Wear 271 (2011) 453–461

Contents lists available at ScienceDirect

Wear

journal homepage: www.elsevier.com/locate/wear

Multibody modeling of railway vehicles: Innovative algorithms for the detection of wheel–rail contact points

Stefano Falomi*, Monica Malvezzi, Enrico Meli

Department of Energy Engineering S. Stecco, University of Florence, Via S. Marta 3, 50100 Firenze, Italy

ARTICLE INFO

Article history:
Received 8 September 2010
Accepted 3 October 2010
Available online 12 October 2010

Keywords:
Wheel–rail contact
Neural networks
Real-time simulations
Multibody simulation

ABSTRACT

The multibody simulation of railway dynamics needs a reliable and efficient method to properly describe the contact between wheel and rail.

In this work innovative methods to evaluate the position of contact points are presented. The aim is to develop a method which can be implemented on-line, assuring a calculation time consistent with real-time calculations of multibody dynamics. At the same time it has to be very accurate, to properly predict the local forces at contact in order to describe even the wear of contact surfaces.

In this work the authors present two different approaches to find stationary points during a multibody simulation. In the former the conditions to define a local minima are wrote in an analytical way. This makes possible to combine the conditions in order to reduce the analytic problem's dimension and then to solve numerically the problem with a low computational burden. The latter approach calculates the location of local minima using a method based on neural networks.

The paper will cover the details of the proposed methods and the performances, in terms of computation time and accuracy, will be compared with those of the conventional algorithms used by commercial softwares, showing their reliability and low computational burden. Moreover, an implementation of the proposed models in a multibody simulator will be presented, in order to show their suitability for this application.

1. Introduction

Numerical simulations of system dynamics are today a standard in the design of railway vehicles. Their typical applications are the suspension kinematics, handling performance and ride comfort as well as the generation of load data for lifetime prediction. One of the key points in this type of simulations is the model of the wheel–rail interaction, which means the definition of the forces exchanged between the wheels and the rail in the contact points. The direction and the magnitude of the contact forces depend on the number and the location of the contact points. The procedure that allows to define the geometry of the contact has then a significant effect on the reliability of the simulation. The aim of the work is to develop a method for the evaluation of the positions of all contact points between wheel and rail, which can be implemented in a multibody simulator of the railway vehicle dynamics. Different solutions of this problem are present in the literature and are implemented in commercial multibody softwares (MSC Adams, Simpack, etc.)

In multibody analysis of railway dynamics there are two different approaches in simulation of wheel–rail contact: the rigid contact formulation and the semi-elastic contact description. In the rigid approach the contact between the bodies is guaranteed by the constraint equations [1–4]. In the formulations based on the semi-elastic approach, the wheel has six degrees of freedom with respect to the rail, and the normal contact forces are defined as a function of the indentation using Hertz's contact theory or using assumed stiffness and damping coefficients [5,6]. This formulation is known as semi-elastic because it allows compenetration of contact bodies, but it is not fully elastic because deformations in contact bodies are not considered. In literature several methods are present for the evaluation of contact points, based on the minimization of the distance or difference between wheel surface and rail surface. Often substantial hypothesis are applied in order to simplify the geometry of the problem [7,8].

The methods present in the literature and their performances have some limits that reduce their suitability in a reliable and efficient simulation of the dynamics of a railway vehicle, because they are often based on arbitrary assumptions, such as not considering all the degrees of freedom of the wheel, or assigning an arbitrary bound to the number of contact points, or introducing geometric hypotheses on the position of the contact points.

The problem of the individuation of contact points, in a semi-elastic formulation, could be generally represented as the research of the local minima of a real function. The methods available in lit-

* Corresponding author. Fax: +39 0554796342.
E-mail address: falomi@mapp1.de.unifi.it (S. Falomi).

0043-1648/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.wear.2010.10.039

erature to solve this general problem can be classified in two main groups: methods based on the value of the function and methods based on the derivatives of the function. In some preceding works the authors presented a method [9,10] in which the contact points are searched minimizing the difference between the wheel and rail surfaces by means of the Simplex method. These procedures do not introduce additional geometric hypotheses and allow an efficient management of the multiple contacts (up to two contact points for wheel). The challenge of this preceding study was the realization of an efficient multibody model, running in real-time conditions; however the developed solutions did not allow a direct implementation of the research procedure in the multibody model. In other words, the developed solutions were used to generate look-up tables to be used during the simulation of the vehicle dynamics.

The other sort of methods are those based on derivatives, which apply the analytic definition of minimizer: for a real valued function, a point is a minimum when all the first order partial derivatives vanish, and the Hessian matrix, which contains the second order partial derivatives, is positive definite. This approach will be the basis of the innovative methods presented in this paper.

In this work the authors will propose two innovative approaches to determine the wheel–rail contact points. In the former the conditions to define a local minima are written in an analytical way. This makes possible to combine the conditions in order to reduce the analytic problem's dimension and then to solve numerically the problem with a low computational burden; that is why this is referred to as semi-analytic approach. The semi-analytic approach can be considered significantly reliable because considers all degrees of freedom of the wheel with respect to the rail, it does not impose any arbitrary bound to the number of contact points and it does not introduce additional geometric hypotheses on the position of the contact points. Moreover the management of multiple contact points is easy and efficient.

Semi-analytic procedures are more reliable and faster than numerical procedures, so they are more efficient in the creation of look-up tables; the weak spot of these procedures is that the computation time is not yet as small as real time applications needs. The on-line implementation is slower than the off-line implementation, because the calculation time is significantly higher than the time required for the reading of look-up tables.

An application of neural networks to the wheel–rail contact problem is then proposed in this paper in order to further reduce the time of evaluation of contact points. The objective is to approximate the unknown function that relates the relative position between the wheel and the rail to the contact points. This can be done by setting an appropriate value to several weight parameters, which are included in the neural network structure, using a process known as training, which requires a set of informations obtained by measurements or reliable methods. In the proposed implementation the sets of data for the training were obtained by the aforementioned semi-analytic method. The advantages of neural network method are mainly related to the computational performance: no iterative calculations are needed and the analytical form is very simple. The main advantages of the semi-analytic methods are maintained: there is no upper limit to the number of minima (training with semi-analytical methods), but the neural network based implementation requires lower computational time, comparable with the time required for the reading of look-up tables, and then are suitable for an on-line real time implementation. The weak spot of neural networks is that the process of training requires a long calculation time, and it must be done again if the profile of wheel or rail has to be changed. Anyway this process can be performed once for each wheel/rail profile pair and can be easily automated.

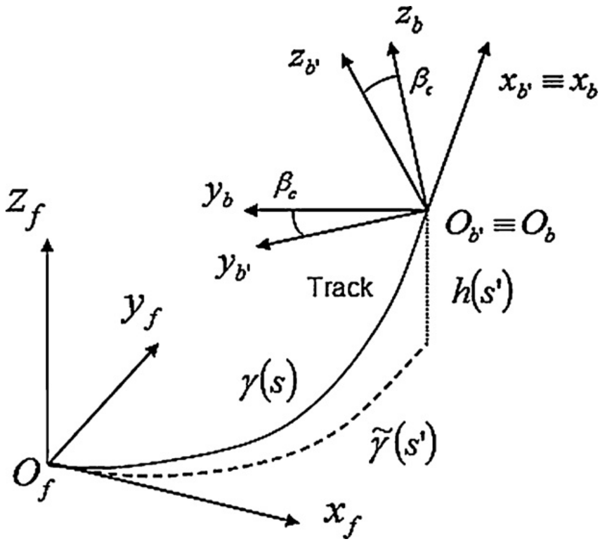

Fig. 1. Definition of the rail track, base and auxiliary reference systems.

2. Track generation and definition of the reference systems

The relative position between wheel and rail is described by means of parameters which relates the relative position of certain coordinate systems defined in this section. First the fixed *global* reference system $O_f\, x_f\, y_f\, z_f$ (Fig. 1) is defined: the x_f axis is tangent to the centerline in the point O_f and the z_f axis is normal to the plane of the rails.

With respect to this fixed *global* system the railway track can be described by means of a three-dimensional curve $\gamma(s)$.

A second reference system (referred as *auxiliary* reference system) $O_b\ x_b\ y_b\ z_b$ (Figs. 1 and 2) is necessary for the problem formulation. It is defined on the rails but follows the wheelset during the simulation.

The x_b axis is tangent to the centerline in the point O_b and the z_b axis normal to the plane of the rails.

In order to define the axes y_b and z_b another reference system $O_{b'}$ $x_{b'}\ y_{b'}\ z_{b'}$ (named *secondary* reference system and shown in Fig. 1) is defined; its unitary vectors are calculated as follows:

$$i_{b'} = i_b = \frac{d\gamma/ds}{\left\|d\gamma/ds\right\|}; \quad j_{b'} = k_f \times i_{b'}; \quad k_{b'} = i_{b'} \times j_{b'} \tag{1}$$

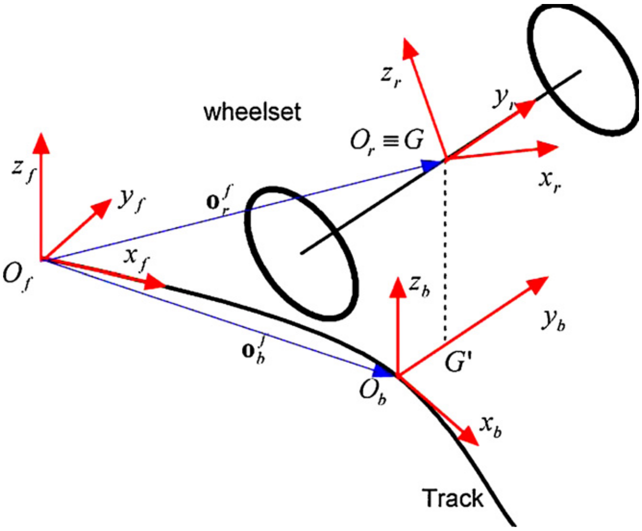

Fig. 2. Track auxiliary reference system and wheelset local reference systems.

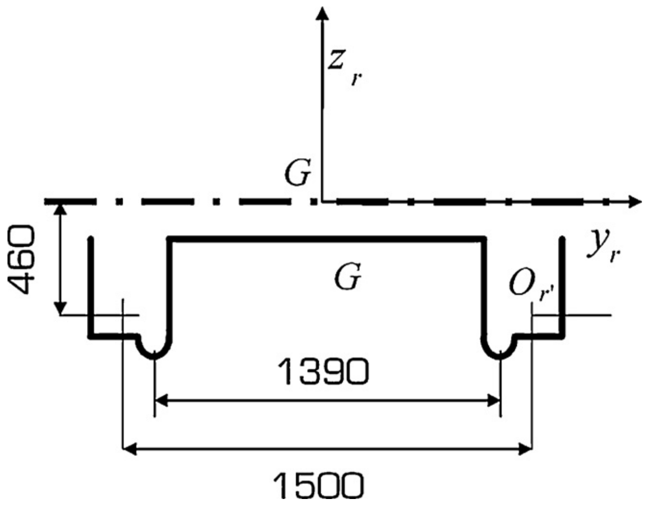

Fig. 3. Generative function of wheelset.

The unitary vectors of the auxiliary system can then be defined as follows:

$$[\, i_b \quad j_b \quad k_b \,] = [\, i_{b'} \quad j_{b'} \quad k_{b'} \,] R_{cant} \tag{2}$$

where the rotation matrix R_{cant} is defined as a function of the cant angle β_c:

$$R_{cant} = R_x(\beta_c) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta_c) & -\sin(\beta_c) \\ 0 & \sin(\beta_c) & \cos(\beta_c) \end{bmatrix} \tag{3}$$

Finally the *local* wheelset reference system $O_r\, x_r\, y_r\, z_r$ is defined. The y_r axis is coincident with the rotation axis of the wheels and is rigidly connected to the axle (except for the rotation around this axis). The x_r axis is contained in the plane $x_b\, y_b$ and the origin coincides with the center of mass $\underline{o}_r$ of the wheelset.

The rotation matrix that links the *local* system with the *auxiliary* one is defined as:

$$[R_2] = [R_z(\alpha)] \cdot [R_x(\beta)] \tag{4}$$

where α and β are respectively the yaw and roll angles of the axle with respect to the track.

In the *local* system the axle (and therefore the wheels) can be described by means of a revolution surface, whose generative function $r(y_r)$ is known and is schematically sketched in Fig. 3.

The following notation will be used in the next sections of the paper: the expression a_x^y means that the variable a is located in the surface x and expressed in the reference system y. In particular, r will denote the axle surface and the *local* reference system, while b will denote the rail surface and the *auxiliary* reference system.

The position of a generic point of the axle in the *local* reference frame has consequently the following analytic expression:

$$\underline{p}_r^r(x_r, y_r) = (\, x_r \quad y_r \quad -\sqrt{r(y_r)^2 - x_r^2}\,)^T \tag{5}$$

while the position of the same in point in the auxiliary reference system is:

$$\underline{p}_r^b(x_r, y_r) = \underline{o}_r^b + [R_2] \cdot \underline{p}_r^r(x_r, y_r) \tag{6}$$

The position of the wheelset center of mass in the auxiliary reference system and the rotation matrix R_2 describe the relative displacement between the wheelset and the rail, that can be represented by means of the displacements G_y and G_z of the center of mass $\underline{o}_r$ with respect to the y_b and z_b directions respectively and the angles α and β previously defined.

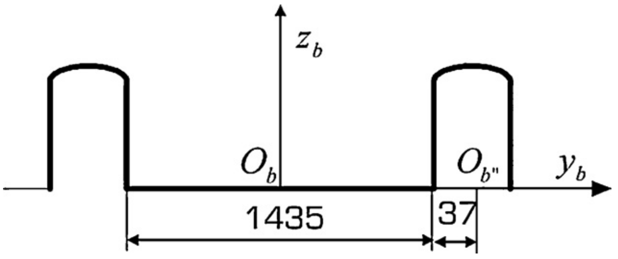

Fig. 4. Generative function of rail.

Similarly the rails can be described in the auxiliary system by means of an extrusion surface. The generative function, indicated with $b(y_b)$ is known and is sketched in Fig. 4.

The position of a generic point of the rail surface in the auxiliary system is:

$$\underline{p}_b^b(x_b, y_b) = (\, x_b \quad y_b \quad b(y_b)\,)^T \tag{7}$$

For both the surfaces the normal unitary vectors (outgoing for convention) can be defined: $\underline{n}_r^r(\underline{p}_r^r)$ is the normal vector to the wheel surface, while: $\underline{n}_b^b(\underline{p}_b^b)$ is the normal vector to the rail surface.

3. Semi-analytic methods

3.1. DIST method

As mentioned in the introduction, in each contact point the distance between the wheel surface and the rail surface assumes a local minimum; this can be defined imposing the following conditions (Fig. 5):

- The unitary normal vectors of wheel and rail surface have to be parallel:

$$\underline{n}_b^b(\underline{p}_b^b) \wedge \underline{n}_r^b(\underline{p}_r^b) = 0 \tag{8}$$

- The normal vectors has to be parallel to the distance $\underline{d}_b = \underline{p}_r^b - \underline{p}_b^b$ between the points of wheel and rail surfaces in which they are applied:

$$\underline{n}_b^b(\underline{p}_b^b) \wedge \underline{d}_b = 0 \tag{9}$$

Only four of the six equations above defined are independent, due to the description of the rail as an extrusion surface and the description of the wheel as a revolution surface. Solutions of this equation's set will be defined with a set of 4 coordinates $(x_{ri}^C, y_{ri}^C, x_{bi}^C, y_{bi}^C)$, while $\underline{p}_{ri}^{b,C} = \underline{p}_r^b(x_{ri}^C, y_{ri}^C)$ and $\underline{p}_{bi}^{b,C} = \underline{p}_b^b(x_{bi}^C, y_{bi}^C)$ will

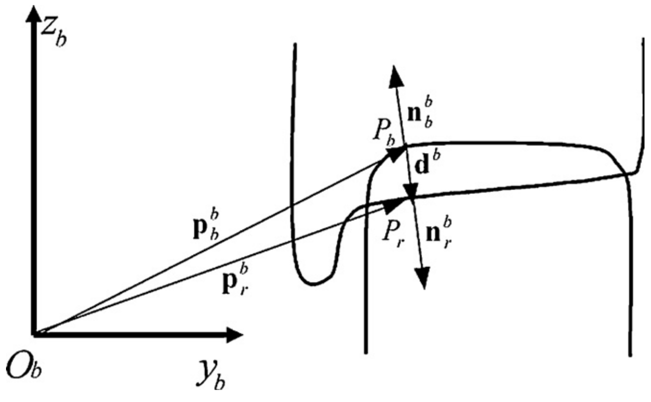

Fig. 5. DIST method: definition of distance between the contact surfaces.

be the contact points respectively on wheel and rail surfaces. Therefore we need to solve a four dimensional problem. The numerical solution of this problem would have a very high computational burden, but, as hinted in the introduction, it is possible to reduce the problem's dimension by combining the four equations defined.

In particular the second component of the vectorial Eq. (8) can be written as:

$$r_{13}\sqrt{r(y_r)^2 - x_r^2} = r_{11}x_r - r_{12}r(y_r)r'(y_r) \tag{10}$$

where r_{ij} is the (i,j) component of the rotation matrix R_2 and $r'(y_r)$ is the derivative of the function $r(y_r)$.

Squaring both members of Eq. (10) it is possible to obtain x_r as a function of y_r. To make equations more readable, Eq. (10) is rewritten in the form:

$$A\sqrt{B^2 - x_r^2} = Cx_r - D \tag{11}$$

Because of the second power of x_r in Eq. (10), two different solutions $x_{r1,2}$ will be obtained for each value of y_r:

$$x_{r1,2}(y_r) = \frac{CD \pm \sqrt{C^2D^2 - (C^2 + A^2)(D^2 - A^2B^2)}}{C^2 + A^2} \tag{12}$$

It is important to remember that squaring both members of Eq. (10) to obtain $x_{r1,2}(y_r)$ lead to the introduction of additional solutions, which are to be excluded because they do not represent effective contact points. So checks on solutions will be performed at the end of this section.

The first component of Eq. (8) can be manipulated obtaining the equation:

$$b'_{1,2}(y_b) = \frac{r_{21}x_{r1,2}(y_r) - r_{22}r(y_r)r'(y_r) - r_{23}\sqrt{r(y_r)^2 - x_{r1,2}(y_r)^2}}{r_{32}r(y_r)r'(y_r) + r_{33}\sqrt{r(y_r)^2 - x_{r1,2}(y_r)^2}} \tag{13}$$

As it is possible to see, two different values of $b'(y_b)$ can be calculated for each value of y_r.

In railway applications, due to the rail geometry, the function $b'(y_b)$ is invertible, then we can calculate $y_{b1,2}(y_r)$.

Finally the second component of Eq. (9) can be written as:

$$x_{b1,2}(y_r) = \underline{r}_1 \cdot \underline{p}_r^r(x_{r1,2}(y_r), y_r) \tag{14}$$

where $\underline{r}_1$ is the first column of matrix R_2.

With this procedure, we obtained an expression of x_r, y_b and x_b as a function of y_r. By substituting these expressions in the first component of Eq. (9), it is possible to obtain two scalar equations in y_r (N.B. two equations are obtained because the expressions of x_r, y_b and x_b as a function of y_r are not unique, but we have 2 expressions for each variable: $x_{r1,2}(y_r)$, $y_{b1,2}(y_r)$ and $x_{b1,2}(y_r)$). These equations are in the form:

$$F_{1,2}(y_r) = 0 \tag{15}$$

which can be easily solved numerically, because finding the zeros of a monodimensional function is a well-known problem in numerical analysis. The solutions of these equations are the coordinates which define the contact points.

As previously stated, checks on solutions are needed to avoid additional incorrect solutions due to the solving of Eqs. (8) and (9). In particular, the following conditions need to be verified for the ith solution:

- x_{ri}^C has to be a real number
- $\sqrt{r(y_{ri}^C)^2 - x_{ri}^{C2}}$ has to be a real number
- (x_{ri}^C, r_{ri}^C) need to be an effective solution of Eq. (10)

Moreover, a generic solution can be an effective contact point only if the contact surfaces are penetrating there, so a check on indentation is needed:

$$\underline{n}_b^b(\underline{p}_b^{b,C}) \cdot \underline{d}_b^C \le 0 \tag{16}$$

An additional check on curvatures of contact surfaces is needed: the generic solution has to verify the following conditions:

$$\begin{aligned} k_{1bi}^C + k_{1ri}^C &> 0 \\ k_{2bi}^C + k_{2ri}^C &> 0 \end{aligned} \tag{17}$$

where the subscript 1 refers to the longitudinal curvature while the subscript 2 refers to the lateral curvature. Curvature is positive when the surface is convex [11].

It is important to stress that the position of contact points depends on 4 variables, which are physically expressed by the parameters that define the relative position between wheel and rail; these parameters are gathered in vector $\underline{\tilde{\phi}} = [\,\alpha \quad \beta \quad G_y \quad G_z\,]^T$.

3.2. DIFF method

The DIFF method is based on the idea that the contact points minimize the difference $D(x_{ri}, y_{ri})$:

$$\nabla D(x_r, y_r) = \underline{0} \tag{18}$$

which is the difference between the wheel surface and the rail surface in the direction identified by the unitary vector $\underline{k}_b$, defined as the third unitary vector of the *auxiliary* reference frame. This means that the DIFF method finds the local minima of the vertical component of the distance between wheel and rail, while the DIST method, described in the previous chapter, searches the local minima of the distance between wheel and rail, considering all the three components of the distance vector. This means that the DIFF method is an approximation of the DIST method, in which only the vertical component is considered. In the next paragraphs the consequences of this approximation will be described. The function $D(x_{ri}, y_{ri})$ is defined as:

$$D(x_r, y_r) = (\underline{p}_r^b(x_r, y_r) - \underline{p}_b^b(x_r^b, y_r^b)) \cdot \underline{k}_b \tag{19}$$

Minimization of the function $D(x_{ri}^C, y_{ri}^C)$ requires that the gradient of D vanishes and that the Hessian matrix $H_D(x_{ri}^C, y_{ri}^C)$ is positively defined.

The function D, applying its definition, is expressed as:

$$\begin{aligned} D(x_r, y_r) &= z_r^b(x_r, y_r) - b(y_r^b(x_r, y_r)) \\ &= G_z + \underline{r}_3 \cdot \underline{p}_r^r(x_r, y_r) - b(G_y + \underline{r}_2 \cdot \underline{p}_r^r(x_r, y_r)) \end{aligned} \tag{20}$$

By imposing the vanishing of the two partial derivatives of D, two scalar equations are obtained. Combining them it is possible to express x_r as a function of y_r. Substituting this expression on the other equation lead to a simple scalar equation in the form:

$$F(y_r) = 0 \tag{21}$$

which can be easily solved numerically.

A generic solution of Eq. (21) is an effective contact point only if the contact surfaces are penetrating there, so a check on indentation is needed:

$$\underline{n}_b^b(\underline{p}_b^{b,C}) \cdot \underline{d}_b^C \le 0 \tag{22}$$

It is very important to notice that in Eq. (20), which gives the complete expression of $D(x_{ri}, y_{ri})$, the parameter G_z is an additive constant. So, by deriving this expression with respect to x_r and y_r, this term will vanish. This means that the equations obtained by the vanishing of partial derivatives depend on α, β and G_y, but they do not depend on G_z. So the position of local minima, which are

Table 1
Configuration variables, variability range.

Variable	Min value	Max value	Step
α [rad]	0	$\pi/180$	$\pi/14{,}400$
β [rad]	$-\pi/240$	$\pi/240$	$\pi/19{,}200$
G_y [mm]	−10	10	0.125

potential contact points, depends only on 3 variables, gathered in vector $\underline{\phi} = [\,\alpha \quad \beta \quad G_y\,]^T$.

As it is stated before, the DIFF method represents an approximation of the DIST method. As we can see, this approximation leads to the fact that the position of local minima depends only on 3 variables, gathered in vector $\underline{\phi}$, instead of the 4 variables in vector $\underline{\phi}$.

This means that DIFF method allows the calculation of local minima with a lower computational burden, if compared with the DIST method. However it is important to understand how this approximation affects the results in terms of the position of local minima. A comparison was conducted considering 2099601 different relative positions between wheel and rail, described in Table 1.

The comparison highlighted that in some configurations, the DIFF method led to an unacceptable error in the evaluation of the position of a local minimum of the distance between contact surfaces. This problem can be easily explained considering that when the contact point is close on the wheel flange, the angle between the normal vectors to contact surfaces and the vertical direction is quite large (up to 70°). Due to this high contact angle, the difference between normal distance (analyzed in the DIST method) and vertical distance (analyzed in the DIFF method) becomes particularly considerable, thus leading to an error in the output of the DIFF method. However the error in the position of a local minimum was higher than 2 mm in less than 0.01% of the configurations. This modest error is acceptable if we consider the lower computational burden of this method. That is why the DIFF method is considered a reliable method, and it is often considered as a benchmark in our investigations.

4. Neural networks

4.1. Theoretical aspects

In the preceding section we described two different deterministic methods to solve the wheel/rail contact problem; these methods gave excellent results in terms of precision, but the computation times, even if far lower than all the numerical procedures tested by the authors, were significantly higher than the reading of look-up tables. The implementation of a model based on neural network was developed in order to find a faster algorithm, that could be implemented on-line, assuring low computational time without the need to store in memory large look-up tables. The semi-analytical procedures, because of their superior performances with respect to the numerical procedures, will be used to generate the reference data needed to define the Neural Networks model.

The identification of a function using neural networks requires three steps. The first step is the collection of reference data; each datum is a vector that contains an allowed input vector and the corresponding desired output vector. Then the user has to choose the network's architecture (organization of neurons in the network and definition of the activation function for each neuron). Finally the network has to be trained: reference data are submitted to the network and the values of parameters are updated in an iterative process in order to minimize the distance between the network's output and the desired output. The distance can be defined in several manners; we decided to use as a measure of the distance the *mean square error* (*mse*).

The chosen architecture is a multilayer perceptron, using in the training process the *Levenberg–Marquardt* algorithm.

In the previous sections two different semi-analytic procedures has been focused: the DIST method and the DIFF method. Concerning their precision in the location of the contact points, these methods can be considered equivalent (as will be shown in the following section), then both of them could be used to train the networks.

Anyway the DIFF method has been chosen because of its simpler structure.

In the wheel–rail contact problem the input is the relative position between wheel and rail, described by the displacement vector and the rotation matrix R_2, while the output is the position of all contact points.

Because of the geometry of the problem, there are two parameters that do not affect the position of the contact points: the rotation of the axle about its axis and its translation in the track direction. The other parameters, which values affect the position of contact points, can be gathered in the vector $\underline{\tilde{\phi}}$.

The substantial difference between the previously described analytic methods is that while in the DIST the value of G_z is needed for the localization of the contact points, in the DIFF method this parameter is used only to check the indentation, in order to determine if a local minimum is an effective contact point.

If the DIFF method is used as reference for the training of the neural network, the function that has to be identified by the network depends on the parameters contained in the vector $\underline{\phi}$. A neural network based on the DIFF method will have three inputs, while if it was based on the DIST method it would have four inputs. In order to obtain a simpler structure of the network and a higher efficiency the DIFF method have then be chosen as reference for the definition and the training of the network.

4.2. Implementation

4.2.1. Classification of the configurations on the basis of the number of outputs

The aim of the presented work is to create a neural network that fits properly the unknown function that relates the position of all local minima to the relative position between wheel and rail. A standard neural network has a fixed number of outputs, defined by the user. The function that the network has to fit in this particular application has a variable number of outputs, depending on the configuration. The number of outputs is the product between the number of local minima (which depends on the configuration) and the number of coordinates used to define the position of each local minimum (usually four: ($x_{rk}^M, y_{rk}^M, x_{bk}^M, y_{bk}^M$)). Firstly, the classification is needed, that evaluates the number of local minima for the examined configuration; then L neural networks are created, where L is the maximum number of local minima that can be obtained for all the allowed configurations. Each network has $4 \times k$ outputs, with $k = 1, \ldots, L$. The classification selects the neural network that has to be applied: when the classification estimates that in a configuration there are k local minima, the neural network with the proper number of output is selected.

In the presented application the ranges in the space of configurations with 1, 2, ..., L local minima were directly detected. Figs. 6 and 7 show the results obtained using the wheel profiles ORE S1002 and rail profile UIC60 with laying angle α_p 1/40; the first one is obtained with $\alpha = 0$, while the second is obtained with $\alpha = \pi/180$ rad. The no filled area represents the range in which only one local minimum is present, while the light gray identify the configurations with 2 minima, and the dark gray area those with 3 minima. In Fig. 6 we can see that for $\alpha = 0$ only configurations with 1 or 2 local minima are found, while Fig. 7 shows that for $\alpha = \pi/180$ rad there exists a very narrow region (enclosed in the ellipse on the

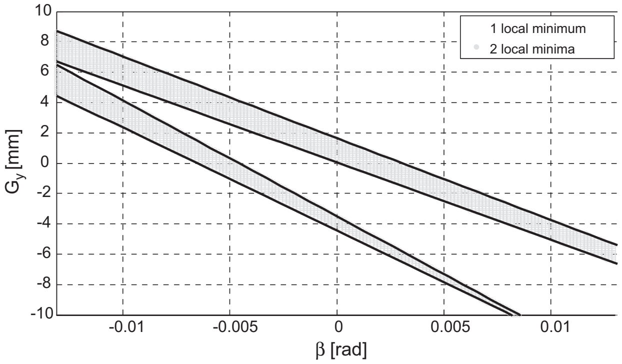

Fig. 6. Number of local minima ($\alpha_p = 1/40$, $\alpha = 0$).

left) in which 3 local minima are found. Straight lines represent the approximation of regions with different number of local minima with planes. As it can be seen the approximation is good for $\alpha = 0$, while when $\alpha = \pi/180$ there is a small region in which the classification fails, that is enclosed for clarity in a dashed rectangle in Fig. 7. The same considerations are valid when the laying angle is 1/20. For the sake of brevity, the corresponding figures are not included in this paper.

It can be observed from Figs. 6 and 7 that the dependence on the angle α is weak (if α varies from 0 to $\pi/180$ rad = 1°), so the partition in domains can be performed in a 2D domain, with dependence on β and G_y. It can be furthermore noted that the domains can be simply separated, with a small error, using straight lines.

It is important to clarify that we analyzed the dependence on α considering a range from 0° to 1°; in this range we found a weak dependence on α; however, if α is bigger (about 2° or 3°), the dependence on α is no more negligible, as it is showed in [12]. In spite of those results, we decided not to consider the dependence on α because in our simulations α is almost always less than 1°.

In order to evaluate the accuracy of the classification, the percentage error on classification E_c was calculated on a set of more than 2 millions of different configurations (those reported in Table 1), choosing the DIFF method as a reference; every time that the classification procedure gave in output a number of minima which was different from that calculated by the DIFF method, a number of error equal to the difference of these two numbers were defined.

For the analyzed configurations, the error E_c is 0.91% for $\alpha_p = 1/40$, and it is 0.85% when $\alpha_p = 1/20$. In both cases the total error is lower than 1%, then the approximated classification can be considered sufficiently accurate.

Once the classification of the configurations on the basis of the number of outputs has been realized, a proper neural network for the localization of the contact points has to be defined for each case.

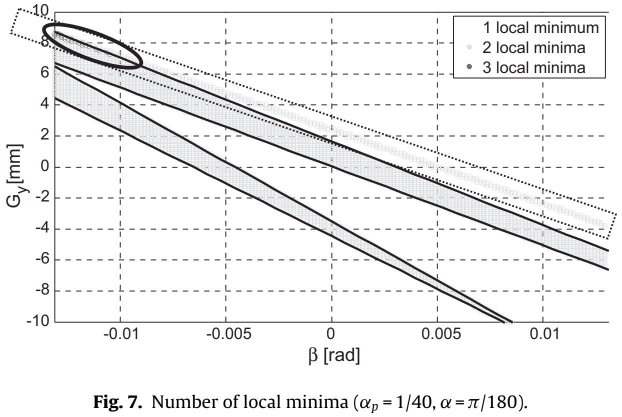

Fig. 7. Number of local minima ($\alpha_p = 1/40$, $\alpha = \pi/180$).

Authors found that for a laying angle $\alpha_p = 1/40$ there can be up to 3 contact points; however configurations with 3 contact points are quite rare, so in the classification the configurations were divided in 2 groups, with 1 and 2 contact points respectively. According to this classification, two neural networks were then defined. For $\alpha_p = 1/20$, the configurations with 4 local contact points were neglected, and the classification procedure divides the configurations in three groups corresponding to 1, 2 and 3 contact points; in this case three different neural networks were then defined.

4.2.2. *Definition of neural networks*

The performances of a neural network depends on its architecture; for the examined problem double-layer networks have been chosen, with hyperbolic tangent activation functions in the hidden layer and linear activation functions in the output layer.

The performances of a network are evaluated analyzing the errors on the test set. For the kth network which outputs are the coordinates of k contact points, for each configuration, an error is defined when the distance between the contact point locations calculated by the DIFF method and the contact points calculated by neural network is more than a specified tolerance, which in our application was set to 1 mm. The percentage error for the kth network on a test set which contains Q_k^t configurations is then given by the ratio between the number of errors identified by the above mentioned algorithm and the total number of contact points calculated for the analyzed configurations:

$$E_k = \frac{\sum_{j=1}^{Q_t^k} e_j}{k \cdot Q_t^k} \tag{23}$$

where e_j is the number of errors on the jth configuration.

From the informations on the number of contact points for each configuration and the errors of NN the total error of the proposed algorithm can be evaluated:

$$E_{NN} = \frac{\sum_{k=1}^{L} E_k \cdot k \cdot A_k}{\sum_{k=1}^{L} k \cdot A_k} \tag{24}$$

where L is the maximum number of contact points per configuration ($L = 3$ if $\alpha_p = 1/40$, $L = 4$ if $\alpha_p = 1/20$), E_k the percentage error for the kth network and A_k the available data for the kth net.

In order to simplify the training, the symmetry on α has been considered: if $x_{bi}^M, y_{bi}^M, x_{ri}^M, y_{ri}^M$ is a local minimum for the configuration $[\,\alpha \quad \beta \quad G_y\,]^T$, thus $-x_{bi}^M, y_{bi}^M, -x_{ri}^M, y_{ri}^M$ is a local minimum for the configuration $[\,\alpha \quad \beta \quad G_y\,]^T$.

Networks were trained using the informations on position of minima for 2099601 different wheel–rail relative positions, obtained varying α, β and G_y in ranges summarized in Table 1, for two different values of laying angle $\alpha_p = 1/40$ and $\alpha_p = 1/20$).

The output of the network is the position of the contact points in the rail surface. Moreover, because the coordinate z_b can be obtained from the y_b according to the relation $z_b = b(y_b)$, the network outputs are the coordinates x_b, y_b for each contact point.

The *Levenberg–Marquardt* algorithm has been chosen for the training, using as Performance Function the *mean square error* on training set data. A limit of 150 epochs for the training process has been set, imposing that it would be stopped earlier if the *mse* on validation set would increase for 5 consecutive epochs (*early stopping*).

4.2.3. *Training*

As previously discussed, for a laying angle $\alpha_p = 1/40$ two networks are necessary; the first gives 2 outputs (coordinates of a single point), while the second gives 4 outputs (coordinates of 2 points). For $\alpha_p = 1/40$ three networks are needed, the first gives 2 outputs (coordinates of a single point), the second gives 4 outputs (coordinates of 2 points), while the third gives 6 outputs (coordinates of three points). As it is stated in the previous chapter, data about the position of local minima for 20099601 configurations were available. In order to reduce the computational load, only a subset of the reference configurations was used in the training process.

The percentage error E_k for each network vary from 0.3% to 4%. From the informations on the number of contact points for each configuration and the errors of each network, the total error of the proposed algorithm can be evaluated.

For $\alpha_p = 1/40$, the calculated global percentage error is $E_{NN} = 2.12\%$ while for $\alpha_p = 1/20$ the error is $E_{NN} = 2.41\%$.

5. Numerical results

In order to analyze the performance of the developed methods for the identification of the wheel/rail contact points, they were implemented within the simulation of the dynamics of a railway vehicle. The objective of this analysis is to check the reliability of the proposed models and to evaluate their numerical efficiency. Because of the little difference in results between semi-analytic methods and neural networks, we decided to show only the results of one method. So, the results of the simulations performed with the neural network method implemented on-line in the developed model were compared with those obtained with a model realized with a commercial multibody software (SIMPACK 8.902).

5.1. The performances of the new methods

In this section the performances of the new procedures for the detection of the contact points will be compared with those of other methods previously developed [9,10].

All methods except DIST are based on the minimization of the surface $D(x_r, y_r)$; in this problem, the position of contact points does not depend on the parameter G_z. Therefore, the configurations on which the methods have been compared are obtained varying only the parameters G_y, α and β.

Nevertheless, in the comparison between DIST and DIFF method the value of G_z is necessary; its value has been chosen, once the value of the other parameters was set, in order to have, in correspondence of the contact points, normal indentations p_n physically acceptable. In this case the bound $p_n \leq p_l = 0.33$ mm has been defined, the limit value p_l has been calculated through the Hertz theory assuming a maximum normal load of 10^5 N applied on a single contact point.

In order to evaluate the performances of the different algorithms, a procedure was defined in which every single algorithm was tested against a reference procedure. We chosen as a reference the GRID (G) method [9,10] that requires the tabulation of the function and find eventual local minima by the comparison of the tabulated values. This method has a very low efficiency because it requires the calculation of the value of the function in a great number of points, but is very reliable. An error is computed every time the tested procedure fails to find a contact point defined by the reference procedure (this means that the tested method did not find this point, or calculated its coordinates with an error higher than a predefined tolerance), and every time the tested method calculates a contact point which was not found by the reference procedure (again within a predefined tolerance). The tested procedures are the multidimensional numerical iterative ones, such as Simplex (S) and Compass Search (CS), the semi-analytic methods such as DIST (d) and DIFF (D) and the Neural Network model (NN). Table 2 summarizes the results of the comparison. Tolerance was set to 2 mm. The acronyms in the first column refer to those already defined: the acronym on the left refers to the tested method, while the acronym on the right refers to the reference method, which is always the GRID method.

Table 2
Global error (reference = GRID; tolerance = 2 mm).

Methods	$\alpha_p = 1/40$	$\alpha_p = 1/20$
S–G	3.6%	7.1%
CS–G	3.2%	5.9%
d–G	0.7%	1.5%
D–G	1.1%	2.1%
NN–G	1.4%	2.6%

First, it can be observed that the semi-analytic methods are those with the lower error percentage. Also neural networks are characterized by good performances, while the numerical procedures are those with the worst precision, which is greatly affected by laying angle. But precision is not the only parameter to measure the performances of these methods: the computation times has to be compared in order to understand if these procedures can be implemented in real time procedures. Table 3 summarizes the mean time required to evaluate the contact points in a generic relative wheel–rail configurations. All the times in question have been obtained with a processor Intel Pentium 4 (3.0 GHz). Table 3 includes also the time required for the reading of look-up tables (referred to as LUT), in order to better appreciate the time performances achieved for the developed methods.

The described results allow to conclude that:

- the performances of the DIST (d) and the DIFF (D) methods are similar in terms of precision and computation times;
- the semi-analytic procedures are reliable, and more accurate than the procedures based on the numerical iterative algorithms, which furthermore require an higher computational time;
- the Neural Network model has a proper accuracy, and implies a calculation time that is much smaller than the time required by all other procedures.

The computation time required by the Neural Network model is almost equal to the time required to read look-up tables, so it can be implemented on-line obtaining acceptable calculation times in dynamic analysis. On the other hand it is necessary to train new networks every time we need to modify the profile of one or both the contact surfaces.

5.2. Dynamic simulations

The railway vehicle chosen for the dynamic simulations is the Manchester Wagon whose physical and geometric characteristics are available in literature [9,10,13].

The multibody 3D model of the Manchester Wagon, has been implemented in the MATLAB® computation environment. The model analyzes the dynamic of a railway vehicle, including a contact model based on the Kalker linear theory with saturation [14] (no Fastsim algorithm is considered). The vehicle is composed of

Table 3
Computation times.

Method	Time [s]
GRID	9.3
CS	0.26
S	0.11
d	0.0011
D	0.0006
NN	0.0003
LUT	0.0003

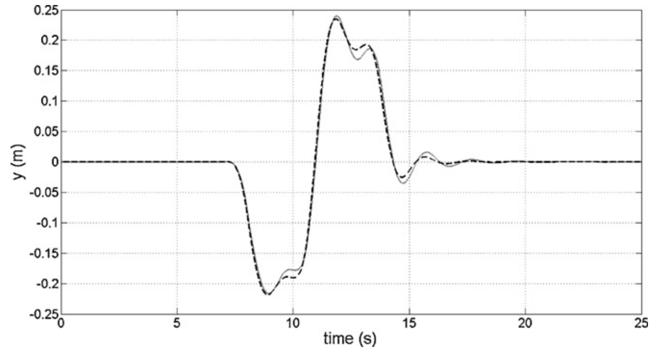

Fig. 8. Lateral displacement (y) of the center of mass of car body.

the car body, two bogies, four axles, primary and secondary suspensions (modeled by three-dimensional nonlinear force elements like bushings, dampers and bumpstops). The wheel profile is the ORE S1002 while the rail profile is the UIC 60, with a laying angle $\alpha_p = 1/40$. The wheel–rail friction coefficient is supposed to be 0.4.

The simulation was performed on a S-shaped curve of radius $R = 190$ m with 4-m-long intermediate tangent track, without irregularities nor superelevation at a velocity of 40 km/h: this scenery reproduces the typical manoeuvre of the train on a railway switch. Each bend is 30 m long (the first is on the right) and the S-shaped curve is preceded by a straight track 50 m long and followed by a straight track 100 m long.

In the following figures, we present a comparison between some results obtained with the SIMPACK 8.902model and the MATLAB® model: for the sake of brevity, we choose to report a selection of curves relating only to some of the most interesting quantities for the running behavior of a train. In every figure, the continuous gray line refers to results obtained with SIMPACK, while the dashed black one refers to MATLAB® results.

It is possible to observe that there is a really good accordance between the two models for the kinematic measurement (see Fig. 8), while Figs. 9 and 10 show some transient differences between the two models. These differences can be explained by the different procedures used by the two models to calculate the contact forces.

The presented results confirm the satisfactory performances in terms of precision of the developed procedures. The multibody model in which these procedures are included give an accurate prediction of the vehicle dynamics. Moreover the comparison in terms of the computational time between the SIMPACK model and the MATLAB® model, shows that the model developed by the authors is faster than the commercial one: Table 4 shows the average computation time with a processor Intel Pentium 4 (3.0 GHz) for a single integration step, using an Ode 5 (Dormand–Prince) integration algorithm (Explicit, Fixed Step, $h = 0.5$ ms).

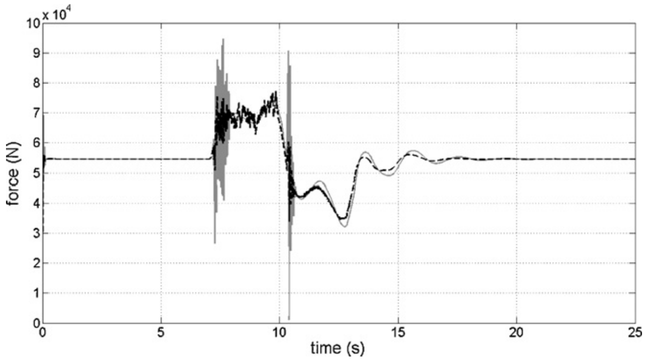

Fig. 9. Forces acting on the first wheelset: left wheel: vertical force (Q).

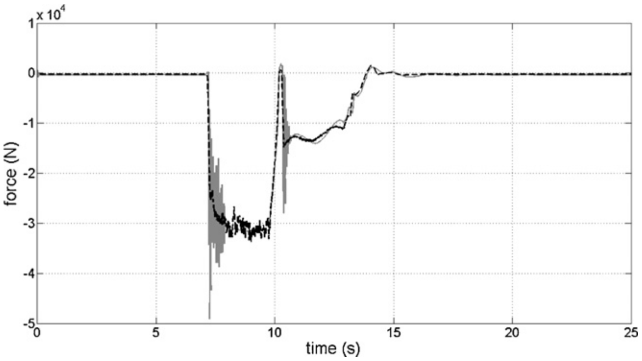

Fig. 10. Forces acting on the first wheelset: left wheel: lateral force (Y).

Table 4
Average computation time for a single integration step.

Model	Time [ms]
SIMPACK 8.902	14.2
MATLAB®	6.2

The multibody model developed by authors has satisfactory performances in terms of precision (Figs. 8–10), but requires less than a half the computation time required by the SIMPACK model.

6. Conclusions

In this work two innovative approaches for the detection of the wheel–rail contact points are presented.

The first is the semi-analytic approach, that considers the wheel and the rail as two mathematical surfaces whose analytic expression is known. This approach has been applied to two different definition of contact points, leading to the development of two different procedures: the first is based on the idea that the contact points minimize the distance between the surfaces and is equivalent to solve an algebraic 4D-system; the second instead is based on the idea that the contact points minimize the difference between the surfaces and is equivalent to solve an algebraic 2D-system. In both cases the original problem has been reduced analytically to a simple mono dimensional problem that is then solved numerically. Since the problem's dimension is one, even elementary non-iterative algorithms like the GRID algorithm (a non-iterative method based on the evaluation of the function in the points of a fixed grid and on the comparison between the obtained results) have shown to be efficient and reliable.

The second approach consists in the application of a black box model, based on neural networks. The aim of this approach is to develop a model reliable as the semi-analytic methods, but requiring a lower calculation time, consistent with real-time constraints of multibody simulations.

The neural network algorithm is composed of a first part in which, on the basis of the wheelset geometric configuration, the number of contact points is defined. Then the location of the contact points is calculated with feedforward neural networks. The networks are trained using the results of semi-analytic procedures based on the minimization of the surface defined as the difference between the wheel surface and the rail surface.

Subsequently the performances of the new procedures have been compared among them and with those of the methods present in the literature. The GRID method and other procedures based on numerical iterative algorithms (like the Compass Search algorithm and the Simplex algorithm) have been considered. The comparison has been carried out in terms of precision and computation times.

The semi-analytic procedures (named DIST and DIFF methods) have similar performances in terms of precision and computation times; both of them are reliable and more accurate and faster than the procedures based on the numerical iterative algorithms; thus they are more efficient in the creation of look-up tables. However these procedures are much slower than the reading of look up tables, so the on-line implementation leads to higher calculation times than the off-line implementation.

The Neural Network model is a less accurate model (but the error does not noticeably affect the multibody simulation), but requires a calculation time that is much smaller than the time required by all other procedures and comparable with the time necessary to read look-up tables, allowing on-line implementations also in real time.

References

[1] A.A. Shabana, J.R. Sany, An augmented formulation for mechanical systems with non-generalized coordinates: application to rigid body contact problems, Nonlinear Dynamics 24 (2001) 183–204.

[2] A.A. Shabana, J.R. Sany, A survey of rail vehicle track simulations and flexible multibody dynamics, Nonlinear Dynamics 26 (2001) 179–210.

[3] A.A. Shabana, M. Berzeri, J.R. Sany, Numerical procedure for the simulation of wheel/rail contact dynamics, Transactions of the ASME 123 (2001) 168–178.

[4] W. Rulka, E. Pankiewicz, MBS approach to generate equations of motion for HiL-simulations in vehicle dynamics, Multibody System Dynamics 14 (2005) 367–386.

[5] A.A. Shabana, K.E. Zaazaa, J.L. Escalona, J.R. Sany, Development of elastic force model for wheel/rail contact problems, Journal of Sound and Vibration 269 (2004) 295–325.

[6] A.A. Shabana, M. Tobaa, H. SugiYama, K.E. Zaazaa, On the computer formulations of the wheel/rail contact problem, Nonlinear Dynamics 40 (2005) 169–193.

[7] J. Pombo, J. Ambrosio, Dynamic analysis of a railway vehicle in real operation conditions using a new wheel–rail contact detection model, International Journal of Vehicle Systems Modelling and Testing 1 (1/2/3) (2005) 79–105.

[8] W. Kik, D. Moelle, Implementation of the wheel–rail element in ADAMS/Rail version 10.1, in: 5th ADAMS/Rail Users' Conference, Haarleem, 2000.

[9] M. Malvezzi, E. Meli, S. Papini, L. Pugi, Parametric models of railway systems for real-time applications, Multibody Dynamics (2007) (Milano, Italy).

[10] J. Auciello, M. Malvezzi, E. Meli, S. Papini, L. Pugi, A. Rindi, Multibody models of railway vehicles for real-time systems, in: XVIII Congresso AIMETA, Brescia, Italy, 2007.

[11] M.P. do Carmo, Differential Geometry of Curves and Surface, Prentice Hall, 1976.

[12] J. Santamaria, E.G. Vadillo, J. Gomez, A comprehensive method for the elastic calculation of the two-point wheel–rail contact, Vehicle System Dynamics 44 (2006) 240–250.

[13] S. Iwnicki, The Manchester Benchmarks for Rail Vehicle Simulators, Swets & Zeitlinger, B.V., Lisse, 1999, ISBN 90 265 1551 0.

[14] J.J. Kalker, Three-dimensional Elastic Bodies in Rolling Contact, Kluwer Academic Publishers, 1990.