FLORE
Repository istituzionale dell'Università degli Studi di Firenze

Determination of wheel–rail contact points: comparison between classical and neural network based procedures

Questa è la Versione finale referata (Post print/Accepted manuscript) della seguente pubblicazione:

Original Citation:

Determination of wheel–rail contact points: comparison between classical and neural network based procedures / S. Falomi; M. Malvezzi; E. Meli; A.Rindi. - In: MECCANICA. - ISSN 0025-6455. - ELETTRONICO. - 44:(2009), pp. 661-686. [10.1007/s11012-009-9202-6]

Availability:

This version is available at: 2158/627626 since:

Published version:

DOI: 10.1007/s11012-009-9202-6

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf)

Publisher copyright claim:

(Article begins on next page)

19 April 2024

Meccanica (2009) 44: 661–686
DOI 10.1007/s11012-009-9202-6

Determination of wheel–rail contact points: comparison between classical and neural network based procedures

Stefano Falomi · Monica Malvezzi · Enrico Meli · Andrea Rindi

Received: 17 June 2008 / Accepted: 23 February 2009 / Published online: 12 March 2009
© Springer Science+Business Media B.V. 2009

Abstract The multibody simulation of railway dynamics needs a reliable efficient method to evaluate the contact points between wheel and rail. In this work some methods to evaluate position of contact points are presented. The aim is to develop a method which is reliable in terms of precision and can be implemented on-line, assuring a calculation time consistent with real-time calculations of multibody dynamics.

Keywords Wheel-rail contact · Neural networks · Semi-analytic methods

1 Introduction

The multibody simulation of a railway vehicle requires a reliable and efficient method to solve the wheel-rail contact problem. From a more general point of view this problem can be represented as the contact between two bodies having generic three dimensional surfaces and can be divided in three main parts: contact geometry, contact kinematics and contact mechanics [1, 2]. Contact geometry deals with the research of the contact points between the bodies while contact kinematics allows to calculate, from the displacement and velocities of the contact bodies, the relative speed in the contact points. The contact mechanics finally calculates the forces exchanged in the contact points between the bodies: their direction and magnitude depend on the location of the contact points and on the relative speed between the contact bodies in these points. The paper describes some studies concerning the first part of the problem: the research of the contact points. The aim of the work is then to develop a method for the evaluation of the contact points between the wheel and the rail, that 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 the 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 [2]. In the rigid approach the contact between the bodies is guaranteed by the constraint equations. Contact points are searched during the dynamic simulation by solving the nonlinear differential and algebraic equations of the constrained multibody system. This approach leads to a model in which the wheel has five degrees of freedom with respect to the rail, because the wheel indentation and lift are not permitted [1, 3]. In the formulations based on the 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 stiff-

S. Falomi · M. Malvezzi (✉) · E. Meli · A. Rindi
Department of Energetics, University of Florence,
via S. Marta n. 3, 50139 Firenze, Italy
e-mail: malvezzi@mapp1.de.unifi.it

ness and damping coefficients [4, 5]. 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 [6, 7].

The methods described 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: in some solutions, not all degrees of freedom of the wheel are considered, often an arbitrary bound to the number of contact points is assigned, the management of multiple contact points can be difficult, additional geometric hypotheses on the position of the contact points are introduced.

In [8] two methods for the evaluation of wheel/rail contact points are presented, in the first one the contact is modeled as a kinematic constraint and thus the vehicle dynamics is described by a DAE (Differential Algebraic Equation) system. This method has been implemented in the Simpack wheel/rail module. The second method is defined as 'quasi-elastic' contact model, the contact point is evaluated as the point in which the distance between the wheel and rail surfaces assumes a local maximum. The crucial point in this approach is the procedure that allows to find the stationary points of a multi-variable function.

In this paper a quasi-elastic approach has been chosen for the evaluation of the contact points. The problem could be generally represented as the research of the local stationary points of a real function. The methods available in literature 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. The methods based on the value of the function can be further divided in: *global methods*, which require the evaluation of the function on a grid, the eventual local minima are found comparing the tabulated values (this method will be referred to as GRID method in the paper), and *local methods*, that are iterative and are known also as *direct search methods* [9]. In the iterative methods a starting point is needed; the function is evaluated in this point and in a certain number of close points. The trial solution for the subsequent step is selected depending on the value of the function in these points. The procedure is stopped when the trial solutions for two subsequent steps are closer than a certain tolerance. There are several methods for the selection of the position of the trial solution for subsequent steps; in order to assess the properties of these procedures, two different algorithms have been implemented: the first is based on the Simplex method proposed by Nelder and Mead [10] (referred to as S method in the paper), that is an iterative method frequently used in practice, while the second is based on the Compass Search (referred to as CS method in the paper), the simpler direct search procedure [9].

In some preceding works the authors presented a method [11, 12] 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 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 lookup tables to be used during the simulation of the vehicle dynamics. The numerical algorithms conventionally used for this type of application present various problems: iterative research algorithms need start points and break conditions, that cannot be defined in a simple way and may affect the reliability of the solution, moreover, the convergence of the procedure cannot be easily assured, the required accuracy cannot be guaranteed a priori, the management of the multiple solutions can be difficult, the required computational burden allows only *off-line* implementation (a real time implementation requires necessarily the use of look-up tables).

Other numerical methods for the definition of the stationary points of a real function are those based on derivatives, which apply the analytic definition of minimizer: for a real valued function $f(\mathbf{x})$, a point is a stationary point when all the first order partial derivatives vanish.

In the first part of this work the authors summarize two semi-analytic methods to determine the wheel-rail contact points. The details of these methods can be found in [13], however, in order to clarify the explanation and to introduce the second part of the work a summary of these procedures is presented in this paper. Both the presented procedures represent the wheel and the rail as two mathematical surfaces whose analytic expression is known (Fig. 1).

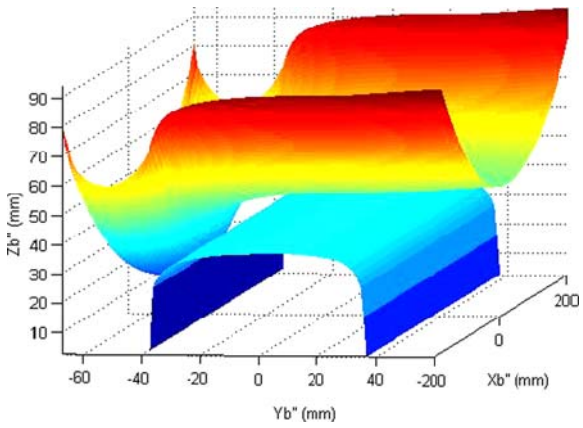


Fig. 1 Wheel and rail surfaces

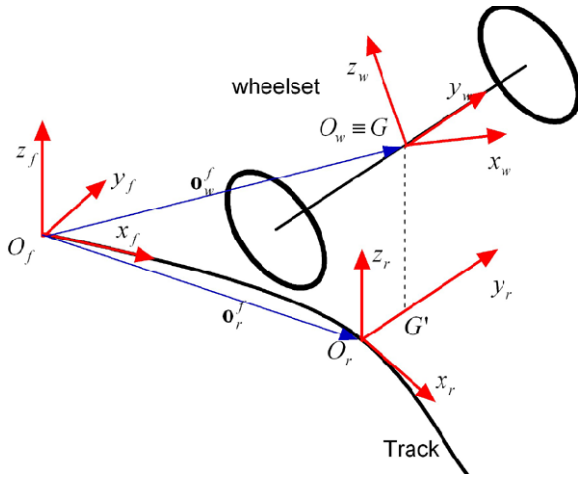


Fig. 2 Track auxiliary reference system and wheelset local reference systems

The first method is based on the idea that the contact points are located in the points in which the *distance* between the wheel surface and the rail surface assumes local maxima. The second method is based on the idea that in the contact points the *difference* between the wheel surface and the rail surface evaluated along a fixed direction assumes local minima. In both cases the original problem can be reduced analytically to a scalar equation with only one unknown that can be solved numerically; that is why they are referred to as semi-analytic methods. This is the main feature that differentiates these methods from the classical procedures described in the literature. The first method will be indicated in the paper with the abbreviation DIST whereas the second with DIFF.

An application of neural networks to the wheel-rail contact problem is then proposed in this paper in order to further reduce the computational time necessary for the evaluation of contact points. Neural networks are non-linear statistical data modeling tools, that can be used to model complex relationships between inputs and outputs or to find patterns in data. This type of approach is often used to model uncertain or complex phenomena, in which the system structure is complex or unknown, while a wide set of input/output pairs are available (for example from experimental data).

The objective of the work presented in this paper is to approximate the unknown function that relates the relative position between the wheel and the rail to the contact points. This type of approach is often referred to as black box modeling. In a model with neural networks, the mathematical model of a problem is decided by the user, disregarding the physical laws which describe the analyzed system. The capacity of the model to reflect the analyzed problem is given by a certain number of parameters, which value is set in a procedure known as training, which needs a certain number of reference data. The training process is iterative, and is stopped when the model, with the current set of parameters, reflects the behavior of the physical problem. In the proposed implementation the sets of data were obtained by the aforementioned semi-analytic method.

The paper is organized as follows: Sect. 2 summarizes the analytic formulation of the problem, while Sect. 3 describes in detail the semi-analytic methods. In Sect. 4 the implementation of neural networks is described, and Sect. 5 presents the results obtained, comparing performances of the proposed procedures other methods previously described.

2 Track generation and definition of the reference systems

A fixed reference system $O_f x_f y_f z_f$ (Fig. 2) is defined, the x_f axis is tangent to the track centerline in the point O_f and the z_f axis is normal to the plane of the rails. With respect to this fixed system the railway track can be described by means of a three-dimensional curve $\mathbf{\Gamma}(s)$, (s is the curvilinear abscissa):

$$\mathbf{\Gamma} : I \subset \mathbb{R} \to \mathbb{R}^3. \tag{1}$$

A second reference system (referred as *auxiliary* reference system) $O_r x_r y_r z_r$ is defined, $\mathbf{i}_r$, $\mathbf{j}_r$ and $\mathbf{k}_r$ represent the unitary vectors relative to the axes x_r, y_r

and z_r respectively. The auxiliary reference system is defined on the rails but follows the wheelset during the simulation. The x_r axis is tangent to the track centerline in the point O_r and the z_r axis normal to the plane of the rails. The position of the point O_r, identified by its coordinates $\mathbf{o}_r^f$ relative to the fixed reference system, can be calculated from the wheelset center of mass location $G \equiv O_w$, identified by its coordinates relative to the fixed reference system $\mathbf{o}_w^f$, imposing the following condition:

$$\left(\mathbf{o}_w^f - \mathbf{o}_r^f\right) \cdot \mathbf{i}_r = 0, \tag{2}$$

since the point O_r is on the curve $\boldsymbol{\Gamma}(s)$ describing the rail track, $\mathbf{o}_r^f = \boldsymbol{\Gamma}(s)$, then (2) can be rewritten as:

$$\left(\mathbf{o}_w^f - \boldsymbol{\Gamma}(s)\right) \cdot \mathbf{i}_r = 0, \tag{3}$$

(3) can be solved with respect to the variable s.

In order to define the axes y_r and z_r another reference system (named *secondary* reference system) is defined, its unitary vectors $\mathbf{i}_{r'}$, $\mathbf{i}_{r'}$ and $\mathbf{i}_{r'}$, relative to the axes $x_{r'}$, $y_{r'}$ and $z_{r'}$ respectively, are calculated as follows:

$$\begin{aligned} \mathbf{i}_{r'} &= \mathbf{i}_r = \frac{d\boldsymbol{\Gamma}}{ds} \Bigg/ \left\| \frac{d\boldsymbol{\Gamma}}{ds} \right\|, \\ \mathbf{j}_{r'} &= \mathbf{k}_f \times \mathbf{i}_{r'}, \\ \mathbf{k}_{r'} &= \mathbf{i}_{r'} \times \mathbf{j}_{r'}. \end{aligned} \tag{4}$$

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

$$\begin{bmatrix} \mathbf{i}_r & \mathbf{j}_r & \mathbf{k}_r \end{bmatrix} = [\mathbf{R}_{cant}] \begin{bmatrix} \mathbf{i}_{r'} & \mathbf{j}_{r'} & \mathbf{k}_{r'} \end{bmatrix}, \tag{5}$$

where the rotation matrix $[\mathbf{R}_{cant}]$ is defined as a function of the cant angle β_c [14]:

$$[\mathbf{R}_{cant}] = [\mathbf{R}_{x,\beta_c}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta_c & -\sin\beta_c \\ 0 & \sin\beta_c & \cos\beta_c \end{bmatrix}. \tag{6}$$

Finally the local reference system $O_w x_w y_w z_w$ is defined. The y_w 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_w axis is parallel to the plane $x_r y_r$ and the origin O_w coincides with the center of mass G of the wheelset.

Indicating with $\mathbf{p}^r$ and $\mathbf{p}^w$ the position of a generic point expressed respectively in auxiliary and local reference systems, then the following standard kinematic relation holds:

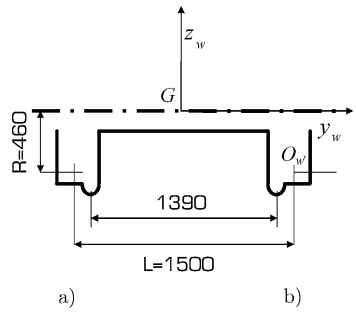


Fig. 3 Generative function of the wheelset

$$\mathbf{p}^r = \mathbf{o}_w^r + [\mathbf{R}_2]\mathbf{p}^w \tag{7}$$

where $\mathbf{o}_w^r$ are the coordinates of the wheelset center of mass expressed in the auxiliary reference system and the matrix $[\mathbf{R}_2]$ (that links the local system with the auxiliary one) is defined as a function of the yaw and roll angles α and β of the axle with respect to the track:

$$[\mathbf{R}_2] = \begin{bmatrix} \cos\alpha & -\sin\alpha\cos\beta & \sin\alpha\sin\beta \\ \sin\alpha & \cos\alpha\cos\beta & -\cos\alpha\sin\beta \\ 0 & \sin\beta & \cos\beta \end{bmatrix}. \tag{8}$$

In order to analytically describe a surface different options can be considered, in this work for both wheel and rail the surfaces two translational parameters have been used, this choice allows to simplify the algebraic manipulations necessary to develop the DIST and DIFF methods described in the following sections.

In the local system the axle (and therefore the wheels) can be described by means of a revolution surface. The generative function, schematically sketched in Fig. 3, is indicated with $r(y_w)$ (the function $r(y_w)$ is known). The profile of the single wheel is plotted in detail in Fig. 4. In this case an ORE S 1002 has been chosen [14]. The position of a generic point of the axle in the local reference frame $\mathbf{p}_w^w$ has consequently the following analytic expression (Fig. 5):

$$\mathbf{p}_w^w(x_w, y_w) = \begin{bmatrix} x_w \\ y_w \\ -\sqrt{r(y_w)^2 - x_w^2} \end{bmatrix} \tag{9}$$

while the position of the same point in the auxiliary system is given by (Fig. 2):

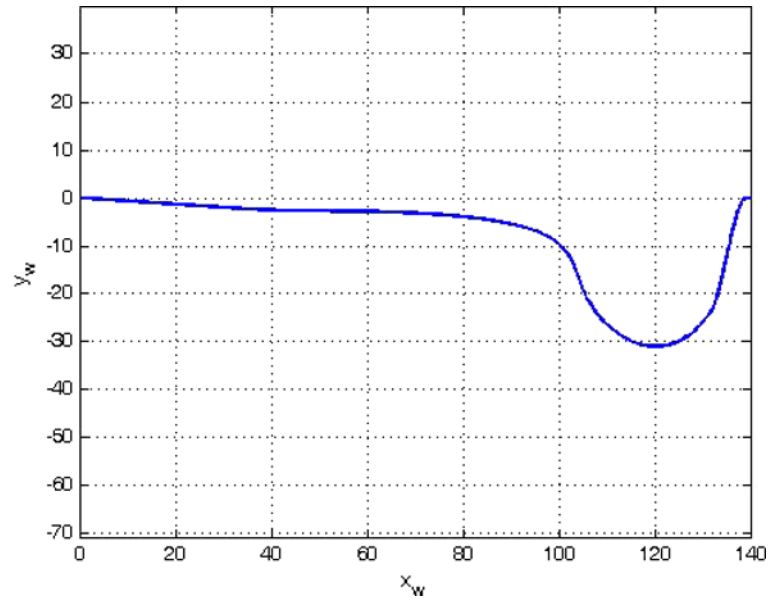


Fig. 4 Profile of the wheel

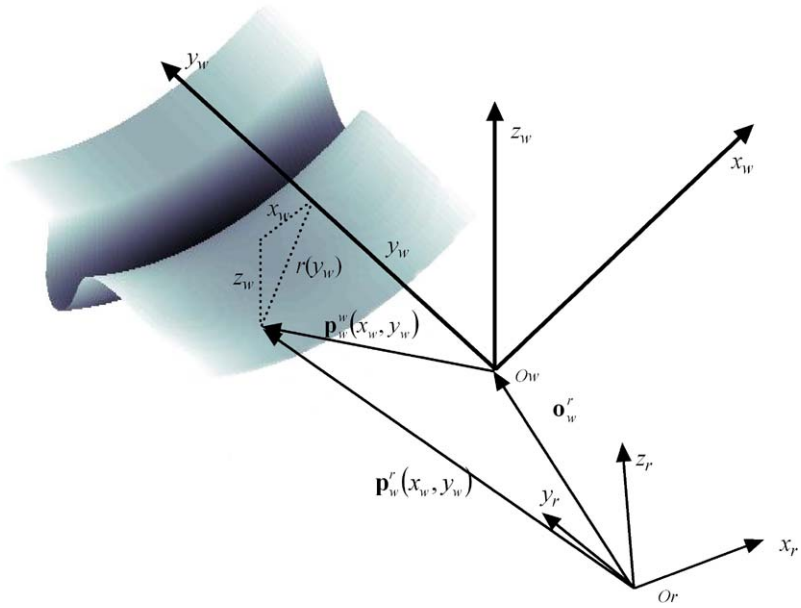


Fig. 5 Coordinates of a point on the wheel surface

$$\mathbf{p}_w^r(x_w, y_w) = \mathbf{o}_w^r + [\mathbf{R}_2]\mathbf{p}_w^w(x_w, y_w) = \begin{bmatrix} x_w^r(x_w, y_w) \\ y_w^r(x_w, y_w) \\ z_w^r(x_w, y_w) \end{bmatrix}; \tag{10}$$

as consequence of the choice of the reference systems the matrix $[\mathbf{R}_2]$, defined in (8) as a function of the wheelset yaw and roll angle, has the following structure:

$$[\mathbf{R}_2] = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ 0 & r_{32} & r_{33} \end{bmatrix} \tag{11}$$

while the coordinates of the wheelset center of mass are:

$$\mathbf{o}_w^r = \begin{bmatrix} 0 \\ G_y \\ G_z \end{bmatrix}. \tag{12}$$

Similarly the rails can be described in the auxiliary system by means of a extrusion surface. The generative function, indicated with $b(y_r)$ is known and is sketched in Fig. 6.

The profile of the single rail is plotted in detail in Fig. 7. This profile is rotated with respect to the x_r axis with an angle α_p corresponding to the railway laying angle, in the figure an UIC 60 is showed [14].

The coordinates of a generic point of the rail in the auxiliary system are:

$$\mathbf{p}_r^r(x_r, y_r) = \begin{pmatrix} x_r \\ y_r \\ b(y_r) \end{pmatrix}. \tag{13}$$

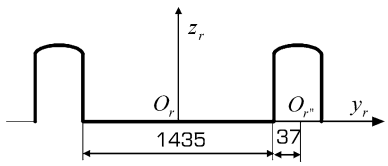


Fig. 6 Railway generative function

For both surfaces the normal unitary vectors (outgoing for convention) can be defined. The normal unitary vector on the wheel surface (Fig. 8) is defined, in the local system, as follows:

$$\begin{aligned}\mathbf{n}_w^w(\mathbf{p}_w^w) &= -\left(\frac{\partial \mathbf{p}_w^w}{\partial x_w} \times \frac{\partial \mathbf{p}_w^w}{\partial y_w}\right) \Bigg/ \left\| \frac{\partial \mathbf{p}_w^w}{\partial x_w} \times \frac{\partial \mathbf{p}_w^w}{\partial y_w} \right\| \\ &= \left(\sqrt{\frac{r(y_w)^2(r'(y_w)^2+1)}{r(y_w)^2 - x_w^2}}\right)^{-1} \\ &\quad \times \begin{bmatrix} x_w/\sqrt{r(y_w)^2 - x_w^2} \\ -r(y_w)r'(y_w)/\sqrt{r(y_w)^2 - x_w^2} \\ -1 \end{bmatrix}. \end{aligned} \tag{14}$$

In this expression $r'(y_w)$ is the wheel profile derivative $r'(y_w) = \frac{dr(y_w)}{dy_w}$. In the auxiliary reference system, the unitary vector normal to the wheel surface can be calculated as:

$$\mathbf{n}_w^r(\mathbf{p}_w^r) = [\mathbf{R}_2]\mathbf{n}_w^w(\mathbf{p}_w^w). \tag{15}$$

The unitary vector normal to the rail surface (Fig. 9), with respect to the auxiliary system is defined as:

$$\mathbf{n}_r^r(\mathbf{p}_r^r) = \left(\frac{\partial \mathbf{p}_r^r}{\partial x_r} \times \frac{\partial \mathbf{p}_r^r}{\partial y_r}\right) \Bigg/ \left\| \frac{\partial \mathbf{p}_r^r}{\partial x_r} \times \frac{\partial \mathbf{p}_r^r}{\partial y_r} \right\| \tag{16}$$

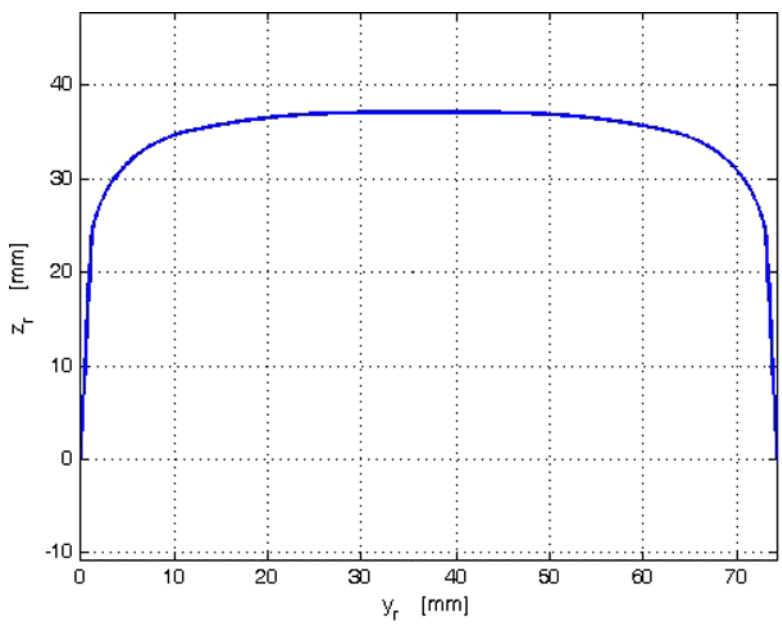


Fig. 7 Rail profile

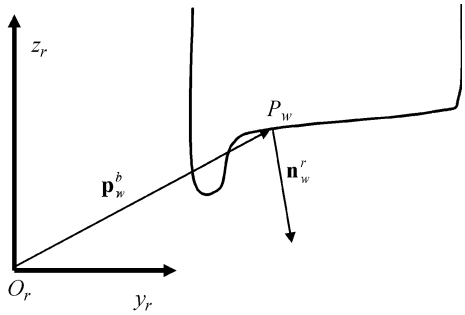


Fig. 8 Normal unitary vector on the wheel surface

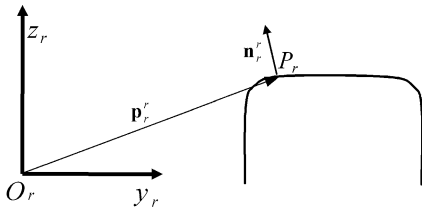


Fig. 9 Normal unitary vector of the rail

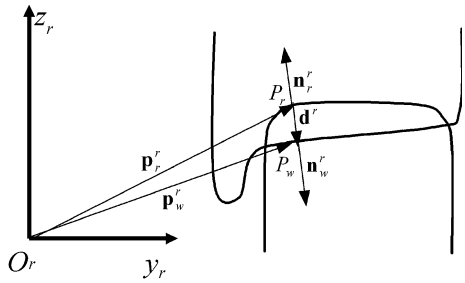


Fig. 10 DIST method: vector representing the distance between the generic point of the wheel and of the rail

$$= \left(\sqrt{1+b'(y_r)^2}\right)^{-1} \begin{bmatrix} 0 \\ -b'(y_r) \\ 1 \end{bmatrix}. \tag{17}$$

In this expression $b'(y_r)$ represents the rail profile derivative $b'(y_r) = \frac{db(y_r)}{dy_r}$.

3 Research of the contact points with semi-analytic methods

3.1 The DIST method

3.1.1 Research of the contact points

As mentioned in the introduction, the DIST method is based on the idea that the in each contact point the distance between the wheel surface and the rail surface assumes a local maximum. The problem can be efficiently solved imposing the following conditions (Fig. 10) [15]:

- the normal unitary vector relative to the rail surface $\mathbf{n}_r^r(\mathbf{p}_r^r)$ has to be parallel to the wheel surface normal unitary vector $\mathbf{n}_w^r(\mathbf{p}_w^r)$:

$$\mathbf{n}_r^r(\mathbf{p}_r^r) \times [\mathbf{R}_2]\mathbf{n}_w^w(\mathbf{p}_w^w) = \mathbf{0}; \tag{18}$$

- the rail surface normal unitary vector $\mathbf{n}_r^r(\mathbf{p}_r^r)$ has to be parallel to the vector $\mathbf{d}^r$ representing the distance between the generic point of the wheel and of the rail:

$$\mathbf{n}_r^r(\mathbf{p}_r^r) \times \mathbf{d}^r = \mathbf{0}. \tag{19}$$

The vector $\mathbf{d}^r$ can be expressed as:

$$\begin{aligned} \mathbf{d}^r(x_w, y_w, x_r, y_r) &= \mathbf{p}_w^r(x_w, y_w) - \mathbf{p}_r^r(x_r, y_r) \\ &= \mathbf{o}_w^r + [\mathbf{R}_2]\mathbf{p}_w^w(x_w, y_w) \\ &\quad - \mathbf{p}_r^r(x_r, y_r). \end{aligned} \tag{20}$$

$\mathbf{d}^r(x_w, y_w, x_r, y_r)$ depends on four parameters, namely the parameters used to identify a point on the rail and on the wheel surface respectively. The conditions defined in (18) and (19) represent a system composed of six equations (since two vectorial constraints are imposed) and four unknowns (x_w, y_w, x_r, y_r), then only four of them are independent. The parallelism constraints expressed in (18) and (19) by means of a cross product could be described imposing the orthogonality between the unitary vectors $[\mathbf{R}_2]\mathbf{n}_w^w(\mathbf{p}_w^w)$ and $\mathbf{d}^r(x_w, y_w, x_r, y_r)$ and the plane tangent to the rail in $\mathbf{p}_r^r$. Each of these conditions can be expressed by means of two dot products, then the over mentioned conditions can be represented by four scalar equations.

The solutions of the system (18)–(19) are indicated with

$$(x_{wi}^C, y_{wi}^C, x_{ri}^C, y_{wi}^C), \quad i = 1, 2, \ldots, n, \tag{21}$$

and the corresponding contact points on the wheel and on the rail (Fig. 10) are;

$$\begin{aligned} \mathbf{p}_{wi}^{r,C} &= \mathbf{p}_w^r(x_{wi}^C, y_{wi}^C), \\ \mathbf{p}_{ri}^{r,C} &= \mathbf{p}_r^r(x_{ri}^C, y_{wi}^C), \quad i = 1, 2, \ldots, n. \end{aligned} \tag{22}$$

The solutions depends on the relative displacement between the wheelset and the rail, defined by the kinematic parameters G_y, G_z, α, β.

Because of the problem geometry, if the conditions $b(y_r) = b(-y_r)$ and $r(y_w) = r(-y_w)$ are satisfied (in other terms if the wheel and the rail profile are symmetric with respect to the y_w and y_r axis), the following symmetry conditions holds:

- if $(x_{wi}^C, y_{wi}^C, x_{ri}^C, y_{wi}^C)$ is a solution associated to the kinematic variables $(G_y, G_z, \alpha, \beta)$, then $(-x_{ri}^C, y_{wi}^C, -x_{ri}^C, y_{ri}^C)$ will be a solution associated to the kinematic variables $(G_y, G_z, -\alpha, \beta)$.
- if $(x_{wi}^C, y_{wi}^C, x_{ri}^C, y_{ri}^C)$ is a solution associated to the kinematic variables $(G_y, G_z, \alpha, \beta)$, then $(-x_{wi}^C, -y_{wi}^C, -x_{ri}^C, -y_{ri}^C)$ will be a solution associated to the kinematic variables $(-G_y, G_z, \alpha, -\beta)$.

3.1.2 Reduction of the problem to a scalar equation

From the second component of the vectorial equation defined in (18) the following expression can be found:

$$r_{13}\sqrt{r(y_w)^2 - x_w^2} = r_{11}x_w - r_{12}r(y_w)r'(y_w). \tag{23}$$

In this expression r_{13}, r_{11} and r_{12} are components of the matrix $[\mathbf{R}_2]$. Indicating for brevity $A = r_{13}$, $B = r(y_w)$, $C = r_{11}$ and $D = r_{12}r(y_w)r'(y_w)$, it can be rewritten as:

$$A\sqrt{B^2 - x_w^2} = Cx_w - D, \tag{24}$$

and therefore, squaring both the members, it can be solved to obtain x_w as a function of y_w:

$$x_{w1,2}(y_w) = \frac{CD \pm \sqrt{C^2D^2 - (C^2 + A^2)(D^2 - A^2B^2)}}{C^2 + A^2}. \tag{25}$$

As it can be seen, there are two possible values of x_w for each value of y_w. From the first component of the vectorial equation defined in (18) the following expression can be found. The index $(_{1,2})$ indicates that the value has to be calculated for both the roots of (25).

$$b'(y_r)_{1,2} = \frac{r_{21}x_{w1,2}(y_w) - r_{22}r(y_w)r'(y_w) - r_{23}\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2}}{r_{32}r(y_w)r'(y_w) + r_{33}\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2}}. \tag{26}$$

The following condition:

$$r_{32}r(y_w)r'(y_w) + r_{33}\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2} \neq 0 \tag{27}$$

is always verified, since:

$$r(y_w)^2 \gg x_w^2, r(y_w) \gg r'(y_w) \quad \text{and} \quad r_{33} \gg r_{32}.$$

Without loss of generality the left and the right side of the track can be considered separately. In the first case $y_w \in [700790]$ mm and $y_r \in [720780]$ mm and in the second $y_w \in [-790\ -700]$ mm and $y_r \in [-780\ -720]$ mm. Under these assumptions the function $b'(y_r)$ is numerically invertible and the values of $y_{r1,2}(y_w)$ can be calculated. Finally from the second component of the vectorial equation defined in (19) the following relation can be written:

$$x_{r1,2}(y_w) = \mathbf{r}_1 \cdot \mathbf{p}_w^w\left(x_{w1,2}(y_w), y_w\right) = r_{11}x_{w1,2}(y_w) + r_{12}y_w - r_{13}\sqrt{r(y_w)^2 - x_{r1,2}(y_w)^2}. \tag{28}$$

The variables x_w, x_r, y_r have been expressed as a function of y_w and can be introduced in the first component of the vectorial equation (19). Then, the following relations can be found:

$$\begin{aligned} F_{1,2}(y_w) = &-b'(y_{r1,2}(y_w)) \\ &\times \left(G_z + \mathbf{r}_3.\mathbf{p}_w^w\left(x_{w1,2}(y_w), y_w\right)\right. \\ &\left. - b(y_{r1,2}(y_w))\right) \\ &- \left(G_y + \mathbf{r}_2.\mathbf{p}_w^w\left(x_{w1,2}(y_w), y_w\right)\right. \end{aligned}$$

$$
\begin{aligned}
&- y_{r1,2}(y_w)\big) \\
&= -b'(y_{r1,2}(y_w))\,(G_z + r_{32}y_w \\
&- r_{33}\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2} \\
&- b(y_{r1,2}(y_w))\big) \\
&- \big(G_y + r_{21}x_{w1,2}(y_w) \\
&+ r_{22}y_w - r_{23}\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2} \\
&- y_{r1,2}(y_w)\big) = 0. \qquad (29)
\end{aligned}
$$

It is a simple scalar equation in the variable y_w (where $y_w \in [700790]$ mm for the left side and $y_w \in [-790\ -700]$ mm for the right side) and can be solved numerically. Also in this case the index $(_{1,2})$ indicates that the equation has to be solved for both the roots of (25).

For simplicity the solutions y^C_{wi} (with $i = 1, 2, \ldots, n$) of (29) are splitted in two groups: y^C_{w1j} (with $j = 1, 2, \ldots, n_1$) indicates the roots of $F_1(y_w) = 0$ (obtained with the first root of (25))and y^C_{w2k} (with $k = 1, 2, \ldots, n_2$) those of $F_2(y_w) = 0$, where $n = n_1 + n_2$. Through (25), (26) and (28) the values of the variables x_w, x_r, y_r corresponding to y^C_{w1j} and y^C_{w2k} can be calculated:

$$
\begin{aligned}
&x^C_{w1j} = x_{w1}(y^C_{w1j}), \qquad x^C_{r1j} = x_{r1}(y^C_{w1j}), \\
&y^C_{r1j} = y_{r1}(y^C_{w1j}), \quad j = 1, 2, \ldots, n_1, \\
&x^C_{w2k} = x_{w2}(y^C_{w2k}), \qquad x^C_{r2k} = x_{r2}(y^C_{w2k}), \\
&y^C_{r2k} = y_{r2}(y^C_{w2k}) \quad k = 1, 2, \ldots, n_2
\end{aligned} \qquad (30)
$$

and therefore, by means of the (9), (10) and (13), the positions of the contact points on the wheel and on the rail are calculated:

$$
\begin{aligned}
&\mathbf{p}^{r,C}_{w1j} = \mathbf{p}^r_w(x^C_{w1j}, y^C_{w1j}), \\
&\mathbf{p}^{r,C}_{r1j} = \mathbf{p}^r_r(x^C_{r1j}, y^C_{r1j}), \quad j = 1, 2, \ldots, n_1, \\
&\mathbf{p}^{r,C}_{w2k} = \mathbf{p}^r_w(x^C_{w2k}, y^C_{w2k}), \\
&\mathbf{p}^{r,C}_{r2k} = \mathbf{p}^r_r(x^C_{r2k}, y^C_{r2k}), \quad k = 1, 2, \ldots, n_2.
\end{aligned} \qquad (31)
$$

Not all the solutions $(x^C_{w1j}, y^C_{w1j}, x^C_{r1j}, y^C_{r1j})$ (with $j = 1, 2, \ldots, n_1$) and $(x^C_{w2k}, y^C_{w2k}, x^C_{r2k}, y^C_{r2k})$ (with $k = 1, 2, \ldots, n_2$) obtained finding the roots of (29) can be accepted, since (23)–(29) contains irrational terms. Consequently the following conditions have to be verified for every j and k:

- x^C_{w1j} and x^C_{w2k} (calculated by (25) as a function of y^C_{w1j} and y^C_{w2k}) have to be real numbers;
- the terms

$$\sqrt{r(y^C_{w1j})^2 - {x^C_{w1j}}^2} \quad \text{and} \quad \sqrt{r(y^C_{w2k})^2 - {x^C_{w2k}}^2}$$

  appearing in (29) have to be real;
- (x^C_{w1j}, y^C_{w1j}) and (x^C_{w2k}, y^C_{w2k}) have to be effective solutions of (23).

If one of these conditions is not verified the solution has to be rejected. Moreover concerning eventual multiple solutions, if m is the multiplicity of the solution only one of these can be considered because the others $m - 1$ have not physical meaning. The multiplicity of the solution has to be evaluated considering all the solutions in both the sets $(x^C_{w1j}, y^C_{w1j}, x^C_{r1j}, y^C_{r1j})$ (with $j = 1, 2, \ldots, n_1$) and $(x^C_{w2k}, y^C_{w2k}, x^C_{r2k}, y^C_{r2k})$ (with $k = 1, 2, \ldots, n_2$).

3.1.3 *Indentation between the contact surfaces*

A generic solution of the system (18)–(19) can be considered an effective contact point only if the normal indentation between the surfaces p_n is negative (according to our convention). Therefore, as regards the i-th solution, the following condition has to be verified:

$$p_{ni} = \mathbf{d}^{r,C}_i \cdot \mathbf{n}^r_r(\mathbf{p}^{r,C}_{ri}) \le 0 \qquad (32)$$

where $\mathbf{d}^{r,C}_i = \mathbf{p}^{r,C}_{wi} - \mathbf{p}^{r,C}_{ri}$.

3.1.4 *Surface curvature*

The curvatures of the surfaces in the contact points have to satisfy some algebraic conditions so that the contact could be physically possible. The generic solution $(x^C_{wi}, y^C_{wi}, x^C_{ri}, y^C_{ri})$ with $i = 1, 2, \ldots, n$ of the system (18)–(19) can be considered an effective contact point only if the following conditions are verified:

$$
\begin{aligned}
&k^C_{1ri} + k^C_{1wi} > 0, \\
&k^C_{2ri} + k^C_{2wi} > 0;
\end{aligned} \qquad (33)
$$

Where k^C_{1wi}, k^C_{2wi}, k^C_{1ri}, k^C_{2ri} are the curvatures of the wheel and rail surfaces in the longitudinal and lateral

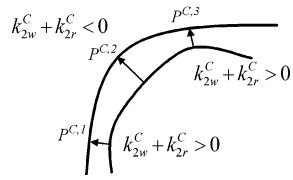


Fig. 11 Normal curvatures in the contact points

rail direction. The situation is shown in Fig. 11 for the second direction, as it can be seen, in the point $P^{C,2}$ the curvatures satisfy the inequality $k_{2ri}^C + k_{2wi}^C < 0$, in other terms in this case the wheel curvature radius is larger than the rail one: even if the point $P^{C,2}$ was a solution of the system (18)–(19) it could not be considered an effective contact point. Because of the geometry of the problem the first of the (33) conditions is always satisfied and thus only the second has to be verified.

The curvatures of the wheel and the rail surfaces in the longitudinal and lateral rail direction can be calculated as a function of the normal principal curvatures [13].

For each wheel/rail configuration the DIST method requires therefore the following steps:

1. determination of the solutions $(x_{wi}^C, y_{wi}^C, x_{ri}^C, y_{ri}^C)$ (with $i = 1, 2, \ldots, n$) of the system (18)–(19);
2. research and elimination of the multiple solutions;
3. check of the analytic conditions;
4. check of the condition on the curvatures;
5. check of the condition on the normal indentation.

These controls allow to verify that the solutions are physically realistic, in the analytic development no simplifications and approximations were assumed then no solution of the problem should be excluded.

3.2 The DIFF method

3.2.1 Research of the contact points

The DIFF method has been developed in order to simplify the method described in the preceding section and then to further improve its computational efficiency. The DIFF method is based on the idea that the contact points minimize the *difference* between the wheel surface and the rail surface in the direction identified by the unitary vector $\mathbf{k}_r$:

$$D(x_w, y_w) = \left(\mathbf{p}_w^r(x_w, y_w) - \mathbf{p}_r^r(x_w, y_w)\right) \cdot \mathbf{k}_r \quad (34)$$

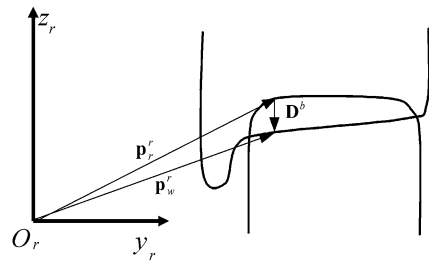


Fig. 12 Minimization of the difference: definition of the difference function

where $\mathbf{p}_w^r$ is defined according to (15), while $\mathbf{p}_r^r(x_w, y_w)$ is defined as follows:

$$\mathbf{p}_r^r(x_w, y_w) = \begin{bmatrix} x_w^r(x_w, y_w) \\ y_w^r(x_w, y_w) \\ b\left(y_w^r(x_w, y_w)\right) \end{bmatrix}. \quad (35)$$

In other terms the point $\mathbf{p}_r^r(x_w, y_w)$ is evaluated as the intersection between the rail surface and a line parallel to the axis z_r passing through the point $\mathbf{p}_w^r$ on the wheel surface (Fig. 12). For each wheel/rail relative configuration, the difference $D(x_w, y_w)$ is then a function depending on two variables (in other terms it is a two dimensional surface). The contact points between the surfaces can be approximated with the minima of the difference function. In some preceding works [11, 12] the authors presented some procedures to find numerically the minima, based on iterative methods (namely Simplex method and Compass search [9, 10]). The efficiency of these methods was not sufficiently high to allow an on line implementation of the procedure.

The problem could be solved analytically imposing the following conditions on the partial derivatives of the function $D(x_w, y_w)$:

$$\frac{\partial D(x_w, y_w)}{\partial x_w} = 0, \qquad \frac{\partial D(x_w, y_w)}{\partial y_w} = 0. \quad (36)$$

Also in this case (x_{wi}^C, y_{wi}^C) with $i = 1, 2, \ldots, n$ indicate the solutions of the system (36), and $\mathbf{p}_{wi}^{r,C} = \mathbf{p}_w^r(x_{wi}^C, y_{wi}^C)$, $\mathbf{p}_{ri}^{r,C} = \mathbf{p}_r^r(x_{wi}^C, y_{wi}^C)$ with $i = 1, 2, \ldots, n$ the corresponding contact points on the wheel and on the rail (for each wheelset/rail configuration, n is the total number of contact points).

Because of the problem geometry, if $b(y_r) = b(-y_r)$ and $r(y_w) = r(-y_w)$, the same symmetry conditions described in the preceding section are verified.

To be considered possible contact points, the solutions of the system (36) have to be effectively a minimum of the surface (34), then the Hessian matrix $H_D(x_w, y_w)$ of $D(x_w, y_w)$, defined as:

$$H_D(x_w, y_w) = \begin{pmatrix} \frac{\partial^2 D}{\partial x_w^2} & \frac{\partial^2 D}{\partial x_w \partial y_w} \\ \frac{\partial^2 D}{\partial x_w \partial y_w} & \frac{\partial^2 D}{\partial y_w^2} \end{pmatrix} \tag{37}$$

has to be positive defined in the points (x_{wi}^C, y_{wi}^C) with $i = 1, 2, \ldots, n$. Since $D(x_w, y_w) : \mathbb{R}^2 \to \mathbb{R}$, this is equivalent to:

$$\frac{\partial^2 D}{\partial x_w^2}(x_{wi}^C, y_{wi}^C) > 0, \qquad \det H_D(x_{wi}^C, y_{wi}^C) > 0 \tag{38}$$

with $i = 1, 2, \ldots, n$.

3.2.2 Reduction of the problem to a scalar equation

The minimization of the difference is equivalent then to solve an algebraic 2D-system (see (36)). However also in this case the problem dimension can be analytically reduced, since it is possible to express the variable x_w as a function of y_w in order to obtain a simple scalar equation in the variable y_w. Introducing the definitions of $\mathbf{p}_w^r(x_w, y_w)$ and $\mathbf{p}_r^r(x_w, y_w)$ into the (34), the following expression can be found:

$$\begin{aligned} D(x_w, y_w) &= \left(\mathbf{p}_w^r(x_w, y_w) - \mathbf{p}_r^r(x_w, y_w)\right) \cdot \mathbf{k}_r \\ &= z_w^r(x_w, y_w) - b\left(y_w^r(x_w, y_w)\right) \\ &= G_z + \mathbf{r}_3 \cdot \mathbf{p}_r^r(x_w, y_w) \\ &\quad - b\left(G_y + \mathbf{r}_2 \cdot \mathbf{p}_w^w(x_w, y_w)\right). \end{aligned} \tag{39}$$

The partial derivatives of the function (36) are:

$$\begin{aligned} &\frac{\partial D}{\partial x_w}(x_w, y_w) \\ &\quad = \mathbf{r}_3 . \frac{\partial \mathbf{p}_w^w}{\partial x_w}(x_w, y_w) \\ &\qquad - b'\left(G_y + \mathbf{r}_2 . \mathbf{p}_w^w(x_w, y_w)\right) \mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w}{\partial x_w}(x_w, y_w) \\ &\quad = 0, \end{aligned} \tag{40}$$

$$\begin{aligned} &\frac{\partial D}{\partial y_w}(x_w, y_w) \\ &\quad = \mathbf{r}_3 . \frac{\partial \mathbf{p}_w^w}{\partial y_w}(x_w, y_w) \\ &\qquad - b'\left(G_y + \mathbf{r}_2 . \mathbf{p}_w^w(x_w, y_w)\right) \mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w}{\partial y_w}(x_w, y_w) \\ &\quad = 0. \end{aligned} \tag{41}$$

The last term of the (40) and (41) can be rewritten as follows:

$$\mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w}{\partial y_w}(x_w, y_w) = \left(r_{22} - r_{23} \frac{r(y_w) r'(y_w)}{\sqrt{r(y_w)^2 - x_w^2}}\right), \tag{42}$$

and its value is different from zero, since $r(y_w)^2 \gg x_w^2$ and $r_{22} \gg r_{23} r'(y_w)$. The term

$$b'(G_y + \mathbf{r}_2 . \mathbf{p}_w^w(x_w, y_w))$$

can be obtained from (41) and can be introduced in (40), in order to achieve:

$$\begin{aligned} &\left(\mathbf{r}_3 . \frac{\partial \mathbf{p}_w^w}{\partial x_w}(x_w, y_w)\right)\left(\mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w}{\partial y_w}(x_w, y_w)\right) \\ &\quad = \left(\mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w}{\partial x_w}(x_w, y_w)\right)\left(\mathbf{r}_3 . \frac{\partial \mathbf{p}_w^w}{\partial y_w}(x_w, y_w)\right) \end{aligned} \tag{43}$$

and, carrying out the calculations:

$$\begin{aligned} r_{21} r_{32} \sqrt{r(y_w)^2 - x_w^2} &= (r_{22} r_{33} - r_{23} r_{32}) x_w \\ &\quad + r_{21} r_{33} r(y_w) r'(y_w). \end{aligned} \tag{44}$$

Setting for brevity $A_1 = r_{21} r_{32}$, $B_1 = r(y_w)$, $C_1 = r_{22} r_{33} - r_{23} r_{32}$ and $D_1 = r_{21} r_{33} r(y_w) r'(y_w)$ (44) can be rewritten as:

$$A_1 \sqrt{B_1^2 - x_w^2} = C_1 x_w + D_!\tag{45}$$

and therefore, squaring both the members, the term x_w can be calculated:

$$\begin{aligned} &x_{w1,2}(y_w) \\ &\quad = \frac{-C_1 D_1 \pm \sqrt{C_1^2 D_1^2 - (C_1^2 + A_1^2)(D_1^2 - A_1^2 B_1^2)}}{C_1^2 + A_1^2}. \end{aligned} \tag{46}$$

The variable x_w is then expressed as a function of y_w and, as it can be seen, there are again two possible values of x_w for each value of y_w. Finally we can introduce the relation (46) in the second member of (40) obtaining:

$$
\begin{aligned}
F_{1,2}(y_w) &= \frac{\partial D(x_{w1,2}(y_w), y_w)}{\partial y_w} \\
&= \mathbf{r}_3 . \frac{\partial \mathbf{p}_w^w(x_{w1,2}(y_w), y_w)}{\partial y_w} \\
&\quad - b' \left(G_y + \mathbf{r}_2 . \mathbf{p}_w^w \left(x_{w1,2}(y_w), y_w \right) \right) \\
&\quad \times \mathbf{r}_2 . \frac{\partial \mathbf{p}_w^w \left(x_{w1,2}(y_w), y_w \right)}{\partial y_w} \\
&= r_{32} y_w - r_{33} \frac{r(y_w) r'(y_w)}{\sqrt{r(y_w)^2 - x_{r1,2}(y_w)^2}} \\
&\quad - b' \left(G_y + r_{21} x_{w1,2}(y_w) + r_{22} y_w \right. \\
&\quad \left. - r_{23} \sqrt{r(y_w)^2 - x_{r1,2}(y_w)^2} \right) \\
&\quad \times \left(r_{22} - r_{23} \frac{r(y_w) r'(y_w)}{\sqrt{r(y_w)^2 - x_{w1,2}(y_w)^2}} \right) \\
&= 0. \qquad (47)
\end{aligned}
$$

The problem is reduced to a simple scalar equation in the variable y_w (where $y_w \in [700790]$ mm for the left side and $y_w \in [-790\ -700]$ mm for the right side), that can be solved numerically [9]. Also in this case the index $(_{1,2})$ indicates that the equation has to be solved for both the roots of (46).

As usual for simplicity the solutions y_{wi}^C (with $i = 1, 2, \ldots, n$) of (47) are split in two groups: y_{w1j}^C (with $j = 1, 2, \ldots, n_1$) indicate the roots of $F_1(y_w) = 0$ (obtained using the first root of (46)) and y_{w2k}^C (with $k = 1, 2, \ldots, n_2$) those of $F_2(y_w) = 0$ where $n = n_1 + n_2$ (obtained using the second root of (46)). Through (46) the values of the variable x_w corresponding to y_{w1j}^C and y_{w2k}^C can be evaluated:

$$
\begin{aligned}
x_{w1j}^C &= x_{w1}(y_{r1j}^C), \quad j = 1, 2, \ldots, n_1, \\
x_{w2k}^C &= x_{w2}(y_{r2k}^C), \quad k = 1, 2, \ldots, n_2
\end{aligned} \qquad (48)
$$

and therefore, by means of the (9), (10) and (35), the positions of the contact points on the wheel and on the rail surfaces can be calculated:

$$
\begin{aligned}
\mathbf{p}_{w1j}^{r,C} &= \mathbf{p}_w^r(x_{w1j}^C, y_{w1j}^C), \\
\mathbf{p}_{r1j}^{r,C} &= \mathbf{p}_r^r(x_{w1j}^C, y_{w1j}^C), \quad j = 1, 2, \ldots, n_1, \\
\mathbf{p}_{w2k}^{r,C} &= \mathbf{p}_w^r(x_{w2k}^C, y_{w2k}^C), \\
\mathbf{p}_{r2k}^{r,C} &= \mathbf{p}_r^r(x_{w2k}^C, y_{w2k}^C), \quad k = 1, 2, \ldots, n_2.
\end{aligned} \qquad (49)
$$

As in the case of the DIST method not all the solutions (x_{w1j}^C, y_{w1j}^C) (with $j = 1, 2, \ldots, n_1$) and (x_{w2k}^C, y_{w2k}^C) (with $k = 1, 2, \ldots, n_2$) of (47) can be accepted. Equation (44) contains irrational terms and consequently the following analytic conditions have to be verified for every j and k:

1. x_{w1j}^C and x_{w2k}^C calculated by means of (46) have to be real;
2. the terms

$$
\sqrt{r(y_{w1j}^C)^2 - {x_{w1j}^C}^2} \quad \text{and} \quad \sqrt{r(y_{w2k}^C)^2 - {x_{w2k}^C}^2}
$$

appearing in (47) have to be real;
3. (x_{w1j}^C, y_{w1j}^C) and (x_{w2k}^C, y_{w2k}^C) have to be effective solutions of (44).

Moreover concerning eventual multiple solutions the same considerations explained for the DIST method are necessary.

3.2.3 *Indentation between the contact surfaces*

The generic solution of (36) can be considered an effective contact point only if the normal indentation between the surfaces p_n is negative. In this case the normal indentation p_n can not be calculated directly from the values of x_w, y_w, because the normal to the contact surfaces in the contact point is not unique with this approach. For this reason it was approximated with the value $\tilde{p}_{ni}$ calculated as follows (Fig. 13):

$$
\tilde{p}_{ni} = \mathbf{D}_i^{r,C} \cdot \mathbf{n}_r^r(\mathbf{p}_{ri}^{r,C}) = p_{vi} \cos \vartheta_i . \qquad (50)
$$

In this expression, $\mathbf{D}_i^{r,C}$ is the vector representing the difference between the surfaces, defined as:

$$
\mathbf{D}_i^{r,C} = \mathbf{p}_{wi}^{r,C} - \mathbf{p}_{wi}^{r,C}, \qquad (51)
$$

p_{vi} represents the indentation evaluated in the z_r direction, defined as:

$$
p_{vi} = \mathbf{D}_i^{r,C} \cdot \mathbf{k}_r, \qquad (52)
$$

and ϑ_i is the angle between the z_r direction and the unitary vector normal to the rail surface:

$$
\cos \vartheta_i = \mathbf{n}_r^r(\mathbf{p}_{ri}^{r,C}) \cdot \mathbf{k}_r . \qquad (53)
$$

Therefore, for each solution of (36), the following condition has to be verified:

$$
\tilde{p}_{ni} \leq 0. \qquad (54)
$$

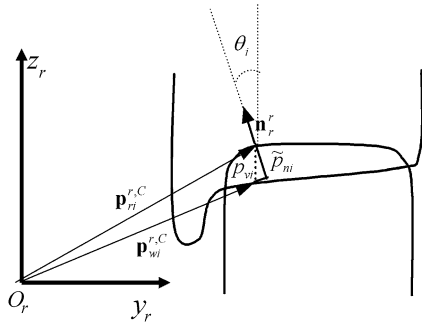


Fig. 13 Minimization of the difference: definition of the surface indentation

The DIFF method requires therefore the following steps:

1. determination of the solutions (x_{wi}^C, y_{wi}^C) (with $i = 1, 2, \ldots, n$) of (36);
2. research and elimination of the multiple solutions;
3. check of the analytic conditions;
4. check of the minimum condition;
5. check of the condition on the normal indentation.

4 Neural networks

As mentioned in the introduction, neural networks belong to the category of *black box models* or *data driven models*. These types of models are often used when the physics of the system or the phenomenon that has to be modeled is unknown, uncertain or difficult to be represented with a classical deterministic approach, but an high number of input/output pairs are available (for example from experimental data).

The neural network model has a predefined mathematical structure, defined by the user, whose outputs depend on a series of parameters; an iterative process known as *training* define the values of these parameters in order to obtain a model able to *copy* the available input/output pairs.

The neural network usually adopted for the approximation of functions belong to the feed-forward type [16]: the connection between neurons are always from the input to the output and there is not a feedback of the output. Each inner layer may contain an unconstrained number of neurons, while the output layer contains a number of neurons equal to the number of outputs of the function that is being approximated. All the neurons of a layer have the same activation function, which has to be chosen during the definition of the network architecture (Fig. 14). Indicating with **p** and **a** the input and output vectors respectively, and with **x** the vector containing weights and bias for all neurons in the network, the following relation can be written:

$$\mathbf{a} = \tilde{\mathbf{F}}(\mathbf{x}, \mathbf{p}).$$

Once the structure of the network (number of hidden layers, number of neurons, activation functions) has been defined, the net parameters (weights and biases) has to be chosen by means of the *training*, an iterative optimization procedure. In the presented application the objective of the neural network is to obtain with a lower computation burden the same precision of the semi analytic methods previously summarized. The values of the network parameters **x** have then to be chosen in order to obtain the best fit between the contact point locations calculated by the network and those obtained with the previously described methods.

In the preceding sections different deterministic methods to solve the wheel/rail contact problem have been described; their main drawbacks are related to the computational costs. The implementation of a model based on neural network was developed in order to find a faster algorithm, which could be implemented online, assuring low computational time without the need to store in memory large look-up tables. The semianalytical procedures, because of their superior performances with respect to the numerical procedures, have been 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 collection of reference data: each datum is a vector which contains an allowed input vector and the corresponding desired output vector;
- the choice of network's architecture (organization of neurons in the network and definition of the activation function for each neuron);
- training: 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; in this application the *mean square error* (*mse*) has been used.

In the presented application, the network architecture is a multilayer perceptron, the training process

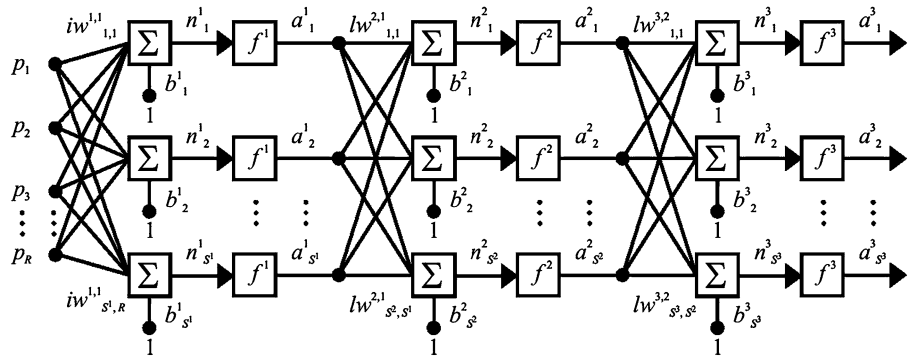


Fig. 14 Scheme of a multilayer perceptron

is the Levenberg-Marquardt algorithm. The reference data are divided in three sets: the *training set*, which is used in the training procedure to update the network parameters, the *validation set*, which is necessary for early stopping procedures, and the *test set*, which is used at the end of the training to evaluate the performance. The validation set is not used to improve the parameter values, but the error on this set is evaluated at each epoch: if the mean square error on this set raises for a certain number of consecutive epochs, the training process is stopped. This feature (*early stopping*) is necessary to avoid a phenomenon called network *overfitting* in which the updating of weights and biases do not correspond to a better approximation of the objective function but only leads to fit the noise on the reference data in training set.

In the previous sections two different semi analytic procedures has been described: 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 $\mathbf{o}_r$ 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. Furthermore, while in the DIST method 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. Then if the DIFF method is used as reference for the training of the neural network, the input of the function that has to be identified by the network is the vector ϕ defined as:

$$\phi = \begin{bmatrix} \alpha \\ \beta \\ G_y \end{bmatrix}. \tag{55}$$

A neural network based on the DIFF method will thus have three inputs (while if it was based on the DIST method it would have four inputs).

4.1 Implementation of a neural network based procedure for the definition of wheel/rail contact points

4.1.1 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: $x^M_{bk}, y^M_{bk}, x^M_{rk}, y^M_{rk}$).

Two different ways have been considered to solve this problem: the first method is to create a network with $4 \times L$ outputs, where 4 is the number of coordinates used to determine the position of a local minimum, and L the maximum number of local minima

Table 1 Variability range of the configuration variables

Variable	Minimum	Maximum	Step
α (rad)	$-\pi/180$	$\pi/180$	$\pi/14400$
β (rad)	$-\pi/240$	$\pi/240$	$\pi/19200$
g_y (mm)	-10	10	0.125

which is possible in all the allowed configurations. For every configuration in which the number of local minima is lower than L, the network is trained in order to give in output the coordinates of effective local minima plus a series of recognizable numeric values.

In the second method a foregoing classification is performed, which evaluates the number of local minima for the examined configuration; then L neural networks are created. 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, a neural network with a proper number of output is applied.

The first method is not adequately efficient, because if a fictitious value is assigned to coordinates of non-existent local minima, the function that has to be fitted with the neural network is strongly discontinuous. The second method has then be chosen for the implementation in the research of the wheel/rail contact points. The classification could also be performed with a neural network which outputs the number of local minima for each configuration, for this problem other types of neural networks could be used, for example the Self Organizing Maps or Learning Quantization Networks [17, 18].

In the presented application the ranges in the space of configurations with $1, 2, \ldots, L$ local minima were directly detected. It can be observed that the dependence on the angle α is weak, 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. These observations are made analyzing the results of a set composed of 4173281 configurations, obtained varying the coordinates α, β and g_y in ranges described in Table 1.

Figures 15 and 16 show the results obtained using the wheel profiles ORE S1002 and rail profile UIC60 with laying angle 1/40; the first one is obtained with $\alpha = 0$, while the second is obtained with $\alpha = \pi/180$. 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 gray area those with 3 minima. 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.

Figures 17 and 18 show the results when the laying angle is 1/20. Considerations made for $\alpha_p = 1/40$ are still valid in this case, even if configurations with up to 4 local minima are present. There is also a deformation of the regions with different number of contact points when α vary from 0 to $\pi/180$, but even in this case it is limited to a quite little region.

In order to evaluate the accuracy of the classification, the following procedure has been adopted to calculate the error on classification E_c:

$$E_c = \frac{\sum_{i=1}^{C} |{N_i}^T - {N_i}^e|}{\sum_{i=1}^{C} {N_i}^e} \tag{56}$$

where C refers to the total number of analyzed configurations, ${N_i}^T$ refers to the number of local minima that the approximated classification recognize for the i-th configuration, while ${N_i}^e$ refers to the number of local minima calculated by the DIFF method. For all the analyzed configurations, the error $E_c \cong 0.91\%$ for $\alpha_p = 1/40$, and $E_c \cong 0.85\%$ 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, for each case a proper neural network for the localization of the contact points has to be defined. Table 2, summarizes the percentage of configurations with 1, 2 or 3 contact points. The presented results refer to the configurations defined in Table 1.

As it can be seen, 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.

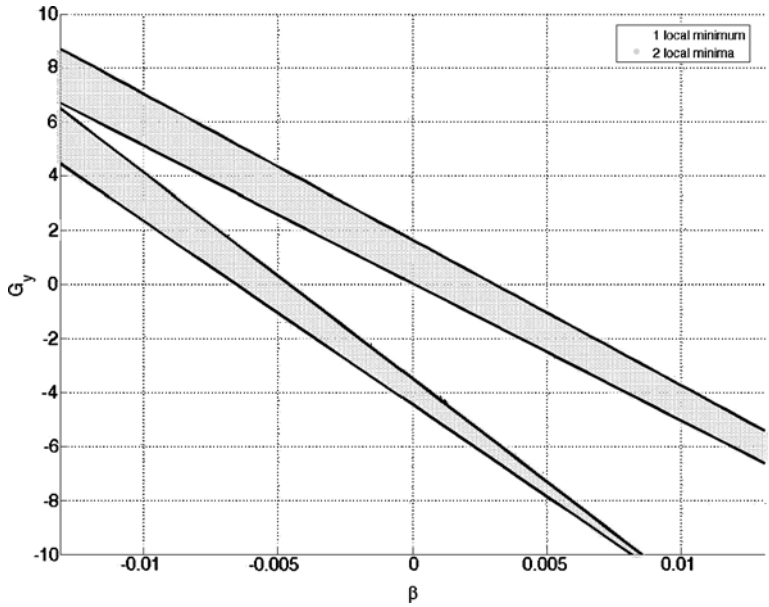


Fig. 15 Number of local minima ($\alpha_p = 1/40$, $\alpha = 0$)

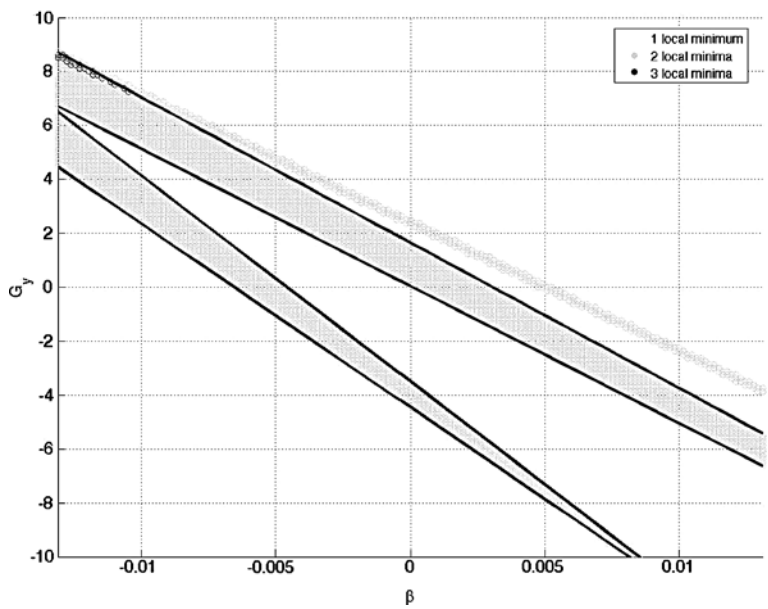


Fig. 16 Number of local minima ($\alpha_p = 1/40$, $\alpha = \pi/180$)

4.1.2 Definition of neural networks and training

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 number of neurons in output layer depends on the number of outputs, while in the hidden layer it can be changed in order to optimize the performances (in terms of speed and precision) of the network. Increasing the number of neurons leads to an higher calculation time both in the training process

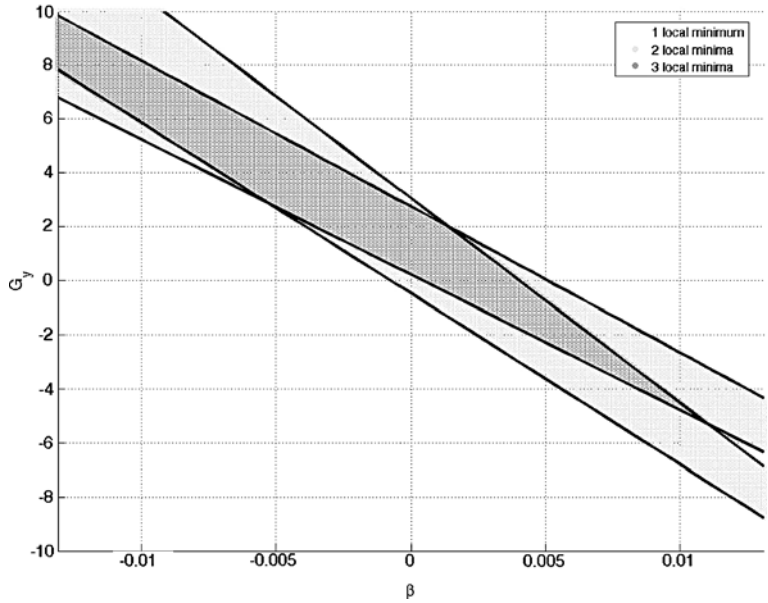


Fig. 17 Number of local minima ($\alpha_p = 1/20$, $\alpha = 0$)

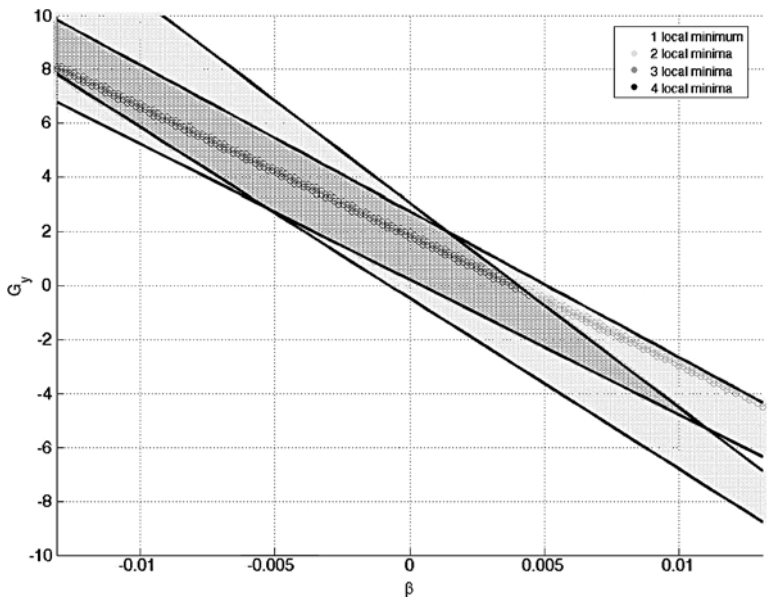


Fig. 18 Number of local minima ($\alpha_p = 1/20$, $\alpha = \pi/180$)

and in the network final implementation. In addition it can be observed that the precision of the model do not necessarily increases with the number of neurons.

The performances of a network are evaluated analyzing the errors on the test set. For the k-th network which outputs are the coordinates of k contact points, for each configuration, the vectors **MinDIFF** (coordinates of local minima calculated by the DIFF method) and **MinNet** (coordinates of local minima calculated by the neural networks) are defined:

$$\mathbf{MinDIFF} = \left[x_{b1}^M, y_{b1}^M, x_{b2}^M, y_{b2}^M, \ldots, x_{bk}^M, y_{bk}^M, \right]^T, \tag{57}$$

Table 2 Number of contact points per configuration, $\alpha_p = 1/40$ and $\alpha_p = 1/20$

Number of contact points	Number of configurations	Percentage
$\alpha_p = 1/40$		
1	3628587	86.96%
2	544292	13.04%
3	402	0.009633%
$\alpha_p = 1/20$		
1	3349863	80.27%
2	441866	10.59%
3	368952	8.4%
4	12600	0.302%

$$\mathbf{MinNet} = \left[x_{b1}^{NN}\, y_{b1}^{NN}\, x_{b2}^{NN}\, y_{b2}^{NN} \vdots x_{bk}^{NN}\, y_{bk}^{NN} \right]^T.$$

The algorithm which define the number of errors e_j on the j-th configuration can be summarized in the following steps:

- $e_j = 0$
- for $i = 1, 2, \ldots, k$
  - if $\sqrt{(x_{bi}^M - x_{bi}^{NN})^2 + (y_{bi}^M - y_{bi}^{NN})^2} \geq toll$
  - $e_j = e_j + 1$
  - end
- end

In other terms, the algorithm increases the error counter when the distance between the contact point calculated by the network and the reference one is greater than a fixed value *toll* (whose value was set equal to 1 mm). are given in Table 5. The percentage error for the network on a test set which contains Q_k^t configurations is then given by:

$$E_k = \frac{\sum_{j=1}^{Q_k^t} e_j}{k \cdot Q_k^t}. \tag{58}$$

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

$$E_{NN} = \frac{\sum_{k=1}^{L} E_k \cdot k \cdot A_k}{\sum_{k=1}^{L} k \cdot A_k} \tag{59}$$

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 k-th network and A_k the available data for the k-th net.

Table 3 Configuration variables, variability range

Variable	Min value	Max value	Step
α (rad)	0	$\pi/180$	$\pi/3600$
β (rad)	$-\pi/240$	$\pi/240$	$\pi/4800$
g_y (mm)	-10	10	0.5

The definition of the error for a neural network allows to compare the performances of networks with different number of neurons and the same k and α_p values. The analyzed results show that, for $k = 1, 2$ and $\alpha_p = 1/40$ and for $k = 1, 2, 3$ and $\alpha_p = 1/20$, the number of errors does not monotonically decrease with the number of neurons but a minimum occurs generally between 20 and 30 neurons in the hidden layer. For example, concerning the neural network which outputs are the coordinates of a single contact point and for $\alpha_p = 1/40$, a minimum in the error can be observed for 25 neurons (Fig. 19).

In order to simplify the training, the symmetry conditions described in the preceding sections have been considered. The networks were trained using 2099601 different wheel-rail relative positions, obtained varying α, β e g_y in the ranges summarized in Table 3, 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^M can be obtained from the y_b^M according to the relation $z_b = b\,(y_b)$, the network outputs are the coordinates x_b^M, y_b^M for each contact point.

For the training the *Levenberg-Marquardt* algorithm has been chosen, 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). For the parameters introduced by the *Levenberg-Marquardt* algorithm, an initial value $\mu = 0.001$, with $\mu_{dec} = 0.1$ has been chosen [16].

4.1.3 Training for laying angle $\alpha_p = 1/40$

As previously discussed, in this case 2 networks are necessary; the first gives 2 outputs (coordinates of a single point), while the second gives 4 outputs (coordinates of 2 points). In order to avoid problems related

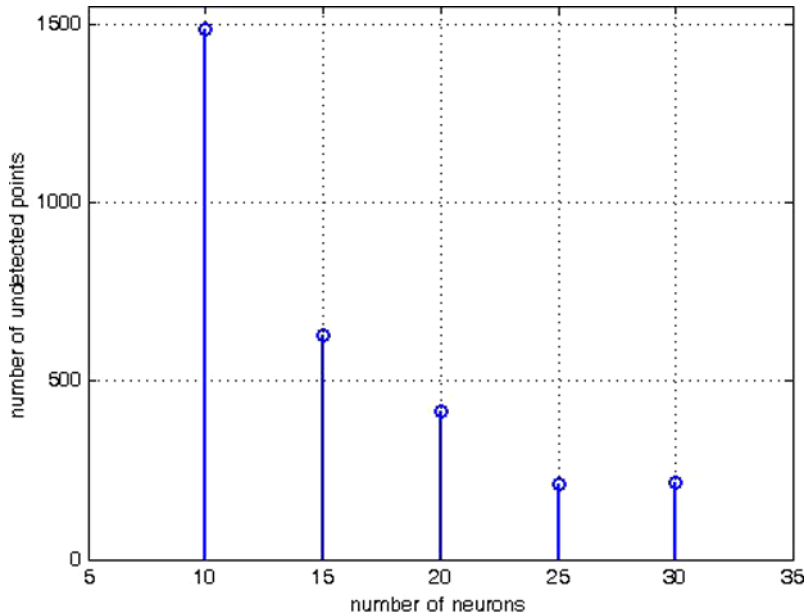


Fig. 19 Errors in test set for different number of neurons in the hidden layer, $\alpha_p = 1/40$, single contact point

Table 4 Data sets for training networks ($\alpha_p = 1/40$)

Number of k contact points	Training set	Validation set	Test set	Available data A_k
1	18258	9128	9128	1825603
2	18254	9127	9127	273797

Table 5 Error on *test set* for $\alpha_p = 1/40$

Number of contact points k	Percentage error on test set E_k
1	2.29%
2	1.45%

to the computational burden of the training process, only a subset of the reference configurations have been used, Table 4 contains the number of samples used in training process, compared with the total available data. The percentage errors for the defined networks are summarized in Table 5.

From the number of contact points for each configuration (Table 2) and the errors of NN (Table 5) the total error of the proposed algorithm can be evaluated; since for $\alpha_p = 1/40$ a network which outputs are the coordinates of 3 local contact points has not been defined, the percentage error in this case is $E_3 = 100\%$. For $\alpha_p = 1/40$, the mean global percentage error is $\mathbf{E}_{NN} = \mathbf{2.12}\%$.

4.1.4 Training for laying angle $\alpha_p = 1/20$

For $\alpha_p = 1/20$ 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). Also in this case only a subset of the reference configurations have been used, in order to reduce the computational load. Table 6 contains the number of data used in training process, compared with the total available data. The performances of each trained network are evaluated by the errors on Test set. The previously described procedures and considerations for the evaluation of the errors have been applied also in this case. The percentage errors for the networks are summarized in Table 7.

From the number of contact points for each configuration (Table 2) and the errors of NN (Table 7) mean total error of the procedure can be evaluated as shown in (59). For $\alpha_p = 1/20$ a network which outputs the coordinates of 4 contact points has not been defined, so its percentage error is $E_4 = 100\%$ The total mean percentage error for $\alpha_p = 1/20$ is $\mathbf{E}_{NN} = \mathbf{2.41}\%$.

Table 6 Data sets for training networks ($\alpha_p = 1/20$)

Number of k contact points	Training set	Validation set	Test set	Available data A_k
1	16854	8427	8427	1685338
2	14822	7411	7411	222315
3	12378	6189	6189	185648

Table 7 Error on *test set* for $\alpha_p = 1/20$

Number of contact points k	Percentage error on test set E_k
1	0.262%
2	2.91%
3	4.17%

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.

5.1 Dynamic simulations

In this section, the results obtained with a multibody simulator developed in the Simulink® environment in which the wheel/rail contact points are calculated by the method previously described are compared with those obtained with a model developed in SIMPACK. The railway vehicle chosen for the dynamic simulations is the Manchester Wagon, whose physical and geometric characteristics are available in [19].

The multibody 3D model of the Manchester Wagon, is schematically shown in Fig. 20. The vehicle is composed of the following parts:

- the car body (Fig. 20a)
- two bogies (Fig. 20b)
- four axles
- primary and secondary suspensions modelled by three-dimensional non linear force elements like bushings, dampers and bumpstops.

The simulation was performed on a S-shaped curve of radius $R = 190$ m with four-meter-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. In the following table are reported the characteristics of the track. Figures 21–23 show some of the results. In each figure, the *continuous grey* line refers to results obtained with SIMPACK, while the *dashed black* one refers to Simulink® results.

Table 8 Track characteristics

Velocity	40 km/h
Wheel profile	`ORE S1002`
Rail profile	`UIC60`
Rail angle	1/40
Wheel–rail friction coefficient	0.4
Initial straight track length	50 m
Radius of curve[a]	190 m
Length of curve[b]	30 m
Intermediate straight track length	4 m
Final straight track length	100 m

[a]Being an S-shaped curve, the track is composed of two bends with same radius (but opposite curvature) with interposed a short straight. First curve right

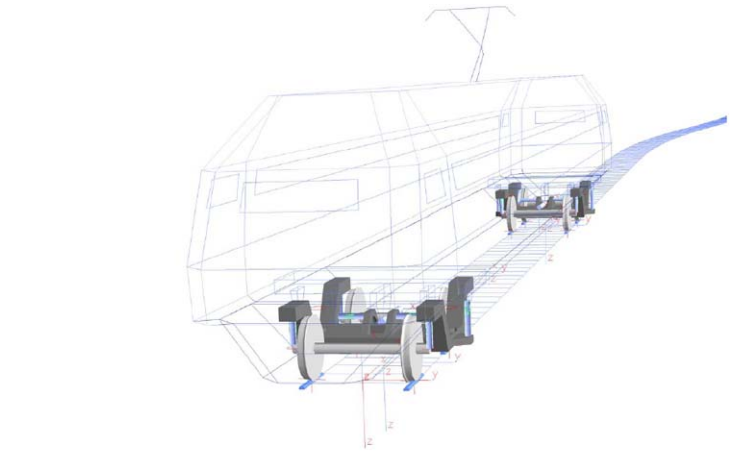
[b]No transition: the vehicle passes abruptly from straight to curve and to straight again (2 times)

As it can be observed there is a good agreement between the two models for the kinematic measurement (Fig. 21), some differences arise in the forces (Figs. 22–23). These differences can be explained by the different procedures used to find the contact points and to calculate the contact forces. The Simulink® procedure calculates the normal component of the contact force by means of a non linear spring-damper model based on Hertz theory, the details of this model can be found in [13].

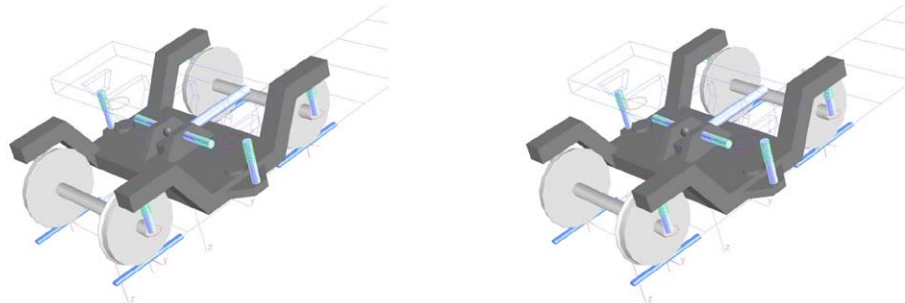
Figures 24 show the position of the contact point on the wheels and the rails of the first wheelset during the simulation. The wheels and the rails were drawn from their respective profiles as cylindrical surfaces long as the entire track.

5.2 Performances of the proposed methods

In this section the performances of the proposed procedures for the detection of the contact points will be compared with those of other methods previously developed ([11, 12]). All the 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



(a) Complete vehicle.



(b) Bogies.

Fig. 20 Multibody model of the Manchester vehicle in SIMPACK

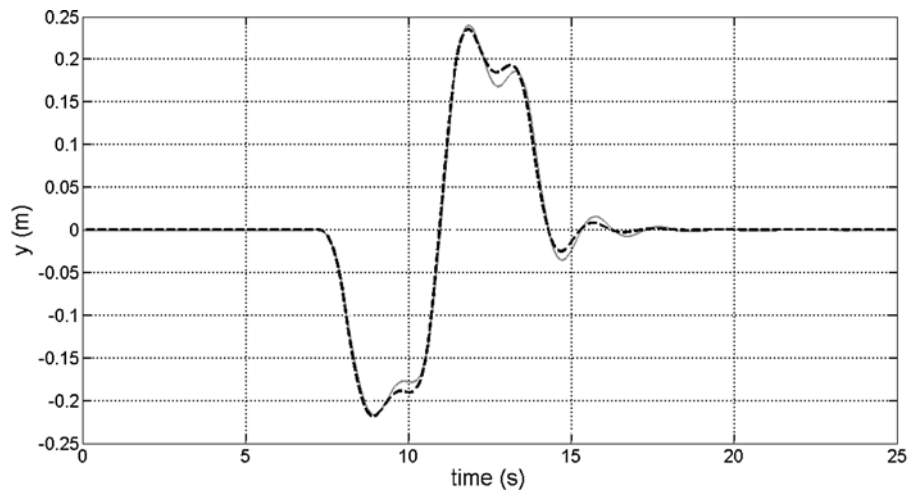


Fig. 21 Lateral displacement (y) of the centre of mass of car body; *continuous grey line*: results obtained with SIMPACK, *dashed black curve*: Simulink® results

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. These parameters have been varied within

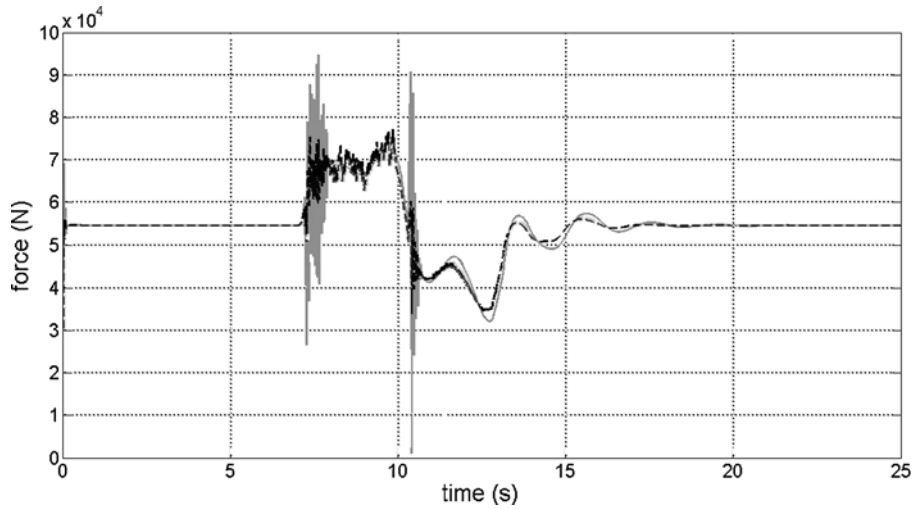


Fig. 22 Vertical force (Q) on the first wheelset; left wheel *continuous grey line*: results obtained with SIMPACK, *dashed black curve*: Simulink® results

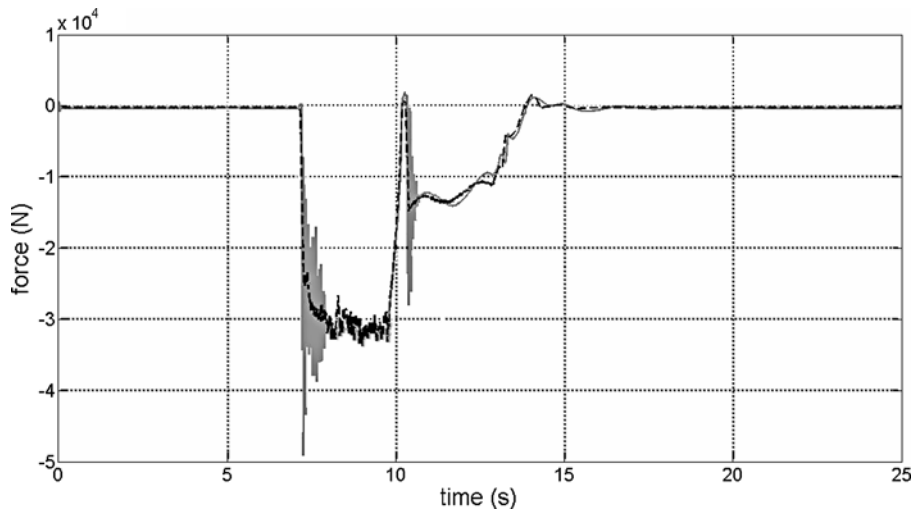


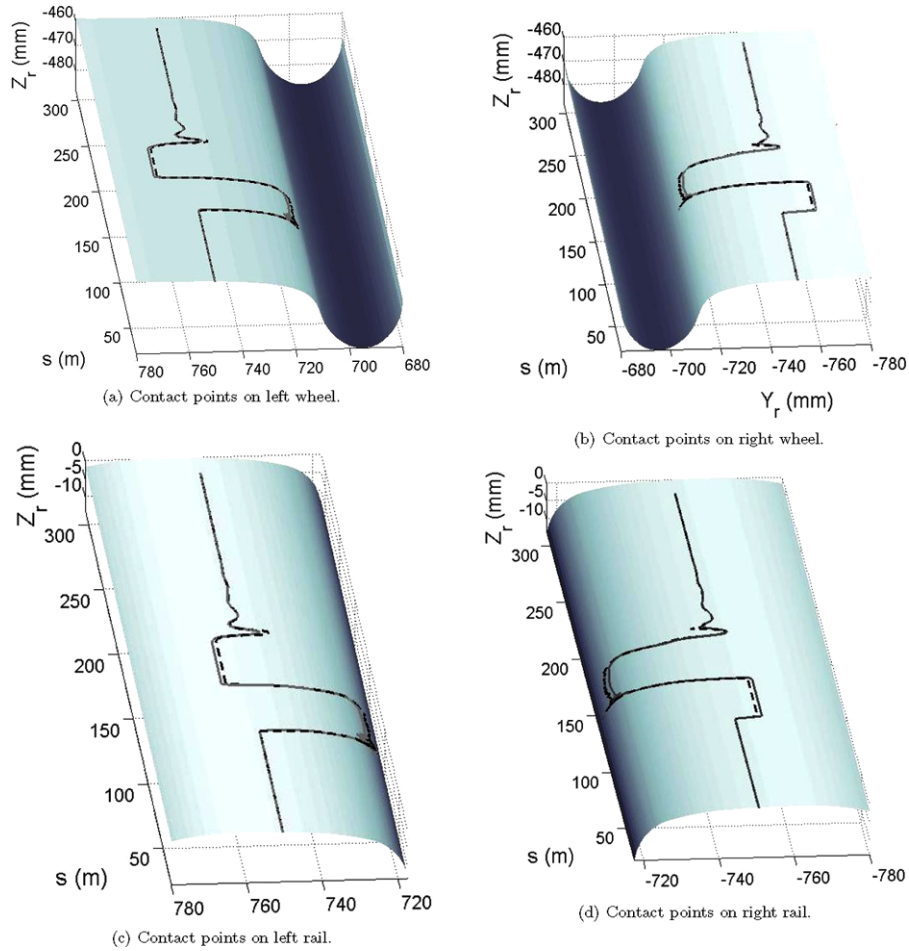
Fig. 23 Lateral force (Y) on the first wheelset; left wheel *continuous grey line*: results obtained with SIMPACK, *dashed black curve*: Simulink® results

Table 9 Range of variability of the parameters

G_y	0 mm ÷ 10 mm
α	0 rad ÷ 0.01 rad
β	−0.01 rad ÷ 0.01 rad

the ranges summarized in Table 9, obtaining a large number of configurations ($\approx 4.2 \cdot 10^6$). The ranges have been chosen taking into account the symmetry on α.

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 obtain, 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.



(a) Contact points on left wheel.

(b) Contact points on right wheel.

(c) Contact points on left rail.

(d) Contact points on right rail.

Fig. 24 Contact points on the wheels and rails; left wheel *continuous grey line*: results obtained with SIMPACK, *dashed black curve*: Simulink® results

A procedure to evaluate the performance of the different algorithms has to be defined. To this purpose a single wheel and the corresponding rail are considered (indifferently on the left or on the right side). $\mathbf{p}_{bi}^{CA}$ with $i = 1, 2, \ldots, N^A$ and $\mathbf{p}_{bj}^{CB}$ with $j = 1, 2, \ldots, N^B$ represent the contact points (on the rail surface) detected by two generic procedures A (chosen as benchmark) and B in the same relative wheel-rail configuration. The number E^{AB} of the errors can be determined through the following algorithm:

1. $\mathbf{v}_A = \mathbf{0}$ and $\mathbf{v}_B = \mathbf{0}$ where $\mathbf{v}_A \in \mathbb{R}^{N^A}$ and $\mathbf{v}_B \in \mathbb{R}^{N^B}$.
2. for $i = 1, 2, \ldots, N^A$
   for $j = 1, 2, \ldots, N^B$
   if $\|\mathbf{p}_{bi}^{CA} - \mathbf{p}_{bj}^{CB}\|_2 \leq toll$ and $v_B(j) = 0$
   set $v_A(i) = 1$ and $v_B(j) = 1$
   break
   end
   end
   end
3. set $n^A = N^A - \|\mathbf{v}_A\|_2^2$ and $n^B = N^B - \|\mathbf{v}_B\|_2^2$
4. $E^{AB} = \max(n^A, n^B)$.

In other words E^{AB} can be represented as the sum of the undetected pairs of points $E_1^{AB} = \min(n^A, n^B)$ and of the points in excess $E_2^{AB} = \max(n^A, n^B) - \min(n^A, n^B)$. More broadly E_{hk}^{AB} is the number of er-

Table 10 Global error, comparison between the different solutions

Error	Reference	Tested	$\alpha_p = 1/40$	$\alpha_p = 1/20$	Tolerance
e^{dD} (%)	DIFF	DIST	1.0	0.9	2 mm
e^{DG} (%)	DIFF	GRID	1.1	4.2	2 mm
e^{DCS} (%)	DIFF	CS	5.0	10	2 mm
e^{DS} (%)	DIFF	S	3.4	10	2 mm
e^{DNN} (%)	DIFF	NN	1.04	1.98	2 mm

rors of the procedure A in the k-th configurations with h contact points. The partial errors e_h^{AB} can then be defined as follows:

$$e_h^{AB} = \frac{\sum_{k=1}^{T_h^A} E_{hk}^{AB}}{hT_h^A}, \quad h = 1, 2, \ldots, P^A \tag{60}$$

where T_h^A is the number of relative wheel-rail configurations with h contact points and P^A is the maximum number of contact points present on a single wheel. The global error e^{AB} can be expressed as:

$$e^{AB} = \frac{\sum_{h=1}^{P^A} \sum_{k=1}^{T_h^A} E_{hk}^{AB}}{T^A} \tag{61}$$

where $T^A = \sum_{h=1}^{P^A} hT_h^A$ is the total number of the contact points.

Table 10 summarizes the error values obtained comparing the different procedures assuming the DIFF method as reference. It can be observed that:

- the DIST and the DIFF methods give substantially the same results (the difference is lower than 1% and this value is not influenced by the laying angle);
- the differences with the GRID method and with the direct search iterative algorithms (Compass Search and Simplex method) are larger and depends on the laying angle;
- the difference between the DIFF and the Neural Network methods is quite small (lower than 2%) and does not depend significantly on the laying angle.

The performances of the various procedures have been compared in terms of computation times. Table 11 summarize the mean time required to evaluate the contact points in a generic relative wheel-rail configurations. All the results have been obtained with a processor Intel Pentium 4 (3.0 GHz).

The described results allow to conclude that:

Table 11 Computation times: comparison between the analyzed methods

Method	Time (s)
GRID	9.3
S	0.255
CS	0.105
DIST	0.045
DIFF	0.037
NN	0.0006

1. the performances of the DIST and the DIFF methods are similar in terms of precision and computation times;
2. the semianalytic procedures are reliable, and more accurate than the procedures based on the numerical iterative algorithms, which furthermore require an higher computational time;
3. 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 comparable with those necessary to read look-up tables, so it could be implemented online obtaining acceptable calculation times in dynamic analysis.

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 (only multiplications and simple functions have to be applied). 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 *online* 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.

The substantial difference between the *classical* approaches (numerical and semi-analytic methods) and neural networks is that the former solve the problem in its exact form, while the latter use an approximation

of the problem. On the other hand, the first methods are not suitable for a *on-line* implementation and then require the definition of lookup tables; the location of the contact points during the simulation is obtained interpolating the values stored in the table, while the method based on the neural networks gives a continuous output.

6 Conclusions

In this work two approaches for the detection of the wheel-rail contact points are presented. The first is the semianalytic approach, which 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 that is then solved numerically. Since the problem 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 neural networks. The aim of this approach is to develop a model reliable as the semianalytic 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 semianalytic procedures based on the minimization of the surface defined as the difference between the wheel surface and the rail surface.

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 semianalytic procedures (named DIST and DIFF methods) have similar performances in terms of precision and computation times; both of them are reliable as regards the precision and are more accurate and faster than the procedures based on the numerical iterative algorithms, so 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 an approximation of the solution (but the error does not noticeably affect the multibody simulation), but requires a calculation time which 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.

References

1. Shabana AA, Sany JR (2001) An augmented formulation for mechanical systems with non-generalized coordinates: application to rigid body contact problems. Nonlinear Dyn 24:183–204
2. Shabana AA, Sany JR (2001) A survey of rail vehicle track simulations and flexible multibody dynamics. Nonlinear Dyn 26:179–210
3. Rulka W, Pankiewicz E (2005) MBS approach to generate equations of motion for HiL–simulations in vehicle dynamics. Multibody Syst Dyn 14:367–386
4. Shabana AA, Zaazaa KE, Escalona JL, Sany JR (2004) Development of elastic force model for wheel/rail contact problems. J Sound Vib 269:295–325
5. Shabana AA, Tobaa M, SugiYama H, Zaazaa KE (2005) On the computer formulations of the wheel/rail contact problem. Nonlinear Dyn 40:169–193
6. Pombo J, Ambrosio J (2005) Dynamic analysis of a railway vehicle in real operation conditions using a new wheel-rail contact detection model. Int J Vehicle Syst Model Test 1(1–3):79–105
7. Kik W, Moelle D (2000) Implementation of the wheel-rail element in ADAMS/rail version 10.1. In: 5th ADAMS/rail users' conference, Haarleem
8. Netter H, Schupp G, Rulka W, Schroeder K (1998) New aspects of contact modelling and validation within multibody system simulation of railway vehicles. Vehicle Syst Dyn 29(S1):246–269
9. Kolda TG, Lewis RM, Torczon V (2003) Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev 45(3):385–482

10. Nelder JA, Mead R (1965) A simplex method for function minimization. Comput J 7:308–313
11. Malvezzi M, Meli E, Papini S, Pugi L (2007) Parametric models of railway systems for real-time applications. In: Multibody dynamics 2007, Milano, Italy
12. Auciello J, Malvezzi M, Meli E, Papini S, Pugi L, Rindi A (2007) Multibody models of railway vehicles for real-time systems. In: XVIII congresso AIMETA, Brescia, Italy
13. Malvezzi M, Meli E, Falomi S, Rindi A (2008) Determination of wheel-rail contact points with semianalytic methods. Multibody Syst Dyn 20:327–358
14. Esveld C (2001) Modern railway track, 2nd edn. Delft University of Technology, Delft
15. do Carmo MP (1976) Differential geometry of curves and surface. Prentice-Hall, Englewood Cliffs
16. Demuth H, Beale M, Hagan M (2007) Neural network toolbox user's guide. The MathWorks
17. Kohoen T (1995) The self organizing map. Springer, Berlin
18. Devijer PA, Kittler J (1982) Pattern recognition, a statistical approach. Prentice-Hall, London
19. Iwnicki S (1999) The Manchester benchmarks for rail vehicle simulators. Swets & Zeitlinger, Lisse, ISBN 90 265 1551 0