3 \cdot A_3 \cdot u_p^3]$$

This constraint calculates the optimal value of the coefficients so that the curve passes through a given point, for example, the end point ( $s = 100$ ) or through the first data of the series at  $F_x = 0$ , which can be very reliable data from the test perspective: those two points can have a better measurement reliability than the points between the maximum value and the full slip point ( $s = 100\%$ ).

It can also be interesting that the curve passes exactly through the maximum test point. Generally, these three situations generate a small increase at the quadratic sum and adjust the curve at those points accurately.

Nevertheless, if the chosen point is the maximum one  $(x_{max}, y_{max})$ , this constraint does not guarantee  $y_{max}$  to be the curve maximum value, as it could have an anterior or posterior maximum.

The curve can be forced to pass exactly through up to five points, as it has five coefficients.

Particularly, the curve can be forced to pass through  $(0, f_y)$ . Naturally, combined constraints can be imposed, so that they can pass through  $(x_p, y_p)$  and  $(0, f_y)$ . This is a combination of the two previous ones.

**R2. The curve takes  $F = y_{max}$  as maximum value (even if it does not pass through  $x_{max}$  exactly).**

$$u_{max} = \frac{-A_2 - \sqrt{A_2^2 - 3A_1A_3}}{3A_3};$$

$$F_{max} = y_{max} = A_0 + A_1 \cdot u_{max} + A_2 \cdot u_{max}^2 + A_3 \cdot u_{max}^3$$

The resulting equation will be the following:

$$r_i = y_i - F_i = y_i - y_{max} + (A_1 \cdot u_{max} + A_2 \cdot u_{max}^2 + A_3 \cdot u_{max}^3) - (A_1 \cdot u_i + A_2 \cdot u_i^2 + A_3 \cdot u_i^3)$$

Let:

$$K = \sqrt{A_2^2 - 3 \cdot A_1 \cdot A_3} = -A_2 - 3 \cdot A_3 \cdot u_{max};$$

$$\frac{\partial r_i}{\partial u_{max}} = r'_{i,u_{max}}$$

$$= A_1 + 2 \cdot A_2 \cdot u_{max} + 3A_3 \cdot u_{max}^2$$

$$\frac{\partial u_{max}}{\partial A_1} = u'_{max,A_1} = \frac{1}{2 \cdot k}; \quad u'_{max,A_2} = \frac{-k - A_2}{3 \cdot k \cdot A_3};$$

$$u'_{max,A_3} = \frac{A_1}{2 \cdot k \cdot A_3} + \frac{A_2 + k}{3 \cdot A_3^2}$$

$$\frac{\partial r_i}{\partial u_i} = r'_{i,u_i} = -(A_1 + 2 \cdot A_2 \cdot u_i + 3A_3 \cdot u_i^2);$$

$$\frac{\partial u_i}{\partial b} = u'_{i,b} = \frac{-x}{(x + b)^2}.$$

The terms of the Jacobian matrix will be as follows:

$$\frac{\partial r_i}{\partial A_1} = u_{max} - u_i + r'_{i,u_{max}} \cdot u'_{max,A_1};$$

$$\frac{\partial r_i}{\partial A_2} = u_{max}^2 - u_i^2 + r'_{i,u_{max}} \cdot u'_{max,A_2}$$

$$\frac{\partial r_i}{\partial A_3} = u_{max}^3 - u_i^3 + r'_{i,u_{max}} \cdot u'_{max,A_3};$$

$$\frac{\partial r_i}{\partial b} = r'_{i,u_i} \cdot u'_{i,b}$$

**R3. The curve reaches its maximum value at  $x_{max}$  (even if it does not take value  $y_{max}$ )**

The resulting equation will be the following:  $r_i = y_i - F_i = y_i - (A_0 + A_1 \cdot u_i + A_2 \cdot u_i^2 + A_3 \cdot u_i^3)$ ;  $b = -(2 \cdot A_2 + A_1 + 3 \cdot A_3)/(A_2 + A_1 + K)$ ;  $u_i = x_i/(x_i + b)$

$$\frac{\partial u_i}{\partial A_1} = u'_{i,A_1} = \frac{-2 - 3 \cdot (2 \cdot A_2 + A_1 + 3 \cdot A_3) \cdot (-2 \cdot k + 3 \cdot A_3)}{2(A_2 + A_1 + K) \cdot K}$$

$$\frac{\partial u_i}{\partial A_2} = u'_{i,A_2} = \frac{(-A_1 + 3 \cdot A_3) \cdot K + (6 \cdot A_3 + A_2) \cdot A_1 + A_3 \cdot A_2}{(A_2 + A_1 + K) \cdot K}$$

$$\frac{\partial u_i}{\partial A_3} = u'_{i,A_3} = -\frac{3}{A_2 + A_1 + K} - \frac{3 \cdot (2 \cdot A_2 + A_1 + 3 \cdot A_3) \cdot A_1}{(A_2 + A_1 + K)^2 \cdot K}$$

The terms of the Jacobian matrix will be as follows:

$$\frac{\partial r_i}{\partial A_0} = -1;$$

$$\frac{\partial r_i}{\partial A_1} = -u_i - (A_1 + 2 \cdot A_2 \cdot u_i + 3 \cdot A_3 \cdot u_i^2) \cdot u'_{i,A_1};$$

$$\frac{\partial r_i}{\partial A_2} = -u_i^2 - (A_1 + 2 \cdot A_2 \cdot u_i + 3 \cdot A_3 \cdot u_i^2) \cdot u'_{i,A_2};$$

$$\frac{\partial r_i}{\partial A_3} = -u_i^3 - (A_1 + 2 \cdot A_2 \cdot u_i + 3 \cdot A_3 \cdot u_i^2) \cdot u'_{i,A_3}$$

**R4. The inflection point is located at the origin.**

The inflection point is at the point  $x_{inf}$

The condition of the inflection point at the origin is as follows:

$$(F_0'' = 0); A_2 = A_1;$$

$$F = A_0 + A_1 \cdot u \cdot (1 + u) + A_3 \cdot u^3$$

The terms of the Jacobian matrix this time will be as follows:

$$\frac{\partial r_i}{\partial A_0} = -1; \quad \frac{\partial r_i}{\partial A_1} = -u_i \cdot (1 + u_i); \quad \frac{\partial r_i}{\partial A_3} = -u_i^3;$$

$$\frac{\partial r_i}{\partial b} = \left[ A_1 + 2 \cdot A_1 \cdot u_i + 3 \cdot A_3 \cdot u_i^2 \right] \cdot \frac{x_i}{(x_i + b)^2}$$

**R5. The slope at a given point  $x_p$  takes a given value**

$$y'_p$$

$$y'_p = (A_1 + 2 \cdot A_2 \cdot u_p + 3 \cdot A_3 \cdot u_p^2) \cdot u'_p;$$

$$u_p = x_p / (x_p + b); u'_p = b / (x_p + b)^2;$$

$$A_1 = (y'_p / u'_p) - 2 \cdot A_2 \cdot u_p - 3 \cdot A_3 \cdot u_p^2$$

$$r_i = y_i - F_i = y_i - (A_0 + [(y'_p / u'_p) - 2 \cdot A_2 \cdot u_p - 3 \cdot A_3 \cdot u_p^2] \cdot u_i + A_2 \cdot u_i^2 + A_3 \cdot u_i^3)$$

$$\frac{\partial r_i}{\partial u'_p} = \frac{y'_p \cdot u_i}{u_p'^2}; \quad \frac{\partial u'_p}{\partial b} = \frac{x_p - b}{(x_p + b)^3};$$

$$\frac{\partial r_i}{\partial u_p} = (2 \cdot A_2 + 6 \cdot A_3 \cdot u_p) \cdot u_i; \quad \frac{\partial u_p}{\partial b} = \frac{-x_p}{(x_p + b)^2}$$

$$\frac{\partial r_i}{\partial u_i} = \frac{y'_p}{u'_p} + 2 \cdot A_2 \cdot u_p + 3 \cdot A_3 \cdot u_p^2 - 2 \cdot A_2 \cdot u_i - 3 \cdot A_3 \cdot u_i^2;$$

$$\frac{\partial u_i}{\partial b} = \frac{-x_i}{(x_i + b)^2}$$

The terms of the Jacobian matrix this time are:

$$\frac{\partial r_i}{\partial A_0} = -1; \quad \frac{\partial r_i}{\partial A_2} = 2 \cdot u_p \cdot u_i - u_i^2;$$

$$\frac{\partial r_i}{\partial A_3} = 3 \cdot u_p^2 \cdot u_i - u_i^3;$$

$$\frac{\partial r_i}{\partial b} = \frac{\partial r_i}{\partial u'_p} \cdot \frac{\partial u'_p}{\partial b} + \frac{\partial r_i}{\partial u_p} \cdot \frac{\partial u_p}{\partial b} + \frac{\partial r_i}{\partial u_i} \cdot \frac{\partial u_i}{\partial b}$$

**Particular case: the slope at the origin takes a given value.**

$F'_0 = A_1/b$ ; In this case, the equation is  $F = A_0 + b \cdot F'_0 \cdot u + A_2 \cdot u^2 + A_3 \cdot u^3$ . The terms of the Jacobian matrix are slightly modified:

$$\frac{\partial r_i}{\partial b} = \frac{-F'_0 \cdot x_i}{(x_i + b)} + \frac{b \cdot F'_0 \cdot x_i}{(x_i + b)^2} + \frac{2 \cdot A_2 \cdot x_i^2}{(x_i + b)^3} + \frac{3 \cdot A_3 \cdot x_i^3}{(x_i + b)^4}$$

The rest of the terms remains equal than in the equation without constraints, except  $A_1$  that disappears.

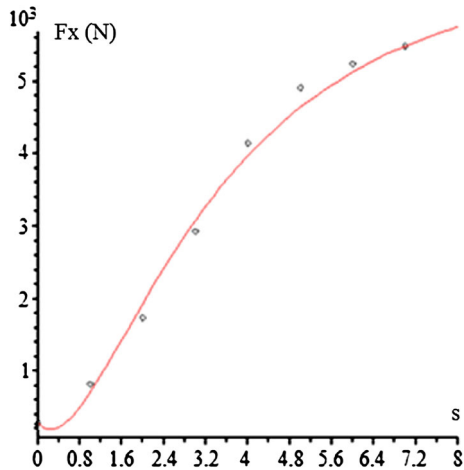
We could use this constraint to force the slope at the origin, to take the same value BCD of the magic formula tyre model.

A combination could be imposed, so that the inflection point is at the origin and also at  $F'_0$  takes a given value.

As a final consideration related to the constraint imposition, it must be stated that optimization algorithms converge when the imposed constraints make sense. If conditions are imposed to the curve that are far from the initial ones with the result of very distorted curve shapes, no final solution will be achieved.

**6 Stability of the polynomial at the origin**

The main drawback of using polynomials, especially rational polynomials, is their trend to oscillation, in other words, their possible instability. If the distribution of real points shows an inflection point at the rise, the polynomial with the best interpolation by least-squares



**Fig. 2** Oscillation at the origin

can have its minimum in the first quadrant (see Fig. 1) between two test points, something definitely unwanted (see Fig. 2).

To avoid it, we have to impose the constraint of a given value of the derivative at the origin,  $A_1 = F'_0 \cdot b$ .  $F'_0$  can be estimated by means of a parabolic regression from the first 4 points and calculating the derivative of this parabola at the origin. This increases slightly the addition of quadratic deviations, but it solves in a simple manner the problem because it moves the minimum point to a different quadrant and stabilizes the curve in the first quadrant. The correct value of  $F'_0$  may also be calculated by imposing the same value of the function, and its derivative, at both sides of the origin if we have data of forces in both sides. Figure 8 shows the solution in the multivariate case too.

**7 Results of the model with real tests and accuracy**

In [17], the adjustment of the proposed polynomial model was compared with the magic formula model in relation to both the longitudinal and lateral forces for different normal load values, and it could be seen that the deviations regarding that model were always very low, keeping a relative value below 1%.

Now, we analyse the optimization to real test data, not the adjustment to a theoretical model. The results in real tests present the typical instability of the tyre material inequalities, the instruments measurement uncertainty and the difficulty to stabilize the longitudinal or lateral slip at some areas of the curve. For this reason,

mathematical models are exactly that models adjusted to each type of tyre, but the resulting points always present the variability specific of tests, so that individual deviations may be produced at some point.

Next, some modelling results from tests on a real tyre are presented, showing the test points, the polynomial modelling proposed in this article and its comparison with the magic formula model.

**Longitudinal force.**

data = [[0, 276], [1, 824], [2, 1742], [3, 2930], [4, 4146], [5, 4913], [6, 5244], [7, 5492], [8, 5693], [9, 5844], [10, 5987], [11, 6097], [12, 6155], [13, 6193], [14, 6226], [15, 6253], [16, 6269], [17, 6277], [18, 6269], [19, 6243], [20, 6226], [21, 6202], [22, 6171], [23, 6152], [24, 6125], [25, 6101], [26, 6072], [27, 6048], [28, 6017], [29, 5978], [30, 5944], [31, 5918], [32, 5887], [33, 5860], [34, 5831], [35, 5797], [36, 5768], [37, 5723], [38, 5691], [39, 5662], [40, 5633], [41, 5597], [45, 5506], [50, 5404], [55, 5261], [60, 5186], [65, 5115], [70, 5028], [75, 4999], [80, 4954], [85, 4929], [90,4915], [95, 4882], [99, 4827], [100, 4796]]:

These are data from real longitudinal force tests on a 175/70 R13 tyre. Normal load = 6 kN. Camber angle = 0 and pure slip conditions.

In our model, we adjust the slope at the origin to the value 408 which is the value for the slope at the origin in a parabolic interpolation of the first 4 points. The obtained result in the optimization is the following:

$$F_x = -6.33 + 2199.78 \cdot u + 28102.83 \cdot u^2 - 26462.59 \cdot u^3; \text{ being } u = s/(s + 5.391)$$

The adjustment to the magic formula is the following:

$$\begin{aligned} f_y &= 2035.56; f_x = -2.055; B = 0.13915; \\ d &= 4226.87; E = 0, 7019; C = 1, 766 \\ F_x &= 4226.87 \cdot \sin(1.766 \cdot \arctan(0.13915 \cdot (x - 2.055) \\ &\quad - 0.7019 \cdot (0.13915 \cdot (x - 2.055) \\ &\quad - \arctan(0.13915 \cdot (x - 2.055)))))) + 2035.56 \end{aligned}$$

Let us see now the convergence of both models using the Gauss–Newton’s method for our example tyre with the previous data.

In Table 1, we show the calculation of the parameters for the longitudinal braking force  $F_x$ .

As we can see in Table 2, the magic formula model also converges using the Gauss–Newton’s method, but

**Table 1** Convergence of the polynomial tyre model parameters

Step	$A_0$	$A_1$	$A_2$	$A_3$	$b$
Initial value	0	1,000	45,000	-40,000	5.5
1st	-8.06629	2,212.796	28,016.681	-26,384.933	5.42352
2nd	-6.91953	2,198.965	28,104.114	-26,461.747	5.38962
3rd	-6.29362	2,199.826	2,810.723	-26,462.611	5.39173
4th	-6.33480	2,199.777	28,102.837	-26,462.591	5.39161
5th	-6.33248	2,199.781	28,102.831	-26,462.592	5.39162
...					
Final value	-6.33261	2,199.781	28,102.831	-26,462.592	5.39162

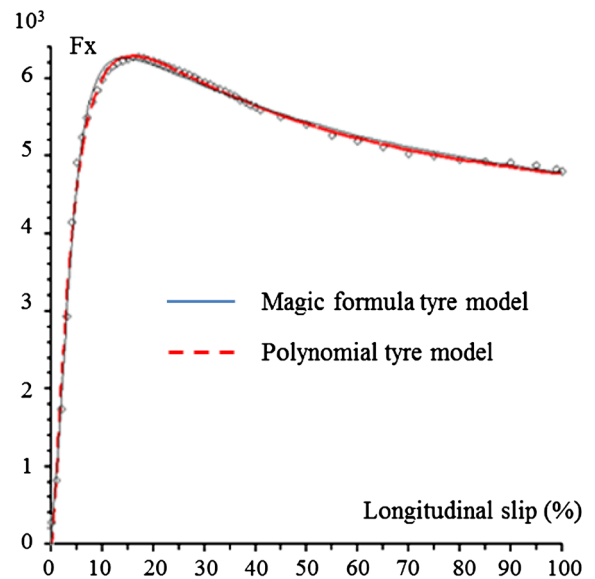
**Table 2** Convergence of the magic formula tyre model parameters

Step	$d$	$C$	$B$	$E$	$f_x$	$f_y$
Initial value	6,000	1.5	0.1	0.5	0	0
1st	5,391.76309	1.55625	0.10651	0.57746	629.05012	-0.68770
2nd	5,095.2668	1.59421	0.11195	0.60858	952.19540	-1.03375
5th	4,649.90929	1.66874	0.12350	0.64867	1,481.00715	-1.55596
10th	4,380.61616	1.73016	0.13279	0.67963	1,834.55287	-1.87485
25th	4,237.11611	1.76466	0.13870	0.70050	2,023.81211	-2.04387
50th	4,226.99866	1.76624	0.13914	0.70194	2,035.43956	-2.05542
75th	4,226.87976	1.76625	0.13915	0.70195	2,035.56032	-2.05555
100th	42,26.87842	1.76625	0.13915	0.70195	2,035.56163	-2.05555
...						
Final value	4,226.87840	1.76625	0.13915	0.70195	2,035.56164	-2.05555

very much slowly and the use of a step's reduction factor  $\alpha = 1/5$  in this case for a correct convergence. Anyway, the time of computing is very reduced.

Then, we show the curves with the results. Firstly, we can see the longitudinal force; in Fig. 3, we present the test points, and the curves of our polynomial model (dotted thick line) and the magic formula tyre model (thin line). Both models approximate the test points quite well.

If we analyse the deviations in Fig. 4, we can observe that our polynomial model (circles) presents a better adjustment in the high area of the curve and in the final part, with moderate, medium and high values of slip. Pacejka's magic formula tyre model (cross marks) presents a better adjustment in the first points of the test. The behaviour of the polynomial tyre model is especially accurate in the area of the maximum value of the curve.



**Fig. 3** Longitudinal force (N) interpolation

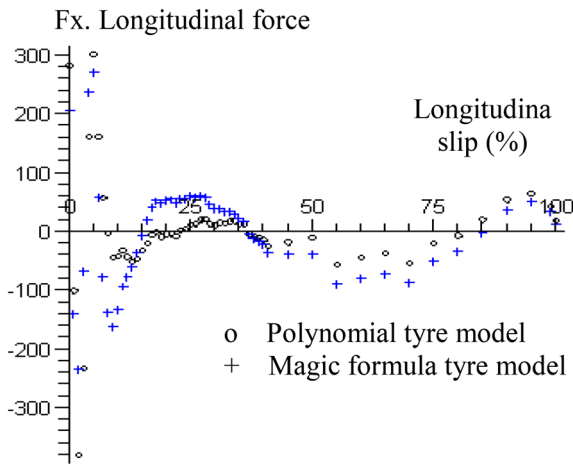


Fig. 4 Absolute deviations (N). Longitudinal force

**Lateral Force**

Data test for the same previous tyre is the following:

Slip Angle = -15 . . . 15 (In this test, we have taken the right-hand side of the file from 0 to 15° (0,262 rad)

Camber Angle = 0°. Normal load  $F_z = 6$  kN. Pure Lateral Force  $F_y$  is measured.

Data = [[0, -98], [0.004, 108], [0.009, 253], [0.013, 480], [0.017, 700], [0.022, 912], [0.026, 1094], [0.031, 1275], [0.035, 1441], [0.039, 1641], [0.044, 1820], [0.048, 1984], [0.052, 2163], [0.057, 2357], [0.061, 2531], [0.065, 2675], [0.07, 2823], [0.074, 2970], [0.079, 3108], [0.083, 3251], [0.087, 3373], [0.092, 3500], [0.096, 3603], [0.1, 3694], [0.105, 3790], [0.109, 3877], [0.113, 3957], [0.118, 4031], [0.122, 4097], [0.127, 4159], [0.131, 4203], [0.135, 4256], [0.14, 4293], [0.144, 4329], [0.148, 4366], [0.153, 4411], [0.157, 4438], [0.161, 4475], [0.166, 4505], [0.17, 4527], [0.174, 4566], [0.192, 4646], [0.209, 4725], [0.227, 4780], [0.244, 4773], [0.262, 4752]:

As we can see in Fig. 5, the curves show a good adjustment to data test for both models. In the polynomial model, we have not imposed constraints. If we look at the deviations in Fig. 6, we can appreciate a slightly better result of the polynomial model, especially in the starting and final areas.

**8 Multivariate analysis**

The main variables influencing the behaviour of a tyre are typically normal load ( $F_z$ ), camber angle ( $a_c$ ) and

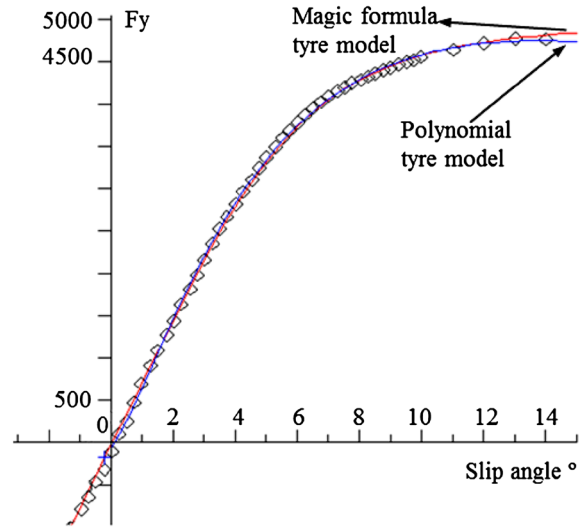


Fig. 5 Lateral force (N). Interpolation

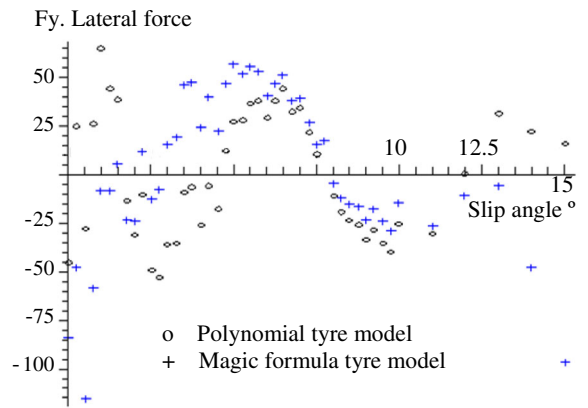


Fig. 6 Absolute deviations (N). Lateral force

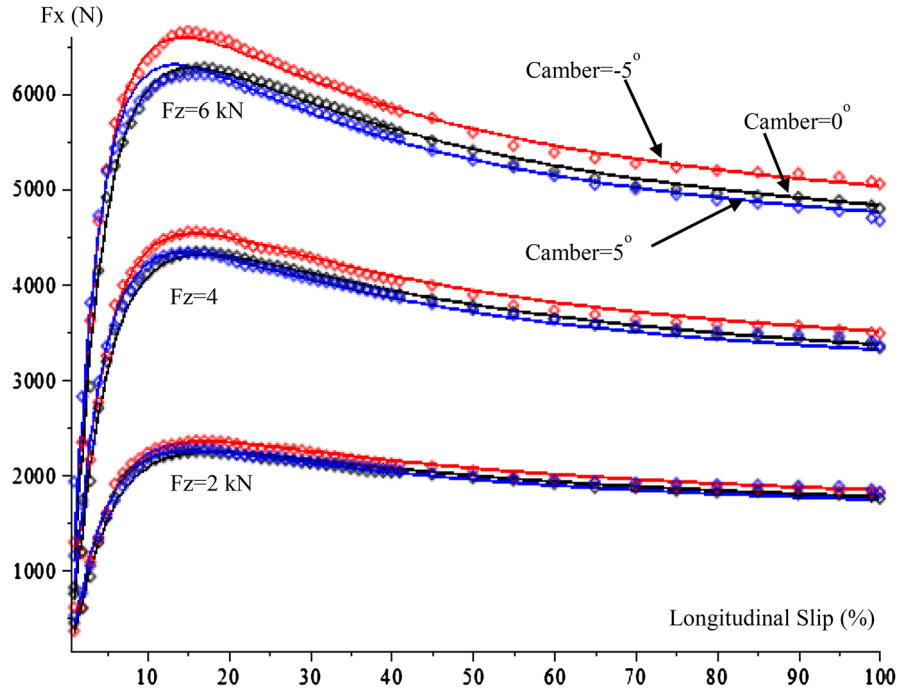
of course longitudinal slip or slip angle. We will have a set of  $m = m_1.m_2.m_3$  test points in the variables  $x$ ,  $F_z$  and  $a_c$ , being  $x$  longitudinal slip or slip angle. Every test point is a vector of four elements  $(x_i, F_{zi}, a_{ci}, y_i)$ . From the observation of the cloud of test points, we build a model in which the normal load and camber influence the peak value and the shape factor  $b$ ; in the multivariate domain we will denote it as  $B^*$ .

$$F_x(x, F_z, a_c) = F^*(F_z, a_c) \cdot \left( A_0 + A_1 \cdot \frac{x}{x + B^*(F_z, a_c)} + A_2 \cdot \left( \frac{x}{x + B^*(F_z, a_c)} \right)^2 + A_3 \cdot \left( \frac{x}{x + B^*(F_z, a_c)} \right)^3 \right);$$

$$F^*(F_z, a_c) = F_1(F_z) \cdot F_2(a_c);$$

$$B^*(F_z, a_c) = B_1(F_z) \cdot B_2(a_c) \tag{5}$$

**Fig. 7** Longitudinal force (N), in pure slip conditions, versus slip. Multivariate model



Initially, we assume a parabolic model for every factor of influence. After an analysis, we could simplify the models of  $F_1$  and  $B_1$  to linear, and we make the mathematic development with degree 2. We have to avoid redundancy of parameters in the possible independent terms ( $d_0$  and  $e_0$ ,  $b_0$  and  $g_0$ ), with the  $A_i$  terms because the optimization algorithms cannot converge if the parameters are redundant, for that reason, we propose finally the following models:

$$\begin{aligned}
 F_1(F_z) &= 1 + d_1 \cdot F_z + d_2 \cdot F_z^2; \\
 F_2(a_c) &= 1 + e_1 \cdot a_c + e_2 \cdot a_c^2; \\
 B_1(F_z) &= 1 + b_1 \cdot F_z + b_2 \cdot F_z^2; \\
 B_2(a_c) &= g_0 + g_1 \cdot a_c + g_2 \cdot a_c^2;
 \end{aligned}$$

The whole model includes 13 parameters.

There is not a theoretical limit in the number of variables and parameters. The limit can be imposed by the computation times and the problem of finding a good initial point with a big number of variables and parameters.

The residuals' vector in this case is  $r_i = y_i - F_i^*$ ; with  $i = 1 \dots m$  (495 test points in our tyre example). We will have 55 points in each of the three values of normal load (2 kN, 4 kN y 6 kN) and in each of the three values of camber angle ( $-5^\circ$ ,  $0^\circ$  y  $5^\circ$ ). In total,  $m = 55 \cdot 3 \cdot 3 = 495$ . The parameters' vector will have 13 elements:  $\beta$  ( $\beta_1, \dots, \beta_{13}$ ). The addition of quadratic devi-

ations  $S(\beta)$  will have the same expression, (2) but now  $m=495$ .

The expressions of the Gradient, Jacobian matrix and Hessian matrix are the same too (but now with  $j = 13$  parameters and  $m = 495$  points). See expression (4) is the same. All methods described in Sect. 3 are valid.

In Fig. 7, we present the results obtained with the same example tyre. Longitudinal force in pure slip conditions is presented with values of  $F_z$  equal to 2, 4 and 6 kN and camber angles of  $-5^\circ$ ,  $0^\circ$  y  $5^\circ$ . Figure 8 shows the adjustment of longitudinal force near the origin.

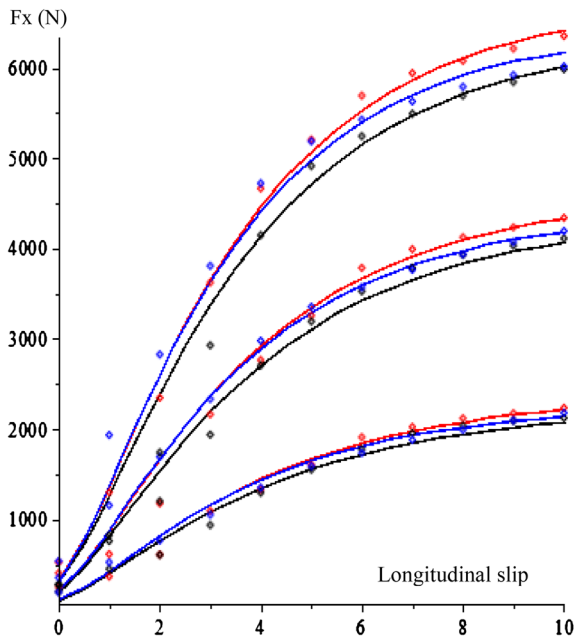
The obtained values of the parameters are the following:

$$\begin{aligned}
 A_0 &= 4.672; A_1 = -11.387; A_2 = 322.675; \\
 A_3 &= -277.546; \\
 b_1 &= -0.2928e - 1; b_2 = -0.86e - 3; \\
 e_1 &= -0.460e - 2; e_2 = 0.113e - 2; \\
 d_1 &= 18.455; d_2 = -.2288; \\
 g_0 &= 6.22766; g_1 = -0.4696e - 1; \\
 g_2 &= -0.2489e - 1;
 \end{aligned}$$

The initial point has been the following:

$$\begin{aligned}
 A_0 &= 5; A_1 = -10; A_2 = 300; A_3 = -300; \\
 b_1 &= b_2 = 0; d_1 = 20; d_2 = e_1 = e_2 = g_1 = g_2 = 0; \\
 g_0 &= 10;
 \end{aligned}$$

The addition of quadratic deviations is  $= 2.695 \cdot 10^6$



**Fig. 8** Adjustment of the slope at the origin

To obtain those results, we have used Quasi-Newton, Gauss–Newton, Nelder–Mead and genetic algorithms, without constraints; all of them converge correctly, but the first three are sensitive to the initial point. The quadratic components of  $F_1(F_z)$  and  $B_1(F_z)$  are very low compared with the linear term, so that we could simplify the model, assuming a linear behaviour for those two functions, by eliminating  $d_2$  and  $b_2$  and repeating the optimization; however, the complete quadratic formulation is presented for a more general expression. We can observe a very good adjustment of the model in respect of test with this real tyre. Then, we present the convergence of those data, using the Gauss–Newton method without step modification.

As we can see in Table 3, we can observe that the proposed model converges easily, and in the eighth step, the error is already under 1%. This is an additional advantage of the use of polynomials.

If we need to adjust the slope at the origin, according to the point 5, we would apply the constraint pointed in 5-R5, but for the multivariate case. The coefficient  $A_1$  would be modified with  $F'_0$  calculated from a parabolic regression with the first 4 points and calculating the slope at the origin. We repeat this three times, one for each of the values of normal load with camber zero, and

finally, a linear regression of the slope at the origin is obtained in function of  $F_z$ , with the three values of the slope. We assume that the slope at the origin changes with normal load  $F_z$ , but not with camber angle, for every value of  $F_z$ . A zoom of the slope at the origin is presented:

We apply the condition of fixed positive derivative for every value of  $F_z$  in Eq. (5).

$$\left(\frac{\partial F^*}{\partial x}\right)_{x=0} = \frac{F^*}{B^*} \cdot A_1 = F'_0 \cdot F_z; A_1 = F'_0 \cdot F_z \cdot \frac{B^*}{F^*}$$

$F'_0 \cdot F_z = 90$ .  $F_z$  is the estimation, for this tyre, of the slopes at the origin from the first 4 points of every curve ( $a_c = 0$ ). The final model is the following:

$$F_x(x, F_z, a_c) = F^* \cdot \left( A_0 + F'_0 \cdot F_z \cdot \frac{B^*}{F^*} \cdot \left( \frac{x}{x + B^*} \right) + A_2 \cdot \left( \frac{x}{x + B^*} \right)^2 + A_2 \cdot \left( \frac{x}{x + B^*} \right)^3 \right);$$

This constraint adds very few computing load because  $B^*$  and  $F^*$  are used in the rest of the model too.

### 9 Conclusion

In this article, a new tyre model is validated using test data of real tyres. Initially, it has been presented the theoretical background of approximation of functions that allow to come up with the model, and the mathematical analysis of the curve proposed as a tyre model by the authors, a degree three polynomial in a simple rational function (1). Then, the article reviews the nonlinear numerical optimization methods, which calculate the model parameters from real test data. Initially, a basic model in five parameters was used, and then, the complete model in 13 parameters, including the effect of normal load and camber angle was also optimized. We could observe a very fast convergence in both cases. The technique of optimization with constraints in the function or its derivatives is applied to the tyre optimization, which it have to be used to avoid strange values of the slope at the origin.

As a conclusion, it can be stated that, during the process of nonlinear multivariate optimization, to calculate its parameters, the new polynomial tyre model presents very good conditions of convergence, faster and simpler than in the magic formula tyre model.

As we could see at Sect. 4.3, there is a huge difference in the simplicity and computational efficiency



**Table 3** Convergence of the multivariate polynomial tyre model

Step	$Cp0$	$Cp1$	$Cp2$	$Cp3$	$b_1$	$b_2$
Initial value	5	-10	300	-300	0	0
1	2.3278	75.761	38.233	-78.112	-0.0291	0.000379
2	3.8508	-35.201	342.114	-273.317	-0.02874	-0.001214
3	5.2796	151.061	-44.590	-59.817	-0.00789	-0.004034
4	4.5607	-16.555	334.262	-285.345	-0.02974	-0.000796
5	4.5253	-3.505	296.722	-259.620	-0.03175	-0.000460
6	4.7024	-10.901	323.034	-278.253	-0.02885	-0.000918
7	4.6706	-11.413	322.679	-277.527	-0.02931	-0.000857
8	4.6721	-11.386	322.677	-277.549	-0.02927	-0.000861
9	4.6720	-11.388	322.675	-277.547	-0.02928	-0.000861
10	4.6720	-11.387	322.675	-277.547	-0.02928	-0.000861
...						
50	4.6720	-11.387	322.675	-277.547	-0.02928	-0.000861

Step	$e_1$	$e_2$	$d_1$	$d_2$	$g_0$	$g_1$	$g_2$
Initial value	0	0	20	0	10	0	0
1	-0.006569	0.001446	24.717	-0.36417	8.2915	-0.02256	-0.01780
2	-0.005044	0.001039	19.637	-0.26208	2.6606	-0.05481	-0.02375
3	-0.004386	0.000992	13.518	-0.14467	6.0431	-0.11890	-0.04808
4	-0.004694	0.001068	17.817	-0.21798	7.5201	-0.07438	-0.03072
5	-0.004569	0.001127	18.767	-0.23390	6.1952	-0.03589	-0.02262
6	-0.004589	0.001134	18.351	-0.22721	6.2423	-0.04775	-0.02515
7	-0.004599	0.001131	18.458	-0.22886	6.2267	-0.04693	-0.02488
8	-0.004599	0.001131	18.455	-0.22882	6.2277	-0.04696	-0.02489
9	-0.004599	0.001131	18.456	-0.22883	6.2277	-0.04696	-0.02489
10	-0.004599	0.001131	18.456	-0.22883	6.2277	-0.04696	-0.02489
...							
50	-0.004599	0.001131	18.456	-0.22883	6.2277	-0.04696	-0.02489

between the Jacobian matrix terms of the polynomial model and the magic formula model, and this is the reason why the observed convergence of the method is much faster in the polynomial model (see for example Tables 1, 2 in Sect. 7, the MF model requires 10 times more steps to obtain stable values of the parameters) and of course, every step is computed very much faster because the terms are much simpler.

This is an additional advantage added to its obvious properties of mathematical simplicity.

As it is a polynomial, the computing is very fast and easy, as we could see in Sects. 4.3 and 3.1; it is also very accurate as we shown in the figures of Sects. 7 and 8. The error relative to the magic formula tyre model is always very reduced, the reduced difference between

the two mathematical formulae had been proven in [17], and in this paper, we can see the similar optimization results compared with real test data, in the figures of Sect. 7.

**Acknowledgments** This work has been financed by the Dirección General de Universidades (Comunidad de Madrid) and the Instituto Madrileño de Desarrollo IMADE.

**References**

1. López, A.: Predicción de Situaciones Inseguras en Vehículos Automóviles. PhD Thesis. Universidad Politécnica de Madrid (1994)
2. Moriano, C.: Método de procesamiento rápido de las ecuaciones de la dinámica vehicular. PhD Thesis. Universidad Antonio de Nebrija (2013)

3. López, A., Moriano, C.: Simulation of the suspension of a vehicle using Chebyshev series. *P I Mech. Eng. D-J Auto.* 954407013513569, first published on January 22, (2014). doi:[10.1177/0954407013513569](https://doi.org/10.1177/0954407013513569)
4. Amirouche, F.M.L., Tung, C.W.: Dynamic collision avoidance and constraint violation compensation. *Nonlinear Dyn.* **7**(3), 385–401 (1995)
5. Ferrara, A., Paderno, J.: Application of switching control for automatic pre-crash collision avoidance in cars. *Nonlinear Dyn.* **46**(3), 307–321 (2006)
6. Ferrara, A., Vecchio, C.: Collision avoidance strategies and coordinated control of passenger vehicles. *Nonlinear Dyn.* **49**(4), 475–492 (2007)
7. Pacejka, H.B.: *Tire and Vehicle Dynamics*. Elsevier, Oxford (2002)
8. Bakker, E., Nyborg, L., Pacejka, H.: Tyre Modelling for Use in Vehicle Dynamics Studies. SAE Technical Paper 870421 (1987). doi:[10.4271/870421](https://doi.org/10.4271/870421)
9. Bakker, E., Pacejka, H., Lidner, L.: A New Tire Model with an Application in Vehicle Dynamics Studies, SAE Technical Paper 890087 (1989). doi:[10.4271/890087](https://doi.org/10.4271/890087)
10. Fox, L., Parker, I.B.: *Chebyshev Polynomials in Numerical Analysis*. Oxford University Press, London (1968)
11. Abramowitz, M., Stegun, I.A.: *Handbook of Mathematical Functions*, 9th edn. Dover, New York (1972)
12. Clenshaw, C.W., Curtis, C.W.: A method for numerical integration on an automatic computer. *Numer. Math.* **2**, 187–205 (1960). doi:[10.1007/BF01386223](https://doi.org/10.1007/BF01386223). QA241.N9 251
13. Valdvoegel, J.: Fast construction of the Fejer and Clenshaw-Curtis quadrature rules *BIT. Numer. Math.* **46**, 195–202 (2006). doi:[10.1007/S10543-006-0045-4](https://doi.org/10.1007/S10543-006-0045-4)
14. Remez, E.Y.: Sur la détermination des polynômes d'approximation de degré donnée. *Comm. Soc. Math. Kharkov* **10**, 41 (1934)
15. Powel, M.J.D.: *Approximation Theory and Methods*. Cambridge University Press, Cambridge (1981)
16. Totik, V.: Orthogonal Polynomials. *Surv. Approx. Theory* **1**, 70–125 (2005)
17. López, A., Vélez, P., Moriano, C.: Approximations to the Magic Formula. *Int. J. Auto. Tech-Kor.* **11**(2), 155–166 (2010); KSAE
18. López, A., Vélez, P., Moriano, C.: Método de procesamiento rápido de las ecuaciones de la dinámica vehicular mediante polinomios de Chebyshev. *Encuentro Internacional de Algebra Computacional*. EACA (2006)
19. López, A., Vélez, P., Moriano, C.: Bivariate Chebyshev expansion of the Pacejka's tyre model. In: 16–20 AIP Conference Proceedings International Conference on Numerical Analysis and Applied Mathematics (ICNAAM), 9/6/2007, vol. 936 Issue 1, p. 347 (2007). doi:[10.1063/1.2790148](https://doi.org/10.1063/1.2790148)
20. Nocedal, J., Wright, S.J.: *Numerical Optimization*, 2nd edn. Springer, New York (2006)
21. Marquardt, D.: An algorithm for least-squares estimation of nonlinear parameter. *SIAM J. Appl. Math.* **11**(2), 431–441 (1963). doi:[10.1137/0111030](https://doi.org/10.1137/0111030)
22. Van Oosten, J.J.M., Bakker, E.: Determination of magic tyre model parameters. *Veh. Syst. Dyn.* (1992). doi:[10.1080/00423119208969995](https://doi.org/10.1080/00423119208969995)
23. Broyden, C.G.: A class of methods for solving nonlinear simultaneous equation. *Math. Comput. (Am. Math. Soc.)* **19**(92), 577–593 (1965)
24. Broyden, C.G.: Quasi-Newton methods and their application to function minimisation. *Math. Comput.* **21**, 368–381 (1967). doi:[10.1090/S0025-5718-1967-0224273-2](https://doi.org/10.1090/S0025-5718-1967-0224273-2)
25. NAG Fortran Library Routine Document: [http://www.nag.co.uk/numeric/fl/manual19/pdf/E04/e04fdf\\_fl18.pdf](http://www.nag.co.uk/numeric/fl/manual19/pdf/E04/e04fdf_fl18.pdf)
26. Davidon, W.C.: Variable metric method for minimization. *SIAM J. Optim.* **1**, 1–17 (1991). doi:[10.1137/0801001](https://doi.org/10.1137/0801001)
27. Fletcher, R.: *Practical Methods of Optimization*. Wiley, New York (1987)
28. Byrd, R.H.: Analysis of a symmetric rank-one trust region method. *SIAM J. Optim.* **6**(4), 1025–1039 (1996)
29. Gay, D.M.: Some convergence properties of Broyden's method. *SIAM J. Numer. Anal.* **16**(4), 623–630 (1979)
30. Kvaalen, E.: A faster Broyden method. *BIT.* **31**(2), 369–372 (1991). doi:[10.1007/BF01931297](https://doi.org/10.1007/BF01931297)
31. Sun, W.Y., Yuan, Y.: *Optimization Theory and Methods: Nonlinear Programming*. Springer, New York (2006)
32. Nelder, J.A., Mead, R.: A simplex method for function minimization. *Comput. J.* **7**, 308–313 (1965). doi:[10.1093/comjnl/7.4.308](https://doi.org/10.1093/comjnl/7.4.308)
33. Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM J. Optim.* **9**(1), 112–147 (1998)
34. Powell, M.J.D.: On search directions for minimization algorithms. *Math. Program.* **4**, 193–201 (1973)
35. McKinnon, K.I.M.: Convergence of the Nelder–Mead simplex method to a non-stationary point. *SIAM J. Optim.* **9**, 148–158 (1999) (Algorithm summary online). doi:[10.1137/S1052623496303482](https://doi.org/10.1137/S1052623496303482)
36. Cabrera, J.A., Ortiz, A., Carabias, E., Simón, A.: An alternative method to determine the magic formula tyre model parameters using genetic algorithms. *Veh. Syst. Dyn.* **41**(2), 109–127 (2004). doi:[10.1076/vesd.41.2.109.26496](https://doi.org/10.1076/vesd.41.2.109.26496)
37. Ortiz, A., Cabrera, J.A., Guerra, A.J., Simon, A.: An easy procedure to determine magic formula parameters: a comparative study between the starting value optimization technique and the imma optimization algorithm. *Veh. Syst. Dyn.* **44**(9), 689–718 (2006)
38. Holland, J.H.: *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Michigan (1975)
39. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, Massachusetts (1989)
40. Balaram, B., Narayanan, M.D., Rajendrakumar, P.K.: Optimal design of multi-parametric nonlinear systems using a parametric continuation based genetic algorithm approach. *Nonlinear Dyn.* **67**(4), 2759–2777 (2012)
41. Garatti, S., Bittanti, S.: Parameter Estimation in the Pacejka's tyre model through the TS method. In: 15th IFAC Symposium on System Identification, Prentice Hall (2009). doi:[10.3182/20090706-3-FR-2004.00217](https://doi.org/10.3182/20090706-3-FR-2004.00217)
42. Garatti, S., Bittanti, S.: Estimation of white-box model parameters via artificial data generation: a two-stage approach. In: Proceedings of the 17th IFAC World Congress, Seoul, South Korea (2008)