



Comparison of numerical and computational aspects between two constraint-based contact methods in the description of wheel/rail contacts

Xinxin Yu¹ · Javier F. Aceituno² · Emil Kurvinen³ · Marko K. Matikainen¹ · Pasi Korkealaakso⁴ · Asko Rouvinen⁴ · Dezhi Jiang¹ · José L. Escalona⁵ · Aki Mikkola¹

Received: 11 August 2021 / Accepted: 3 January 2022 / Published online: 24 January 2022
© The Author(s) 2022

Abstract

The numerical and computation aspects of the Knife-edge Equivalent Contact (*KEC*) constraint and lookup table (*LUT*) methods are compared in this paper. The *LUT* method implementation uses a penetration-based elastic contact model for the flange and a constraint-based formulation at the wheel tread. For the *KEC* method, where an infinitely narrow rail contacts an equivalent wheel, regularization of the tread-flange transition is adopted to simultaneously account for tread and flange contacts using constraints. A comparison between the two methods is carried out using well-known numerical integrators to show the applicability and limitations of both methods.

Two fixed-step-size integrators, the explicit Runge–Kutta (RK4) and the predictor–corrector Adam–Bashforth–Moulton (ABM) methods, and two variable-step-size Matlab built-in function integrators, the explicit *ode45* and implicit *ode15s*, were applied to get the numerical solutions to the dynamic problems and study the relative numerical performance of the two contact description methods. To complete the railway vehicle model, both contact methods were implemented for the multibody model of a benchmark railway vehicle (the Manchester wagon 1). Numerical results were obtained for different railway tracks with and without irregularities. Profiles of the S1002 wheel and LB-140-Area rail, which demonstrate the two-point contact phenomenon, were considered. Both methods were implemented in Matlab and validated against commercial simulation software. The kinematic results for both approaches show good agreement, but the *KEC* method was up to 20% more efficient than the *LUT* method regardless of integrator used.

Keywords Time integration · Lookup table · *KEC* method · Manchester wagon · Wheel–rail contact

Nomenclature

- \mathbf{A}^t Rotation matrix from a track frame with respect to the global frame
- \mathbf{A}^{bti} Rotation matrix of the body track frame with respect to the global frame
- $\mathbf{A}^{bti,i}$ Rotation matrix of body frame with respect to the body track frame
- $\mathbf{A}^{wti,wi}$ Rotation matrix of wheelset frame with respect to the wheelset track frame

Extended author information available on the last page of the article

- $\mathbf{A}^{w_i, w_{li}}$ Rotation matrix of wheelset intermediate frame with respect to the wheelset track frame
- \mathbf{A}^{w_{li}, w_i} Rotation matrix of wheelset frame with respect to the wheelset intermediate frame
- $\mathbf{A}^{t, rp}$ Rotation matrices of the rail profile frames with respect to the track frame
- \mathbf{C}^q Jacobian matrix of *LUT* constraints
- \mathbf{C}^{LUT} *LUT* constraint equations
- \mathbf{C}^{KEC} *KEC* constraint equations
- D^{damp} Nonlinear damping constant for flange contact in *LUT* method [$\text{kN} \cdot \text{s}/\text{m}^2$]
- D Damping coefficient [$\text{kN} \cdot \text{s}/\text{m}$]
- d Deformation length of the spring [m]
- \dot{d} Time derivative of spring deformation [m/s]
- f^{susp} Magnitude of the suspension force
- f^{lk} Value of the wheel equivalent profile
- f_{klt}^k *KEC*-lookup table of wheel equivalent profiles
- h^t Functions that define railhead profiles
- h^w Functions that define wheel profiles
- K Spring stiffness [kN/m]
- K^{hertz} Hertzian stiffness for flange contact in *LUT* method [$\text{N}/\text{m}^{1.5}$]
- K^{series} Damper series stiffness [kN/m]
- K^{bend} Bending stiffness [kN/rad]
- K^{shear} Shear stiffness [kN/m]
- l_0 Undeformed spring length [m]
- L_w Wheel profile positioning with respect to the track centerline [m]
- L_r Rail profile positioning with respect to the track centerline [m]
- \mathbf{M} Vehicle generalized mass matrix
- \mathbf{N} Matrix that represents the direction of the reaction forces
- $\bar{\mathbf{n}}_c^{w_i}$ Unit normal vector with respect to the wheel surface at the contact point
- $\hat{\mathbf{n}}_c^{rp}$ Unit normal vector with respect to the railhead surface at the contact point defined in rail profile frame
- $\bar{\mathbf{n}}_c^{rp}$ Unit normal vector of the contact point to the railhead surface defined in wheelset track frame
- $\bar{\mathbf{n}}_{c,fla}^{rp}$ Unit normal vector at flange contact point to the rail surface defined in wheelset track frame
- \mathbf{q} Vector of multibody system coordinates
- \mathbf{q}^{w_i} Vector of relative wheelset track frame coordinates
- \mathbf{Q} Vectors of generalized applied forces and generalized quadratic-velocity inertia forces
- \mathbf{Q}^{susp} Vector of generalized suspension forces
- \mathbf{Q}_{fla}^{nor} Vector of generalized wheel–rail normal flange forces
- \mathbf{Q}^{nor, w_i} , $\mathbf{Q}_{tre}^{nor, w_i}$, $\mathbf{Q}_{fla}^{nor, w_i}$ Vector of generalized wheel–rail normal contact forces, generalized wheel–rail normal contact forces at tread and flange
- \mathbf{Q}^{tang} Vector of generalized tangential tread and flange forces
- R Radius of curved track [m]
- \mathbf{R}^t Position vector of a track frame with respect to the global frame
- R_x^t , R_y^t , R_z^t Position components of the position vector \mathbf{R}^t in *X*, *Y*, and *Z* direction
- \mathbf{R}^{bti} Position vector of the body track frame with respect to the global frame
- \mathbf{R}^{ijt} Relative distance vector from body track frames *it* to *jt*
- \mathbf{R}_P Absolute position vector of an point *P* in global frame
- $\bar{\mathbf{r}}^{ir}$ Position vectors of rail irregularity vectors of rail profile frames with respect to ideal rail profile frames

- $\bar{\mathbf{r}}^{ip}$ Position vectors of ideal railhead frames with respect to track frame
 \mathbf{r}_P^i Position vector of point P that belongs to body i with respect to the global frame
 $\bar{\mathbf{r}}^i$ Position vector of body frame with respect to its the body track frame
 $\bar{\mathbf{r}}^{wi}$ Position vector of wheelset frame with respect to the wheelset track frame
 $\bar{\mathbf{r}}_c^{wi}$ Position vectors of contact points with respect to the wheelset track frame
 $\dot{\bar{\mathbf{r}}}_c^{wi}$ Time derivative of the position vector $\bar{\mathbf{r}}_c^{wi}$
 $\bar{\mathbf{r}}_{c,fla}^{wi}$ Position vectors of flange contact points with respect to the wheelset track frame
 $\bar{\mathbf{r}}_c^{ip}$ Position vectors of contact points with respect to the track frame
 $\dot{\bar{\mathbf{r}}}_c^{ip}$ Time derivative of the position vector $\bar{\mathbf{r}}_c^{ip}$
 $\bar{\mathbf{r}}_{c,fla}^{ip}$ Position vectors of flange contact points with respect to the track frame
 r_0 Rolling radius of the wheel when centered in the track [m]
 \mathbf{s} Vector of wheel and rail surface parameters
 \mathbf{s}_{fla} Vector of flange surface parameters
 s Arc-length associated with the track centerline [m]
 Δs Increment of the arc-length [m]
 $\Delta \hat{s}, \Delta \bar{s}$ Increment of the transition lengths of the regularization for the tread–flange transition [mm]
 s^{wi} Arc-length associated with the wheelset track centerline [m]
 s_1^w Transverse wheel surface parameter [m]
 s_2^w Angular wheel surface parameter [m]
 s_1^r Longitudinal rail surface parameter [m]
 s_2^r Transverse rail surface parameter [m]
 s_{fla}^w, s_{fla}^r Transverse wheel surface and rail surface parameters associated with flange contact [m]
 s^k Lateral positions of the contact point in the *KEC*-equivalent profiles [m]
 s_{clt}^w, s_{clt}^r Contact lookup table of wheel and rail surface parameters [m]
 s_{klt}^w, s_{klt}^r *KEC*-lookup table of wheel and rail surface parameters [m]
 $\hat{\mathbf{t}}_c^{wli}$ Unit tangential vector with respect to the wheel surface at the contact point
 $\bar{\mathbf{t}}_c^{wi}$ Unit tangential vector defined in wheelset track frame
 $\bar{\mathbf{t}}_{c,fla}^{wi}$ Unit tangential vector for flange contact point defined in wheelset track frame
 $\hat{\mathbf{u}}_P^i$ Position vector of point P with respect to body frame for body i in the body frame
 $\hat{\mathbf{u}}_c^{wli}$ Position vectors of the contact points at left and right wheel with respect to the wheelset intermediate frame
 $\hat{\mathbf{u}}_c^{ip}$ Position vectors of contact points with respect to rail profile frames
 V Forward velocity [m/s]
 y^{ir} Position components of rail irregularity in lateral direction
 y^{wi} Lateral displacement of the wheelset frame with respect to the wheel track frame (ideal track centerline)
 \bar{y}^{wi} Lateral displacement of the wheelset frame with respect to the irregular track centerline
 $y_{\max}^{wi}, y_{\min}^{wi}$ Maximum and minimum values of the lateral displacements of wheelset
 Δy^{wi} Increment of the lateral displacements of wheelset
 z^{ir} Position components of rail irregularity in vertical direction
 z^{wi} Vertical displacement of the wheelset frame with respect to the wheel track frame
 \bar{z}^{wi} Vertical displacement of the wheelset frame with respect to the irregular track centerline
 z_{clt} Contact lookup table of wheelset vertical displacement
 λ Array of Lagrange multipliers
 β Orientation angle of the rail profiles
 δ Linearized rotation angle due to the irregularity

δ^{wi} , $\dot{\delta}^{wi}$ Wheel–rail penetration at the flange contact and its rate

φ^t , θ^t , ψ^t Euler angles which define the orientation of track centerline with respect to global frame

φ^{wi} , θ^{wi} , ψ^{wi} Euler angles which define the orientation of relative wheel frame with respect to wheelset track frame

$\bar{\varphi}^{wi}$ Roll angle of the wheelset frame with respect to the irregular track centerline

φ_{cli} Contact lookup table of wheelset roll angle

μ Coefficient of friction

ξ_a Alignment of the railhead centerlines' irregularities [mm]

ξ_c Cross level of the railhead centerlines' irregularities [mm]

ξ_g Gauge variation of the railhead centerlines' irregularities [mm]

$\xi_{g,max}$, $\xi_{g,min}$ Maximum and minimum values of the gauge variation the railhead centerlines' irregularities [mm]

$\Delta\xi_g$ Increment of the gauge variation [mm]

ξ_v Vertical profile of the railhead centerlines' irregularities [mm]

$(O; X, Y, Z)$ Global frame

$(O^t; X^t, Y^t, Z^t)$ Track frame

$(O^i; X^i, Y^i, Z^i)$ Body frame

$(O^{lrp}; X^{lrp}, Y^{lrp}, Z^{lrp})$ Left rail profile frame

$(O^{rrp}; X^{rrp}, Y^{rrp}, Z^{rrp})$ Right rail profile frame

$(O^{bti}; X^{bti}, Y^{bti}, Z^{bti})$ Body track frame for body i

$(O^{wi}; X^{wi}, Y^{wi}, Z^{wi})$ Wheelset frame

$(O^{wli}; X^{wli}, Y^{wli}, Z^{wli})$ Wheelset intermediate frame

1 Introduction

A railway vehicle is a complex mechanical system that consists of a large number of bodies, wheel/rail contacts, and complicated suspension elements. Because of the high computational loads, the real-time simulation of the railway vehicle is challenging [1], especially with respect to the wheel–rail contact [2]. Computationally efficient and physically accurate simulations can be used for a wide variety of applications, such as optimization problems and embedded models. Moreover, to analyze the validity and efficiency of different numerical methods, benchmark problems can be found in the literature. Examples include the Manchester benchmark [3] and the switches and crossings benchmarks [4].

In computational railway dynamics, a number of formalisms have been introduced to analyze wheel/rail contact. Among these, two well-known approaches are commonly used in multibody railway simulations. The first is the elastic approach [5–7], where interpenetration between wheel/rail surfaces is allowed, and normal contact forces are computed based on interpenetration and Hertzian parameters. However, the large magnitude Hertzian parameters that come about when applying the elastic approach can result in a stiff system of ordinary differential equations. Therefore, implicit integrators with small time steps are used, which increases computational cost. In [8], Liu and Bruni compared Hertzian and non-Hertzian wheel/rail contact models for multibody simulation. In [6], the Hertzian and non-Hertzian models are studied and compared to give the insight of proper selection of the parameters for better computational accuracy and efficiency. The second approach is the constraint method, which is also called Lagrange multiplier method. In this approach, the contact between wheel and rail is described with a set of kinematic constraint equations [9]

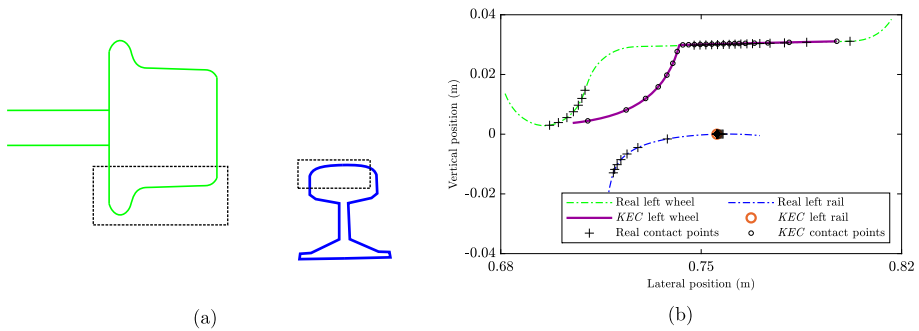


Fig. 1 (a) Real S1002 wheel profile and LB.140-AREA rail profile geometry, (b) contact points on real wheel/rail profiles and corresponding *KEC*-equivalent wheel/rail profiles, the real wheel/rail profiles correspond to the profiles in the dashed boxes on the left

to ensure both surfaces are in contact without penetration or separation. In this way, the normal contact forces can be computed using the Lagrange multipliers that are associated with the constraint equations.

Two constraint-based formulations for the wheel/rail contact in a multibody simulation have been studied here. The first is the *LUT* method. The *LUT* method is an offline constraint approach where the contact detection part of the analysis is carried out in a preprocessing stage and the solution is stored in a lookup table. In the subsequent dynamic simulations, the contact points are obtained by interpolating between the stored data points [10–12]. In the work of [10], an offline contact lookup table is used to predict tread contact point and an online elastic approach is used to predict flange contact. This algorithm was later extended to cover the combination of nodal and nonconformal contact detection to solve contact point jumps that can occur in turnouts [11]. The authors of [12] propose a constraint contact *LUT* method that accounts for track irregularities with two entries, lateral displacement, and track gauge variation. Using the Kalker book of tables for non-Hertzian contact (KBTNH) to build the lookup table is proposed [13] to generate the fast creep force. In that study, a regularization of the nonelliptical contact patch is introduced to solve the wheel/rail normal contact problem with non-Hertzian methods. In addition, the KBTNH is extended to include the full symmetry relations for creep forces and moment [14]. A comprehensive analysis to improve accuracy and efficiency when interpolating the lookup tables with multiple input parameters is presented in [15].

The second approach is the *KEC* method. The *KEC* method used here is an online constraint approach that was first proposed [16] to model wheel/rail contact. In the *KEC* method, the wheel/rail profile combination (see Fig. 1(a)) is established using an equivalent wheel profile (see solid lines in Fig. 1(b)) in contact with an infinitely narrow rail, which yields the equivalent allowable relative motion. This equivalent profile combination produces the same wheel/rail contact kinematics as the real wheel/rail profiles. As shown in Fig. 1(b), the contact points for a set of discrete values of the wheelset lateral displacement are located at the real wheel/rail profiles (dotted-dashed lines in Fig. 1(b)). The wheelset with wheels S1002 profile and rails LB.140-AREA profile are considered in the figure, and the geometry parameters for both wheel/ rail profiles are listed in Table 1. Accordingly, the corresponding contact points can be found on the *KEC*-equivalent profiles. See the solid lines in Fig. 1(b). The exact position of the contact points can be determined from the online solution of the *KEC* constraints. In addition, [17] proposed a regularization method to apply a continuous transition from tread to flange between each wheel/rail pair for the dynamic simulation of

Table 1 Parameters for the wheel/rail profile combination that illustrate the two-point contact scenario

Model	L_w (m)	L_r (m)	r_0 (m)	β (rad)
Parameters	0.7515	0.7555	0.457	1/40

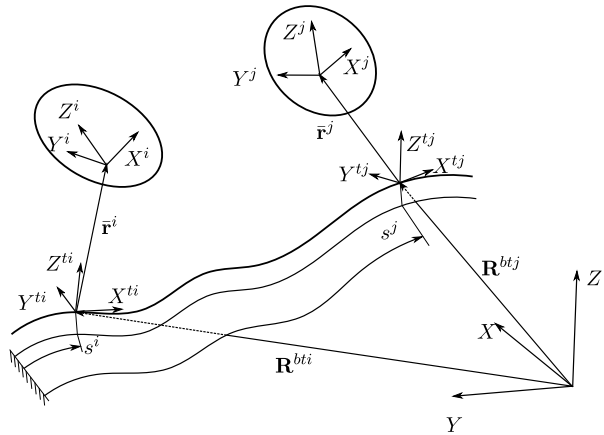
a two-point contact scenario. This regularization method can help avoid the finite contact point jumps that occur between tread and flange.

Active research to develop computationally efficient models for multibody simulation of railroad vehicles is ongoing. Focus areas include linearizing the equations of motion [18], establishing an appropriate time integration method [18], efficiently detecting wheel/rail contact, and formalizing corresponding contact computation [10, 19, 20]. An explicit predictor–corrector integration method has been developed to solve the large-scale coupling dynamics of a railway vehicle and track system using the explicit method as the predictor and the implicit Newmark method as a corrector [21]. A quasielastic contact model proposed in [19] employs a two-dimensional spline approximation of contact solutions and stores the spline coefficients as a table in a pre–processing stage. The method is implemented in the wheel/rail module of the commercial software SIMPACK. In addition, a modal substructuring approach is proposed in [22] to improve the computational efficiency of a vehicle–track simulation by using a reduced number of modal coordinates. A hybrid contact detection algorithm is developed in [10] to eliminate the online iterative search for the second contact point. Moreover, a differential contact model is presented in [20] to reduce the computation cost by integrating the differential modeling of the wheel/rail contact problem with multibody modeling of the railway dynamics. The work reported in [18] proves that a simplified model based on weakly coupled lateral and vertical dynamics solves about four times faster than the full three-dimensional (3D) coupled dynamic model.

Several works have been published to study the pros and cons of the *LUT* and *KEC* methods with respect to computational efficiency. In the case of vehicle ML95 (Lisbon subway company) negotiating a curved track without irregularities in [12], a numerical solution is reached six times faster using the *LUT* method instead of directly applying contact constraints using the Matlab integrator *ode45*, *ode113*, and *ode4*. The authors of [16] conclude that the *KEC* method reduces computational cost up to 20% compared to the *LUT* method using Matlab implicit integrator *ode15s*. And in [17], the *KEC* method proved to be 12.6 times more efficient than the 3D elastic contact simulation with the Matlab implicit integrator *ode15s*. Expanding on this preceding research, the objective of this work is to study the numerical and computational aspects of both methods using the multibody model of the Manchester wagon 1 proposed in [3]. Since efficient and stable time integration is a key issue [23] for the computational multibody dynamics of railroad vehicles, different well-known numerical integrators have been studied and compared in this paper. Two fixed-step-size integrators, RK4 and a predictor–corrector based ABM, and two variable-step-size Matlab integrators, *ode15s* and *ode45*, are applied to solve railway dynamic problems. This will reveal numerical and computational aspects of the constraint-based contact approaches and will look at ways an accurate physics-based wheel/rail vehicle system could be analyzed in real-time.

The following Sect. 2 of this paper briefly introduces the multibody modeling of railway vehicles. Computer simulation of railway vehicles is presented in Sect. 3. Different integration methods are briefly introduced in Sect. 4. Sect. 5 compares the kinematics solutions of the modeled Manchester Wagon 1 with those obtained using standard commercial software and investigates the numerical and computational aspects of the *KEC* and *LUT* methods in detail. Finally, Sect. 6 provides a summary and offers conclusions.

Fig. 2 Kinematics of the bodies i and j of a railway vehicle with relative body track frame coordinates, vectors \mathbf{R}^{bti} and \mathbf{R}^{btj} are the absolute position vectors of body track frames $(O^i; X^i, Y^i, Z^i)$ with respect to global frame $(O; X, Y, Z)$, $\bar{\mathbf{r}}^i$ and $\bar{\mathbf{r}}^j$ are position vectors of body i and j with their track frames



2 Multibody modeling of railway vehicles

This section introduces details of the multibody modeling of railway vehicles. To this end, track kinematics vehicle kinematics and wheel/rail contact will be discussed. More insight into the kinematics and contact mechanism can be found in [12, 16, 24].

2.1 Track geometry and kinematics

In the multibody simulation of railway vehicles, the use of moving coordinate systems that follow the vehicles motion along the track centerline has been widely used in the literature [25, 26]. This method has advantages in the interpretation of the results because system coordinates are referred to track geometry. This work uses the so-called ‘track frame’ $(O^i; X^i, Y^i, Z^i)$ for each vehicle body. The ‘track frame is defined along the track centerline and has the X^i axis parallel to the tangential line of the track centerline, the Y^i axis connecting the centerline of the two rails and the Z^i axis perpendicular to the plane of the rails. See Fig. 2.

The absolute position vector and rotation matrix from a track frame to the global frame can be expressed as the functions of the arc-length s as follows:

$$\mathbf{R}^i(s) = \begin{bmatrix} R_x^i(s) \\ R_y^i(s) \\ R_z^i(s) \end{bmatrix},$$

$$\mathbf{A}^i(s) = \begin{bmatrix} \cos(\theta^i) \cos(\psi^i) & \sin(\varphi^i) \sin(\theta^i) \cos(\psi^i) - \cos(\varphi^i) \sin(\psi^i) & \sin(\varphi^i) \sin(\psi^i) + \cos(\varphi^i) \sin(\theta^i) \cos(\psi^i) \\ \cos(\theta^i) \sin(\psi^i) & \cos(\varphi^i) \cos(\psi^i) + \sin(\varphi^i) \sin(\theta^i) \sin(\psi^i) & \cos(\varphi^i) \sin(\theta^i) \sin(\psi^i) - \sin(\varphi^i) \cos(\psi^i) \\ -\sin(\theta^i) & \sin(\varphi^i) \cos(\theta^i) & \cos(\varphi^i) \cos(\theta^i) \end{bmatrix}, \tag{1}$$

where R_x^i, R_y^i, R_z^i are the components of the position vectors in the $X, Y,$ and Z direction, ψ^i (*azimut* or *heading* angle), θ^i (*vertical slope*, positive when downwards in the forward direction), and φ^i (*cant* or *superelevation* angle) are Euler angles which define the track centerline orientation.

With the description of the track frame, the geometry of the left and right rails centerlines can be obtained by defining two additional profile frames (left rail profile frame

Fig. 3 Description of track irregularity geometry: \vec{r}^{lrp} and \vec{r}^{rrp} are position vectors of ideal left and right railhead frames with respect to track frame, \vec{r}^{lir} and \vec{r}^{rir} are irregular vectors of left and right rail heads, \hat{u}_{lc}^{lrp} and \hat{u}_{lc}^{rrp} are position vectors of contact points with respect to rail profile frames, β is the orientation angle of the rail profiles, δ is the linearized rotation angle due to the irregularity, and L_r is the rail profile positioning with respect to the track centerline

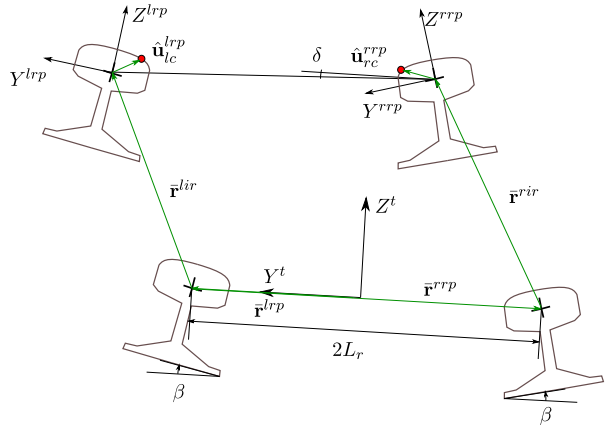
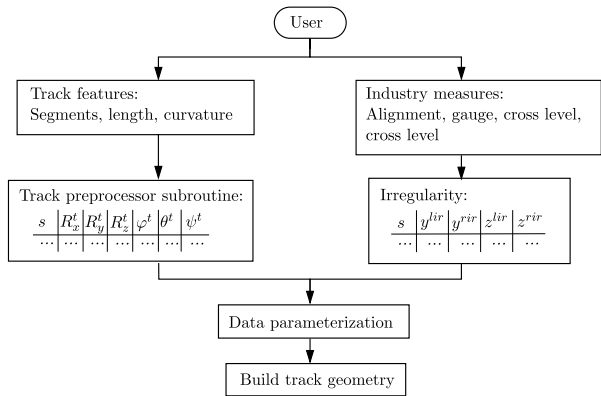


Fig. 4 Procedure of building track preprocessor



$\langle O^{lrp}; X^{lrp}, Y^{lrp}, Z^{lrp} \rangle$ and right rail profile frame $\langle O^{rrp}; X^{rrp}, Y^{rrp}, Z^{rrp} \rangle$ at the railheads. Both rail profiles are separated a distance $2L_r$ and rotated an angle β in the ideal track configuration, as it is shown in Fig. 3. In addition, this ideal geometry description allows a straightforward definition of the irregular track by means of the left and right irregularity vectors \vec{r}^{lir} and \vec{r}^{rir} and the linearized irregularity angle δ defined as $\delta = (z^{lir} - z^{rir})/2L_r$. See Fig. 3. Note that the components of the irregularity vectors can be extracted from the following well-known centerlines' irregularities measured in the railway industry:

- Alignment (ξ_a) $\xi_a(s) = (y^{lir} + y^{rir})/2,$
- Vertical profile (ξ_v) $\xi_v(s) = (z^{lir} + z^{rir})/2,$
- Gauge variation (ξ_g) $\xi_g(s) = y^{lir} - y^{rir},$
- Cross level (ξ_c) $\xi_c(s) = z^{lir} - z^{rir}.$

All of these components can be implemented in a track preprocessor subroutine to parameterize the three-dimensional (3D) ideal track centerline and track irregularities as continuous functions (cubic splines in this work) of the longitudinal arc-length trajectory coordinate s . A schematic representation of the procedure used for the track preprocessor is presented in Fig. 4.

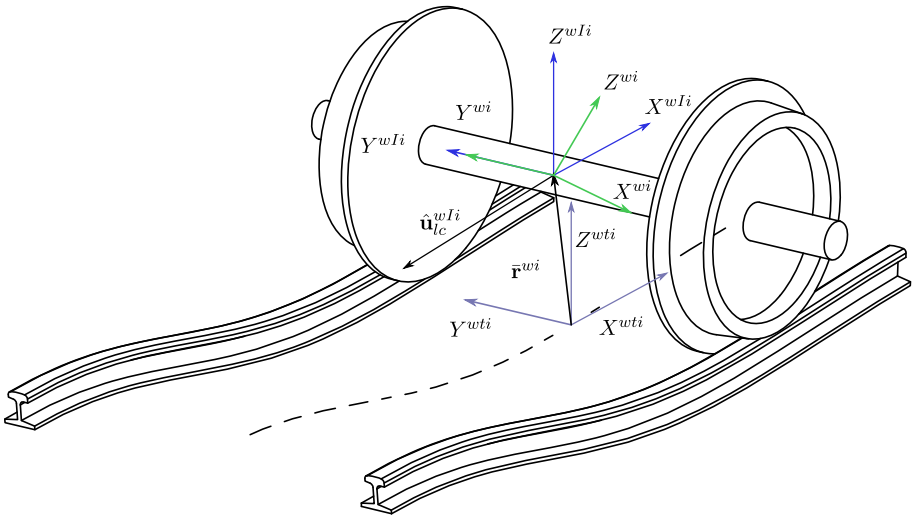


Fig. 5 Frames for rigid wheelset kinematics: $\bar{\mathbf{r}}^{wi}$ is the position vector of wheelset frame with respect to the wheelset track frame in the wheelset track frame, $\hat{\mathbf{u}}_{lc}^{wIi}$ is the position vector of the left contact point at left wheel with respect to the wheelset intermediate frame

In this work, the track preprocessor for track centerline is discretized with constant step $\Delta s = 0.5$ m, which results in a Matlab storage size of around 200 kB (matrix size 2000×7) for 1000 m track. As for track irregularities, the track preprocessor is discretized with $\Delta s = 0.1$ m, which results in a Matlab storage size of around 300 kB (matrix size 10000×5) for 1000 m track.

2.2 Wheel/rail contact

As shown in Fig. 2, a body track frame $\langle O^i; X^i, Y^i, Z^i \rangle$ can be associated with body i at any time-instant, when a body moves along the track with arc length s^i . Therefore, the position and orientation of the body track frame can be computed according to Eq. (1), as:

$$\mathbf{R}^i = \mathbf{R}^t(s^i), \quad \mathbf{A}^i = \mathbf{A}^t(s^i). \tag{2}$$

Similarly, a wheelset track frame $\langle O^{wi}; X^{wi}, Y^{wi}, Z^{wi} \rangle$ is associated with the wheelset i and with time. See Fig. 5. The position and orientation of the wheelset track frame can be calculated by replacing s^i with s^{wi} in Eq. (2).

The set of relative wheelset track frame coordinates of a rigid wheelset i (superscript wi) is given by:

$$\mathbf{q}^{wi} = [s^{wi} \quad y^{wi} \quad z^{wi} \quad \varphi^{wi} \quad \theta^{wi} \quad \psi^{wi}]^T, \tag{3}$$

where y^{wi} is the lateral displacement and z^{wi} is the vertical displacement of the wheelset frame with respect to the wheel track frame, respectively, and the X -component along the track centerline is zero; φ^{wi} , θ^{wi} , and ψ^{wi} are Euler angles which define the orientation of relative wheel frame with respect to wheelset track frame.

The position vectors of right and left contact points, defined in the right and left railheads as shown in Fig. 3, are written with respect to wheelset track frame:

$$\begin{aligned} \bar{\mathbf{r}}_{rc}^{rrp} &= \bar{\mathbf{r}}^{rrp} + \bar{\mathbf{r}}^{rir} + \mathbf{A}^{t,rrp} \hat{\mathbf{u}}_{rc}^{rrp}, \\ \bar{\mathbf{r}}_{lc}^{lrp} &= \bar{\mathbf{r}}^{lrp} + \bar{\mathbf{r}}^{lir} + \mathbf{A}^{t,lrp} \hat{\mathbf{u}}_{lc}^{lrp}, \end{aligned} \tag{4}$$

where *lr* and *rr* represent *left rail* and *right rail*, $\bar{\mathbf{r}}^{rrp}$ and $\bar{\mathbf{r}}^{lrp}$ are position vectors of ideal left and right railhead frames with respect to wheelset track frame, $\bar{\mathbf{r}}^{lir}$ and $\bar{\mathbf{r}}^{rir}$ are rail irregularity vectors of rail profile frames with respect to ideal left and right rail profile frames, $\mathbf{A}^{t,lrp}$ and $\mathbf{A}^{t,rrp}$ are rotation matrices from the railhead frames to the wheelset track frame, $\hat{\mathbf{u}}_{rc}^{rrp}$ and $\hat{\mathbf{u}}_{lc}^{lrp}$ contain the components of the position vector of contact points in the rail profile frames as shown in Fig. 3. The vectors and matrices are written as

$$\begin{aligned} \bar{\mathbf{r}}^{lrp} &= \begin{bmatrix} 0 \\ L_r \\ 0 \end{bmatrix}, & \bar{\mathbf{r}}^{rrp} &= \begin{bmatrix} 0 \\ -L_r \\ 0 \end{bmatrix}, & \bar{\mathbf{r}}^{lir}(s^{wi}) &= \begin{bmatrix} 0 \\ y^{lir} \\ z^{lir} \end{bmatrix}, & \bar{\mathbf{r}}^{rir}(s^{wi}) &= \begin{bmatrix} 0 \\ y^{rir} \\ z^{rir} \end{bmatrix}, \\ \mathbf{A}^{t,lrp}(s^{wi}) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta + \delta) & -\sin(\beta + \delta) \\ 0 & \sin(\beta + \delta) & \cos(\beta + \delta) \end{bmatrix}, & & & & & (5) \\ \mathbf{A}^{t,rrp}(s^{wi}) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\beta + \delta) & -\sin(-\beta + \delta) \\ 0 & \sin(-\beta + \delta) & \cos(-\beta + \delta) \end{bmatrix}, & & & & & \end{aligned}$$

The position vectors of contact points $\hat{\mathbf{u}}_{rc}^{rrp}$ and $\hat{\mathbf{u}}_{lc}^{lrp}$ in the rail profiles from Eq. (4) are parametrized according to the railhead profile geometry as shown in Fig. 6:

$$\hat{\mathbf{u}}_{rc}^{rrp} = \begin{bmatrix} 0 \\ s_2^{rr} \\ h^{rr}(s_2^{rr}) \end{bmatrix}, \quad \hat{\mathbf{u}}_{lc}^{lrp} = \begin{bmatrix} 0 \\ s_2^{lr} \\ h^{lr}(s_2^{lr}) \end{bmatrix}, \tag{6}$$

where s_2^{lr} and s_2^{rr} are left and right transverse rail surface parameters, respectively, and h^{lr} and h^{rr} are the functions that define the railhead profiles.

Also, special treatment is given to the railway vehicle wheelset bodies in a railway vehicle. An additional frame called ‘wheelset intermediate frame’ (*wI*) is defined as shown in Fig. 5. The *wI*-frame is defined as a body frame but shows no pitch rotation. As will be shown in the following section, it is a convenient frame to define contact point position vectors, because they are not influenced by the wheelset pitch rotation, which is particularly high in these bodies. More details about body and wheelset kinematics can be found in [12, 16, 24]. The position vector of contact points on the surface of the left and right wheel profile in Fig. 5 can be obtained in wheelset track frame, such as

$$\begin{aligned} \bar{\mathbf{r}}_{lc}^{wi} &= \bar{\mathbf{r}}^{wi} + \mathbf{A}^{wti,wI} \hat{\mathbf{u}}_{lc}^{wI}, \\ \bar{\mathbf{r}}_{rc}^{wi} &= \bar{\mathbf{r}}^{wi} + \mathbf{A}^{wti,wI} \hat{\mathbf{u}}_{rc}^{wI}, \end{aligned} \tag{7}$$

where vector $\bar{\mathbf{r}}^{wi} = [0 \quad y^{wi} \quad z^{wi}]^T$ is the position vector of the wheelset frame with respect to the wheelset track frame, $\hat{\mathbf{u}}_{lc}^{wI}$ and $\hat{\mathbf{u}}_{rc}^{wI}$ are the position vectors of the contact points at the left and right wheels with respect to the wheelset intermediate frame, and $\mathbf{A}^{wti,wI}$ is the

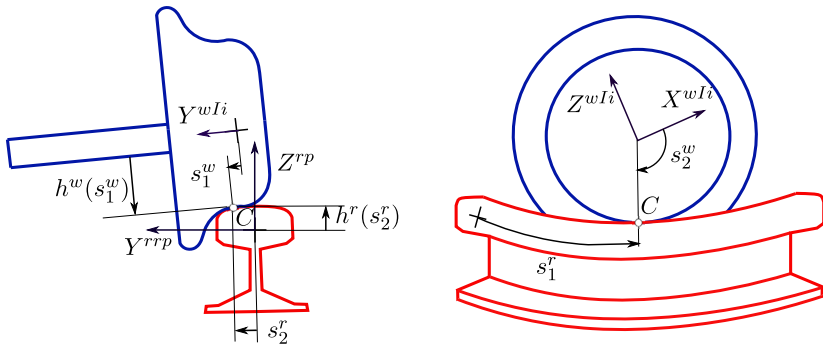


Fig. 6 Wheel profile and rail profile geometry: s_1^w and s_2^w are transverse and angular wheel surface parameters, and s_1^r and s_2^r are longitudinal and transverse rail surface parameters, respectively

rotation matrix of the wheelset intermediate frame with respect to the wheelset track frame, which is given by the following equation:

$$\mathbf{A}^{wti,wi} = \mathbf{A}^{wti,wli} \mathbf{A}^{wli,wi}, \tag{8}$$

where $\mathbf{A}^{wti,wli}$ is the rotation matrix of the wheelset intermediate frame with respect to the wheelset track frame, $\mathbf{A}^{wli,wi}$ is the rotation matrix of the wheelset frame with respect to the wheelset intermediate frame. These are given by

$$\mathbf{A}^{wti,wli} = \begin{bmatrix} 1 & -\psi^{wi} & 0 \\ \psi^{wi} & 1 & -\varphi^{wi} \\ 0 & \varphi^{wi} & 1 \end{bmatrix}, \quad \mathbf{A}^{wli,wi} = \begin{bmatrix} \cos(\theta^{wi}) & 0 & \sin(\theta^{wi}) \\ 0 & 1 & 0 \\ -\sin(\theta^{wi}) & 0 & \cos(\theta^{wi}) \end{bmatrix}. \tag{9}$$

The position vectors of contact points $\hat{\mathbf{u}}_{rc}^{wli}$ and $\hat{\mathbf{u}}_{lc}^{wli}$ in the wheel profiles from Eq. (7) are parametrized according to the wheel profile geometry as shown in Fig. 6:

$$\hat{\mathbf{u}}_{rc}^{wli} = \begin{bmatrix} h^{rw}(s_1^{rw}) \cos(s_2^{rw}) \\ -L_w + s_1^{rw} \\ -h^{rw}(s_1^{rw}) \sin(s_2^{rw}) \end{bmatrix}, \quad \hat{\mathbf{u}}_{lc}^{wli} = \begin{bmatrix} h^{lw}(s_1^{lw}) \cos(s_2^{lw}) \\ L_w + s_1^{lw} \\ -h^{lw}(s_1^{lw}) \sin(s_2^{lw}) \end{bmatrix}, \tag{10}$$

where s_1^w and s_2^w are the transverse and angular wheel surface parameters, respectively, and h^{lw} and h^{rw} are the functions that define the wheel profiles.

Based on the parameterization of the wheel and rail surfaces, contact can be modeled using the already defined elastic or constraint approach. For both approaches, 3D surface-to-surface contact can be reduced to a planar two-dimensional (2D) curve-to-curve contact assuming that the contact points lie in the $Y^{wti}-Z^{wti}$ plane of the wheelset track frame. This decreases the number of surface parameters from four to two, since the contact points at the left and right rails have the same arc-length parameter as the wheelset ($s_1^r = s_1^w$), and the angular wheel parameters of the contact points are constant ($s_2^w = \pi/2$) in the wI -frame. Therefore, the simplified nomenclature $s^w = s_1^w$ and $s^r = s_2^r$ can be and is used in the rest of this paper. However, the main implication of this planar contact is that the wheelset yaw angle is neglected for the location of the contact points, which avoids the consideration of the so-called lead-lag contacts (flange contact points that are longitudinally displaced with respect to tread contacts). These contacts have an effect when the vehicle is negotiating very

narrow curves, which usually occurs at very low velocities. Nevertheless, the planar contact approach is sufficiently accurate for most practical applications as shown in [12].

The use of 2D wheel/rail contact using the constraint approach is considered in the following. To this end, the following subsections briefly describe the basis of the following two planar constraint contact methods: contact lookup tables and the *KEC* method.

2.3 LUT method

Creation of constraint contact lookup table

The procedure to build a constraint contact lookup table starts by solving the contact constraints for different values of the wheelset relative position with respect to the track. The algorithm is summarized as follows:

1. Two sets of discrete numerical values are assigned to the lateral displacement of the wheelset y^{wi} and gauge variation ξ_g :

$$\begin{aligned}
 y^{wi} &= y_{\min}^{wi} + \Delta y^{wi}, & y^{wi} &\in [y_{\min}^{wi}, y_{\max}^{wi}], \\
 \xi_g &= \xi_{g,\min} + \Delta \xi_g, & \xi_g &\in [\xi_{g,\min}, \xi_{g,\max}],
 \end{aligned}
 \tag{11}$$

where Δy^{wi} and $\Delta \xi_g$ are increments.

2. The wheelset coordinates s^{wi} , θ^{wi} , and ψ^{wi} are set to zero (contact points are assumed to lie in the $Y^{wi}-Z^{wi}$ plane of the wheelset track frame).
3. For a one-point contact scenario (when two wheel/tread contacts occur at both wheels), the six contact constraint equations of Eq. (12) are solved for different values of y^{wi} and the nominal distance $2L_r^* = 2L_r + \xi_g$ in two for-loops to obtain six outputs: the wheelset kinematics z^{wi} , φ^{wi} , and the vector of tread surface parameters \mathbf{s} ,

$$\mathbf{f}(z^{wi}, \varphi^{wi}, \mathbf{s}) = \begin{bmatrix} \bar{\mathbf{r}}_{lc}^{wi} - \bar{\mathbf{r}}_{lc}^{lrp} \\ (\bar{\mathbf{t}}_{lc}^{wi})^T \bar{\mathbf{n}}_{lc}^{lrp} \\ \bar{\mathbf{r}}_{rc}^{wi} - \bar{\mathbf{r}}_{rc}^{rrp} \\ (\bar{\mathbf{t}}_{rc}^{wi})^T \bar{\mathbf{n}}_{rc}^{rrp} \end{bmatrix} = \mathbf{0},
 \tag{12}$$

where rc represents right wheel and lc left wheel for one wheelset, $\mathbf{s} = [s^{lw} \ s^{lr} \ s^{rw} \ s^{rr}]^T$ is the vector of the surface parameters, $\bar{\mathbf{t}}_c^{wi}$ is the unit tangential vector, and $\bar{\mathbf{n}}_c^{rp}$ is the normal vector defined in the wheelset track frame, which can be expressed as follows:

$$\begin{aligned}
 \bar{\mathbf{t}}_c^{wi} &= \mathbf{A}^{wti,wi} \hat{\mathbf{t}}_c^{wli}, \\
 \bar{\mathbf{n}}_c^{rp} &= \mathbf{A}^{t,rp} \hat{\mathbf{n}}_c^{rp},
 \end{aligned}
 \tag{13}$$

where $\hat{\mathbf{t}}_c^{wli}$ is the unit tangential vector with respect to the wheel surface at the contact point and $\hat{\mathbf{n}}_c^{rp}$ is the unit normal vector with respect to the railhead surface at the contact point [16, 25]. In Eq. (12), given the value of lateral displacement y^{wi} , six contact constraints for one wheelset are solved to get six outputs: the wheelset coordinates z^{wi} and φ^{wi} , and the surface parameters \mathbf{s} of the contact points at the tread.

4. The solution of Eq. (12) is stored in a lookup table with two entries. The lookup table defines the following functions:

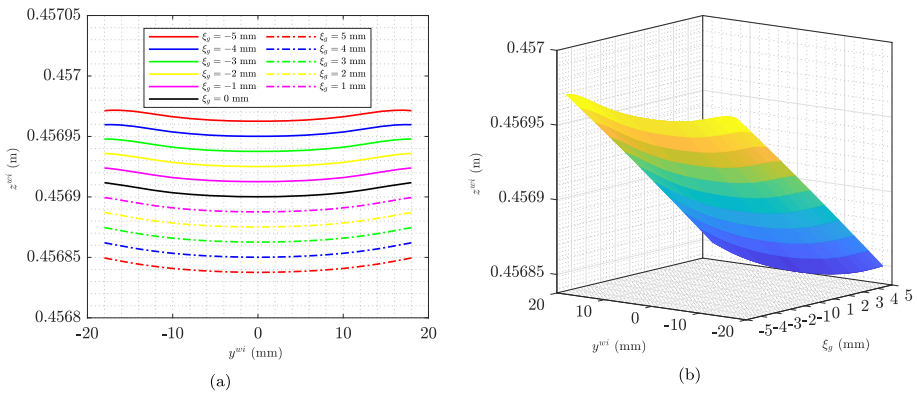


Fig. 7 (a) Wheelset vertical displacement with respect to track centerline for different values of gauge irregularity; (b) Contact lookup table for wheelset vertical displacement z^{wh} with two entries. As it is observed, there is no wheel climbing due to the nature of the elastic flange contact used

$$\begin{aligned}
 z^{wi} &= z_{clt}(y^{wi}, \xi_g), & \varphi^{wi} &= \varphi_{clt}(y^{wi}, \xi_g), \\
 s^{lw} &= s_{clt}^{lw}(y^{wi}, \xi_g), & s^{lr} &= s_{clt}^{lr}(y^{wi}, \xi_g), & s^{rw} &= s_{clt}^{rw}(y^{wi}, \xi_g), & s^{rr} &= s_{clt}^{rr}(y^{wi}, \xi_g),
 \end{aligned}
 \tag{14}$$

where *clt* refers contact lookup table. Figure 7(a) shows the wheelset vertical displacement coordinate z^{wi} with respect to the track centerline within a range of track gauge variations ξ_g and Fig. 7(b) illustrates the contact lookup table.

Contact detection for two-point contact scenario

For the two-point contact scenario, contact detection at the flange is addressed using the maximum relative-indentation condition [5]. Therefore, two nonlinear equations are solved to obtain two flange surface parameters $s_{fla} = [s_{fla}^w \ s_{fla}^r]^T$ as follows:

$$\mathbf{f}(s_{fla}) = \begin{bmatrix} (\bar{\mathbf{t}}_{c,fla}^{wi})^T (\bar{\mathbf{r}}_{c,fla}^{wi} - \bar{\mathbf{r}}_{c,fla}^{rp}) \\ (\bar{\mathbf{t}}_{c,fla}^{wi})^T \bar{\mathbf{n}}_{c,fla}^{rp} \end{bmatrix} = \mathbf{0},
 \tag{15}$$

where $\bar{\mathbf{t}}_{c,fla}^{wi}$ and $\bar{\mathbf{n}}_{c,fla}^{rp}$ are unit tangential and normal vectors, $\bar{\mathbf{r}}_{c,fla}^{wi}$ and $\bar{\mathbf{r}}_{c,fla}^{rp}$ are position vectors of the contact points on the wheel and railhead, all the above vectors are associated with the wheel–flange contact, and *c* represents left contact *lc* or right contact *rc*. Contact point detection at both the tread and flange for the *LUT* method is determined in a pre-processing stage that builds the contact lookup tables. During the dynamic simulations, the contact lookup table is used to interpolate the stored discrete data to get the contact point location.

Contact force computation for two-point contact scenario

For the *LUT* method, constraint contact is not appropriate when the number of contact points varies (i.e., from one contact point to the two-contact point scenario). For this reason, when flange contact occurs in the two-point contact scenario (before wheel climbing), a hybrid method is usually adopted [12]. In this approach, wheel/tread contact with the rail

is considered using the constraint method and the wheel/flange contact with the rail is considered using the elastic method. The equations of motion for a railway vehicle subject to contact lookup table constraints can be written in the form of the following differential-algebraic equation (DAE):

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{q}} + (\mathbf{C}_{\mathbf{q}}^{LUT})^T \boldsymbol{\lambda} &= \mathbf{Q} + \mathbf{Q}_{fla}^{nor} + \mathbf{Q}^{tang} + \mathbf{Q}^{susp}, \\ \mathbf{C}^{LUT}(\mathbf{q}, t) &= \mathbf{0}, \end{aligned} \tag{16}$$

where \mathbf{M} is the vehicle generalized mass matrix, \mathbf{C}^{LUT} includes the LUT constraint equations $\mathbf{C}^{wi,LUT}$ for the wheelset and forward velocity holonomic constraints, $\mathbf{C}_{\mathbf{q}}^{LUT}$ is the corresponding Jacobian matrix, $\boldsymbol{\lambda}$ is the array of Lagrange multipliers, \mathbf{Q}^{tang} is the vector of generalized tangential tread and flange forces, \mathbf{Q}^{susp} is the vector of generalized suspension forces, and \mathbf{Q} includes the vectors of generalized applied forces and generalized quadratic-velocity inertia forces. The wheelset LUT constraint equations contain two independent nonlinear equations using the lookup tables for vertical displacement and roll angle as presented in [12],

$$\mathbf{C}^{wi,LUT}(\mathbf{q}^{wi}) = \begin{bmatrix} \bar{z}^{wi} - z_{clt}(\bar{y}^{wi}, \xi_g) \\ \bar{\varphi}^{wi} - \varphi_{clt}(\bar{y}^{wi}, \xi_g) \end{bmatrix} = \mathbf{0}, \tag{17}$$

where \bar{y}^{wi} , \bar{z}^{wi} , and $\bar{\varphi}^{wi}$ are wheelset kinematics with respect to the irregular track centerline, which have the relationship with respect to those related to the ideal track centerline:

$$\bar{y}^{wi} = y^{wi} - \xi_a, \quad \bar{z}^{wi} = z^{wi} - \xi_v, \quad \bar{\varphi}^{wi} = \varphi^{wi} - \frac{\xi_c}{2L_r}. \tag{18}$$

The normal contact forces in the tread are computed as reaction forces associated with contact constraints from Eq. (16). Therefore, those forces are computed using Lagrange multipliers

$$\mathbf{Q}_{tre}^{nor,wi} = -(\mathbf{C}_{\mathbf{q}}^{wi,LUT})^T \boldsymbol{\lambda}, \tag{19}$$

where $\mathbf{C}_{\mathbf{q}}^{wi,LUT}$ is the LUT constraint Jacobian matrix associated with wheelsets.

In addition, the normal contact forces for the flange are included in the right-hand side of the equations of motion as the applied force from Eq. (16) and it is computed using a Hertzian model based on interpenetration and interpenetration rate:

$$\begin{aligned} \mathbf{Q}_{fla}^{nor,wi} &= \left(\frac{\partial \delta^{wi}}{\partial \mathbf{q}^{wi}} \right)^T f_{fla}^{nor,wi}, \\ f_{fla}^{nor,wi} &= K^{hertz} (\delta^{wi})^{1.5} + D^{damp} \dot{\delta}^{wi} \delta^{wi}, \end{aligned} \tag{20}$$

where K^{hertz} and D^{damp} are Hertzian contact stiffness and damping coefficients, the Hertzian stiffness is computed based on the curvatures at the wheel/rail surface and material properties [25, 27] under the assumption that the contacting bodies behave like infinite semispaces. The variables δ^{wi} and $\dot{\delta}^{wi}$ from Eq. (20) are the wheel/rail penetrations at the flange and rate at the flange contact. These are given by

$$\delta^{wi} = [\bar{\mathbf{r}}_c^{wi} - \bar{\mathbf{r}}_c^{rp}]^T \bar{\mathbf{n}}_c^{wi}, \quad \dot{\delta}^{wi} = [\dot{\bar{\mathbf{r}}}_c^{wi} - \dot{\bar{\mathbf{r}}}_c^{rp}]^T \bar{\mathbf{n}}_c^{wi}, \tag{21}$$

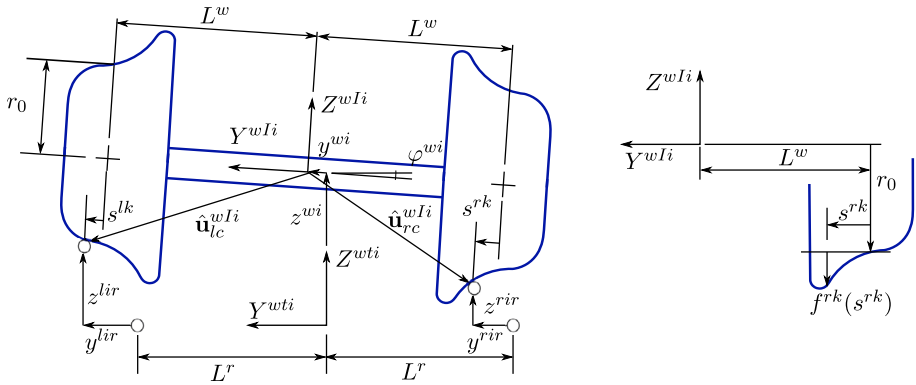


Fig. 8 Wheelset with knife-edge constraints

where $\hat{\mathbf{n}}_c^{wi}$ is the unit normal vector with respect to the wheel surface at the contact point; $\dot{\delta}^{wi}$ is not the time derivative of δ^{wi} because the time derivative of normal vector $\hat{\mathbf{n}}_c^{wi}$ has not been considered.

When the two-point contact scenario occurs, the wheel–flange contact event suddenly appears and is treated as an impact. Therefore, the elastic flange contact event may require small time step size for the time integration [28] and the flange contact stiffness controls numerical simulation performance [24].

2.4 KEC method

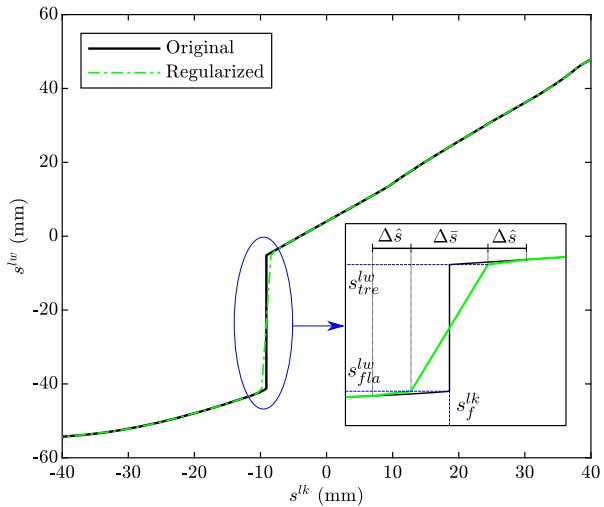
The *KEC* method is a computationally efficient online constraint-based method, in which an equivalent wheel profile that contacts an infinitely narrow rail provides the same wheelset kinematics as the real wheel/rail profiles. See Fig. 8.

Construction of *KEC*-lookup table

To use the *KEC* method, the equivalent wheel profiles must be obtained first. To this end, the results of Eq. (14) are needed to relate the location of the contact points in the *KEC* and real profiles by building a *KEC*-lookup table [16]. This algorithm is summarized as follows:

1. The track irregularities y^{lir} , z^{lir} , y^{rir} , and z^{rir} are set to zero.
2. For a one-point contact scenario, the six contact constraint equations of Eq. (12) are solved in one “for-loop” to obtain six outputs: the two wheelset coordinates and the four wheel surface parameters.
 - When two wheel/tread contacts occur at both wheels, lateral displacement y^{wi} is used as the independent coordinate to find the wheelset kinematic values z^{wi} , ϕ^{wi} , and the tread surface parameters $\mathbf{s} = [s^{lw} \ s^{lr} \ s^{rw} \ s^{rr}]^T$.
 - When one wheel/tread contact occurs at one wheel and one wheel–flange contact occurs at another one (wheel climbing), vertical displacement z^{wi} is used as an independent coordinate to find the wheelset kinematic values of y^{wi} , ϕ^{wi} , two tread surface parameters $\mathbf{s} = [s^w \ s^r]^T$ and two flange surface parameters $\mathbf{s}_{fla} = [s_{fla}^w \ s_{fla}^r]^T$.

Fig. 9 Regularization of the transverse curve parameter relation for the left equivalent and real wheels when the KEC-method is used



3. The *KEC* constraint equations are solved with different values of y^{wi} in one for-loop [16],

$$\mathbf{f}(s^k, \mathbf{f}^k) = \begin{bmatrix} y^{wi} + s^{lk} + \varphi^{wi}(r_0 + f^{lk}) - y^{lir} \\ y^{wi} + s^{rk} + \varphi^{wi}(r_0 + f^{rk}) - y^{rir} \\ z^{wi} + \varphi^{wi}(L^w + s^{lk}) - f^{lk} - r_0 - z^{lir} \\ z^{wi} + \varphi^{wi}(-L^w + s^{rk}) - f^{rk} - r_0 - z^{rir} \end{bmatrix} = \mathbf{0}, \tag{22}$$

where $s^k = [s^{lk} \ s^{rk}]^T$ and $\mathbf{f}^k = [f^{lk} \ f^{rk}]^T$, s^k and f^k are lateral positions of the contact point in the *KEC*-equivalent profiles and the value of the wheel equivalent profile for that lateral displacement (see Fig. 8 on the right), and r_0 is the rolling radius of the wheel when centered in the track. The four *KEC*-equivalent profile constraints for the wheelset given in Eq. (22) are solved to get four outputs: s^{lk} , f^{lk} and s^{rk} , f^{rk} associated with the contact points.

4. The equivalent wheel profiles (f^{lk} , f^{rk}) and the location of the contact points in the real profiles (s^{lw} , s^{rw} , s^{lr} , s^{rr}) as functions of their location in the equivalent s^{lk} , s^{rk} are stored in the *KEC*-lookup table as follows:

$$\begin{aligned} s^{lw} &= s_{klt}^{lw}(s^{lk}), & s^{lr} &= s_{klt}^{lr}(s^{lk}), & f^{lk} &= f_{klt}^{lk}(s^{lk}), \\ s^{rw} &= s_{klt}^{rw}(s^{rk}), & s^{rr} &= s_{klt}^{rr}(s^{rk}), & f^{rk} &= f_{klt}^{rk}(s^{rk}), \end{aligned} \tag{23}$$

where *klt* represents the *KEC*-lookup table. These functions are used to find the position of the contact points in the real profiles once the position in the *KEC* profile has been obtained. The surface parameter's s^{lw} location of contact points in the real left-wheel profile is plotted as a function of the lateral position s^{lk} of the contact point in the *KEC* profile in Fig. 9.

The computation of *KEC*-equivalent wheel profiles requires solving Eq. (22) including no track irregularity. The authors of [16] state that the area of allowable motion for the wheel with null irregularity cannot be guaranteed to be the same as for wheels with irregularities. Later, in [24], the *KEC*-equivalent wheel profiles were proved to be acceptable for use in irregular tracks with accuracy.

Contact detection for two-point contact scenario

As shown in Fig. 9, with a specific value of the equivalent parameter s_f^{lk} , two simultaneous contact points in the real profile s_f^{hw} and s_t^{hw} appear, which corresponds to a two-point contact scenario (before wheel climbing). However, this discontinuity in Fig. 9 might generate an unstable numerical integration problem. Therefore, the two-point contact scenario can be simulated by adopting a smoothed transition from tread to flange with the help of a regularization approach [17]. Using the transition lengths [17] of the regularization for the tread–flange transition to account for the two-point contact scenario is critical for simulation stability and accuracy. In this work, the transition lengths used are $\Delta\hat{s} = 0.5$ mm and $\Delta\bar{s} = 1$ mm. If the transition lengths are too small, the equation of motion matrix will be ill-conditioned, and the simulation will stall. Alternatively, if the transition lengths are larger, the simulation results may be less accurate.

Contact force computation for two-point contact scenario

One advantage of the *KEC* method is that contact forces on the tread and flange can be treated equally as reaction forces. The equations of motion of a railway vehicle subject to *KEC* constraints can also be written in the following DAE form:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{q}} + \mathbf{N}\lambda &= \mathbf{Q} + \mathbf{Q}^{tang} + \mathbf{Q}^{susp}, \\ \mathbf{C}^{KEC}(\mathbf{q}, t) &= \mathbf{0}, \end{aligned} \tag{24}$$

where \mathbf{C}^{KEC} includes the *KEC* constraint equations for wheelset $\mathbf{C}^{wi,KEC}$ and forward velocity holonomic constraints, the matrix \mathbf{N} includes the holonomic constraint Jacobian matrix and the matrix \mathbf{N}^{wi} , which represents the direction of the reaction forces and is associated with all the wheelsets in the vehicle. The wheelset *KEC* constraint equations $\mathbf{C}^{wi,KEC}$ in the dynamic simulation are the same as in Eq. (22). In this case, lateral displacement is treated as an independent coordinate, and the outputs of the nonlinear constraint equations from Eq. (22) become the vector of wheelset dependent coordinates $\mathbf{x}^{wi} = [z^{wi} \ \varphi^{wi} \ s^{lk} \ s^{rk}]^T$ [16].

The normal contact forces for the wheels are computed as the reaction forces as follows:

$$\mathbf{Q}^{nor,wi} = -\mathbf{N}^{wi}\lambda, \tag{25}$$

where \mathbf{N}^{wi} represents the direction of the reaction forces and torques and is associated with the wheelset. Due to the nonlinearity, the lateral position vector \mathbf{s}^k cannot be eliminated from Eq. (22). Therefore, the reaction forces for the wheelset $\mathbf{Q}^{nor,wi}$ cannot be obtained using the *KEC* constraint Jacobian matrix $\mathbf{Q}^{nor,wi} \neq -\mathbf{C}_{\mathbf{q}}^{wi,KEC}\lambda$ in Eq. (19) [17, 24].

As shown in the left and right of Fig. 10, points *A* and *B* represent the wheel/tread and wheel/flange contacts when a two-point contact scenario occurs. The transition length between points *A* and *B* in the left of Fig. 10 corresponds to the forbidden area between points *A* and *B* on the left wheel profile in the right. However, in the numerical simulation, the lateral position of point *C* in the transition length s^{lk} of the *KEC*-equivalent profile (see left of Fig. 10) might be obtained by solving Eq. (22). In this case, the contact forces will be directly applied at the contact point *C* in the forbidden area in the right of Fig. 10. Using contact force $\mathbf{Q}^{nor,wi}$ at point *C* directly, the vehicle dynamic behavior might not be accurate. Therefore, the normal contact force $\mathbf{Q}^{nor,wi}$ is transformed into two normal contact forces at the tread-end point $\mathbf{Q}_{tre}^{nor,wi}$ and the flange starting point $\mathbf{Q}_{fla}^{nor,wi}$,

$$\mathbf{Q}^{nor,wi} = \mathbf{Q}_{tre}^{nor,wi} + \mathbf{Q}_{fla}^{nor,wi}. \tag{26}$$

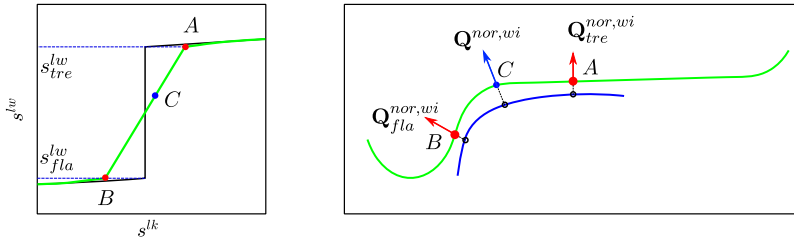


Fig. 10 (Left) Regularization of the transverse curve parameter relation for the left equivalent and real wheels; (Right) Two-point wheel/rail normal contact force computation while a regularization approach is used

2.5 Computation of tangential contact forces

In the dynamic simulation of the rail vehicle, the tangential forces and spin moment are generated between wheels and rails due to the relative motion of rolling and sliding [25]. Because it offers good computational efficiency and accuracy, the tangential contact force here was computed using Polach creep contact theory [29, 30] and treated as applied forces in the equations of motion Eqs. (16) and (24). In general, the computation of tangential contact forces requires the inputs of: (a) normal contact forces acting on the contact points, (b) relative contact velocity, (c) Kalker’s constants, and (d) coefficients of friction. In this work, the coefficient of friction was considered constant. The relative contact velocity and Kalker’s constants were computed based on the rail and wheel surface curvatures, which can be stored in a lookup table. The normal contact forces used here depends on two different situations:

1. If the normal contact force is computed as the reaction forces, such as the normal tread contact force with *LUT* method (see Eq. (19)) and normal tread or/and flange contact force with *KEC* method (see Eq. (25)), the normal forces obtained last time-step is used to find tangential contact force. This assumption can work is simply due to the small time-steps used in railway dynamic simulation.
2. If the normal contact force is computed as the applied forces, such as the normal flange contact force with *LUT* method (see Eq. (20)), current time-step normal contact forces are implemented for tangential contact force computation.

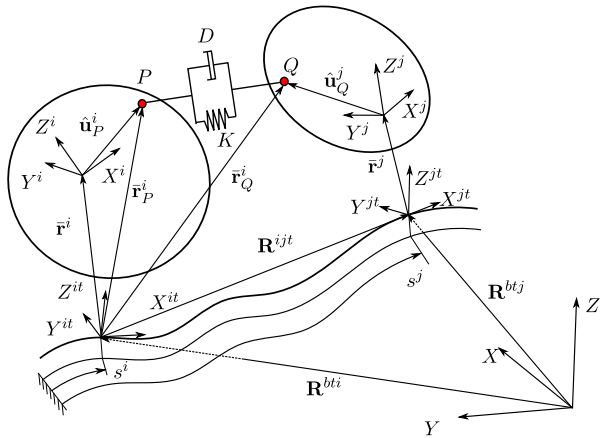
2.6 Computation of suspension forces

This work considers that each modeled body is accompanied by a track frame along the track centerline such that the body coordinates are defined with respect to this frame. See Fig. 11. However, when computing suspension forces, it is more convenient to employ a master track frame. Therefore, the relative kinematics for all moving bodies can be projected on the same track frame.

This way, the absolute position vector of a point *P* and *Q* in the global frame (see Fig. 11) are given by:

$$\begin{aligned}
 \mathbf{R}_P &= \mathbf{R}^{bti} + \mathbf{A}^{bti}(\bar{\mathbf{r}}^i + \mathbf{A}^{bti,i}\hat{\mathbf{u}}_P^i), \\
 \mathbf{R}_Q &= \mathbf{R}^{bti} + \mathbf{R}^{ijt} + \mathbf{A}^{btj}(\bar{\mathbf{r}}^j + \mathbf{A}^{btj,j}\hat{\mathbf{u}}_Q^j),
 \end{aligned}
 \tag{27}$$

Fig. 11 Kinematics projected to master track frame: \mathbf{R}^{ijt} is the relative distance vector from body track frames it to jt , $\hat{\mathbf{u}}_P^i$ and $\hat{\mathbf{u}}_Q^j$ are the position vectors of the points P and Q with respect to their body frames, $\bar{\mathbf{r}}_P^i$ and $\bar{\mathbf{r}}_Q^j$ are the position vectors of point P and Q , which is projected into master track frame i



where $\mathbf{R}^{ijt} = \mathbf{R}^{bjt} - \mathbf{R}^{bti}$ is the relative distance vector from body track frames it to jt , vectors \mathbf{R}^{bti} and \mathbf{R}^{bjt} are the position vectors of body track frames with respect to global frame, matrices \mathbf{A}^{bti} and \mathbf{A}^{bjt} are rotation matrices of body track frames, $\bar{\mathbf{r}}^i$ and $\bar{\mathbf{r}}^j$ are position vectors of body i and j with their track frames, $\hat{\mathbf{u}}_P^i$ and $\hat{\mathbf{u}}_Q^j$ are the position vectors of the points P and Q with respect to their body frames, and matrices $\mathbf{A}^{bti,i}$ and $\mathbf{A}^{btj,j}$ are the rotation matrices of body frames with respect to the body track frames. If the position vectors of points P and Q are referenced with respect to the master track frame with their components expressed with respect to this frame, Eq. (27) can be expressed as

$$\begin{aligned} \bar{\mathbf{r}}_P^i &= \bar{\mathbf{r}}^i + \mathbf{A}^{bti,i} \hat{\mathbf{u}}_P^i, \\ \bar{\mathbf{r}}_Q^j &= (\mathbf{A}^{bti})^T [\mathbf{R}^{ijt} + \mathbf{A}^{bjt} (\bar{\mathbf{r}}^j + \mathbf{A}^{btj,j} \hat{\mathbf{u}}_Q^j)]. \end{aligned} \tag{28}$$

Therefore, the suspension force of the spring–damper in parallel in Fig. 11 is computed as follows:

$$\begin{aligned} \mathbf{Q}_i^{susp} &= \left(\frac{\partial d}{\partial \mathbf{q}^i} \right)^T f^{susp}, \\ f^{susp} &= Kd + D\dot{d}, \end{aligned} \tag{29}$$

where \mathbf{q}^i is the set of generalized coordinates for body i , K is the spring stiffness, D is the damping coefficient, and the deformation length of the spring d is defined as:

$$d = \sqrt{(\bar{\mathbf{r}}_P^i - \bar{\mathbf{r}}_Q^j)^T (\bar{\mathbf{r}}_P^i - \bar{\mathbf{r}}_Q^j)} - l_0, \tag{30}$$

where l_0 is the undeformed length of the spring. The time derivative \dot{d} is expressed as [31]

$$\dot{d} = \frac{\partial d}{\partial \mathbf{q}^i} \dot{\mathbf{q}}^i. \tag{31}$$

The calculations of suspension forces with different suspension elements in the Manchester wagon 1 are given in Appendix A.

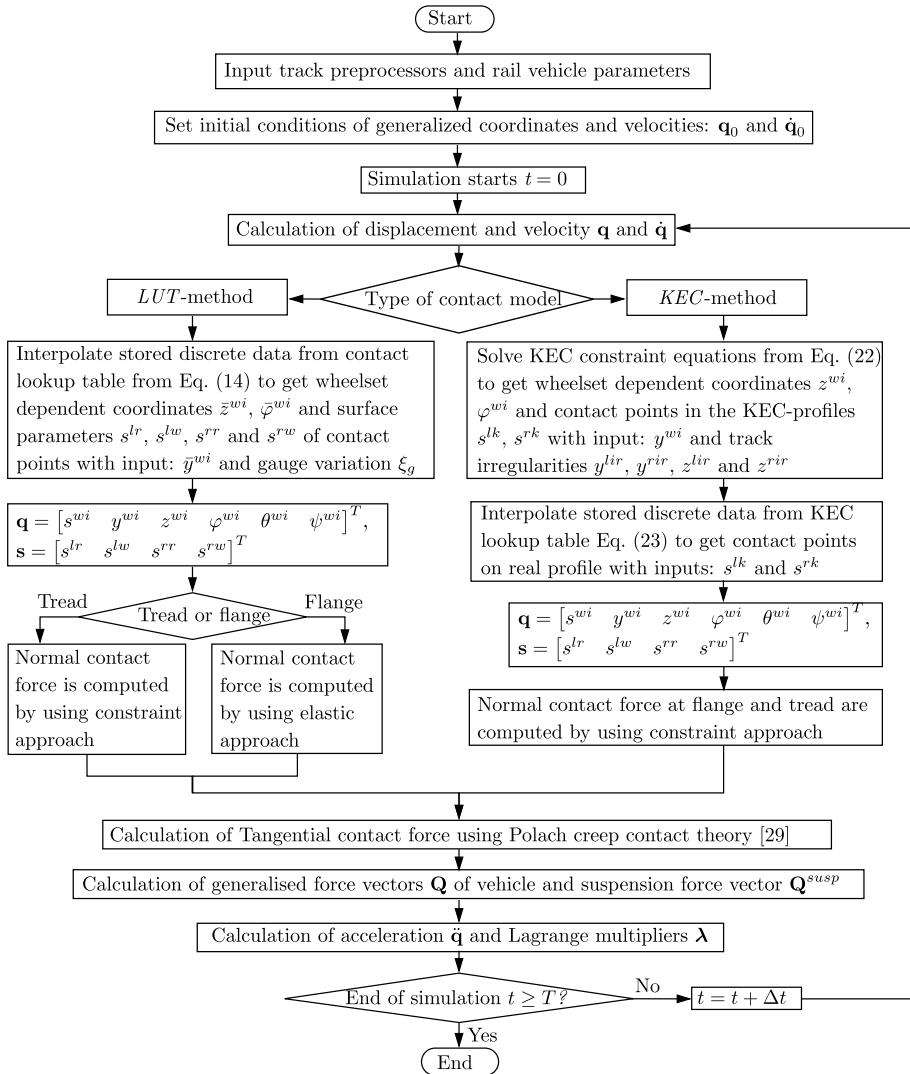


Fig. 12 Flow chart of simulation procedure

3 Computer simulation

The dynamic simulation of a railway vehicle integrates Eq. (16) or (24) forward in time. This procedure, developed in the framework of multibody dynamics, is described here as a function of the two wheel/rail contact methods used in this work and illustrated in Fig. 12 as follows:

1. Set initial conditions for the system generalized coordinates and velocities.
2. If using the contact *LUT* method, interpolate stored discrete data from the contact lookup table with two entries, $\bar{y}^{wi} = y^{wi} - \xi_a$ and ξ_g , and obtain the wheelset coordinates $\bar{z}^{wi} = z^{wi} + \xi_v$, $\bar{\varphi}^{wi} = \varphi^{wi} + \xi_c/2L_r$, and surface parameters s^{lw} , s^{lr} , s^{rw} , s^{rr} . Normal contact

forces in the tread are computed as reaction forces associated with contact constraints, while in this work, the normal contact forces for the flange are computed as a Hertzian penetration-based model.

3. If using the *KEC* method, solve the *KEC* constraint equations from Eq. (22) to get the wheelset coordinates z^{wi} , φ^{wi} , and *KEC*-profile positions s^{lk} , s^{rk} with inputs y^{wi} and track irregularities y^{lir} , y^{rir} , z^{lir} , z^{rir} . Interpolate stored discrete data from the *KEC*-lookup table to get surface parameters s^{lw} and s^{lr} with the entry of s^{lk} and s^{rw} and s^{rr} with the entry of s^{rk} . Normal contact forces on the tread and the flange are treated equally as the reaction forces. The two-point contact scenario can be simulated with a smooth transition of the normal contact forces from tread to flange.
4. Include the contact force vectors at each wheel, suspension force vectors (primary and secondary), Newton–Euler generalized forces and quadratic-velocity generalized inertia forces into the vector of external forces of the system. Solve the equations of motion of the railway vehicle to get the generalized accelerations $\ddot{\mathbf{q}}$.
5. Return the new generalized positions \mathbf{q} and velocities $\dot{\mathbf{q}}$ at time $t + \Delta t$ with the numerical integration subroutine.
6. Go to step 2 or 3 until the analysis is finished.

4 Integration methods

The numerical integration method can be divided into explicit and implicit integration. The decision to use an explicit or implicit method depends on the studied system. In the explicit method, the calculation of the variables at each time step requires the solution of the linear equations that are functions of the values of the variables in previous time steps. In the implicit method, the calculation of the variables at each time step requires the solution of the nonlinear equations that are functions of the values of the variables in the current and the previous time steps. Therefore, the implicit method uses iteration to solve nonlinear equations at each time step.

This study uses a fourth order Runge–Kutta method as the fixed-step-size explicit integrator [32]. It is a single step method, and four function evaluations are required in each integration step. As an alternative time-integration scheme, the predictor–corrector ABM method is used as the fixed-step-size in this work. It allows determining the errors in each iteration step (local truncation error) and a correction term can be included. The Adams–Bashforth–Moulton scheme can be written as follows:

$$\begin{aligned}
 \text{(Predictor) } \mathbf{y}_{n+1} = & \mathbf{y}_n + \Delta t (55\mathbf{f}(t_n, \mathbf{y}_n) - 59\mathbf{f}(t_{n-1}, \mathbf{y}_{n-1}) + 37\mathbf{f}(t_{n-2}, \mathbf{y}_{n-2}) \\
 & - 9\mathbf{f}(t_{n-3}, \mathbf{y}_{n-3}))/24,
 \end{aligned} \tag{32}$$

$$\text{(Corrector) } \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta t (9\mathbf{f}(t_{n+1}, \mathbf{y}_{n+1}) + 19\mathbf{f}(t_n, \mathbf{y}_n) - 5\mathbf{f}(t_{n-1}, \mathbf{y}_{n-1}))/24,$$

where $t_{n-0..3}$ and $\mathbf{y}_{n-0..3}$ are the three previous time steps and state variables.

The ABM is a multistep method that has two function evaluations in each integration step, the predictor and the corrector. Compared to the RK4 with four function evaluations, the ABM should be twice as fast.

The Matlab built-in function *ode45* is typically selected as the variable-step-size explicit integrator, as it can provide an acceptable solution in most cases. It is a single-step explicit integrator based on the Runge–Kutta (4,5) formula (Dormand–Prince method). For stiff systems, the solution time with *ode45* increases, and other Matlab built-in time integrators

Table 2 Simulation parameters for the vehicle

Wheelsets	Parameters	Bogies	Parameters	Car	Parameters
Mass m^{wh}	1813 kg	Mass m^b	2615 kg	Mass m^c	32000 kg
Roll inertia I_{xx}^{wh}	1120 kg·m ²	Roll inertia I_{xx}^b	1722 kg·m ²	Roll inertia I_{xx}^c	56800 kg·m ²
Pitch inertia I_{yy}^{wh}	112 kg·m ²	Pitch inertia I_{yy}^b	1476 kg·m ²	Pitch inertia I_{yy}^c	1970000 kg·m ²
Yaw inertia I_{zz}^{wh}	1120 kg·m ²	Yaw inertia I_{zz}^b	3067 kg·m ²	Yaw inertia I_{zz}^c	1970000 kg·m ²

schemes such as *ode15s* can perform better; *ode15s* is a variable order and variable-step-size implicit solver based on numerical differentiation formulas (NDFs) of orders 1 to 5. Optionally, the maximum order (5) can be reduced or a backward difference formulation can be used [33].

The railway vehicle models usually contain stiff springs for suspension components or contact forces modeling. The explicit integration methods, such as RK4, are not suitable for a stiff problem. In many cases, the explicit integration method is conditionally stable and it need least effort for one time integration step. However, the method requires a small time-step size to solve the stiff problem and makes the simulation less efficient [34]. Furthermore, the predictor–corrector ABM method includes the previous step information to prevent the numerical instability. The Matlab *ode45* is based on Runge–Kutta algorithm and has similar features as the RK4 for a stiff problem. The Matlab *ode15s* is based on NDF formulation, where the time-step and solution order can be changed. These features are desired in the stiff problem solution, as dynamically adjusting these parameters allows obtaining the solution efficiently. However, it requires the iterative solution for the nonlinear equations and the update of the Jacobian matrix at each time step. As the number of unknowns gets larger, the numerical effort for the update of Jacobian matrices starts to be more expensive [23].

5 Simulation results

5.1 Vehicle model description

This section describes the vehicle model used in the numerical simulations of Sect. 3, which is based on the Manchester wagon used in Iwnick [3] as a benchmark. Figure 13 illustrates the various components of the vehicle formed by one carbody, two bogie frames and four wheelsets with primary and secondary suspension elements.

The mass and inertia properties of the different bodies are presented in Table 2. The wheel and rail profiles used for the contact lookup table and computation of the *KEC* profiles are the S1002 wheel and LB140-Area rail profiles shown in Fig. 14, which present a unique two-point contact scenario in the tread–flange transition. Wheel and rail profile positioning with respect to the track centerline (L_w , L_r), rail inclination (β) and wheel nominal radius (r_0) are given in Table 1. The coefficient of friction for the wheel rail tangential contact force is $\mu = 0.2364$ and the Hertzian stiffness and damping constant for flange contact with the *LUT* method are listed in Table 3.

The primary and secondary suspension shown in Fig. 13 involves a total of 48 elements classified into springs and dampers in series, spring and dampers in parallel, and bumpstops, with their stiffness and damping properties given in Table 4.

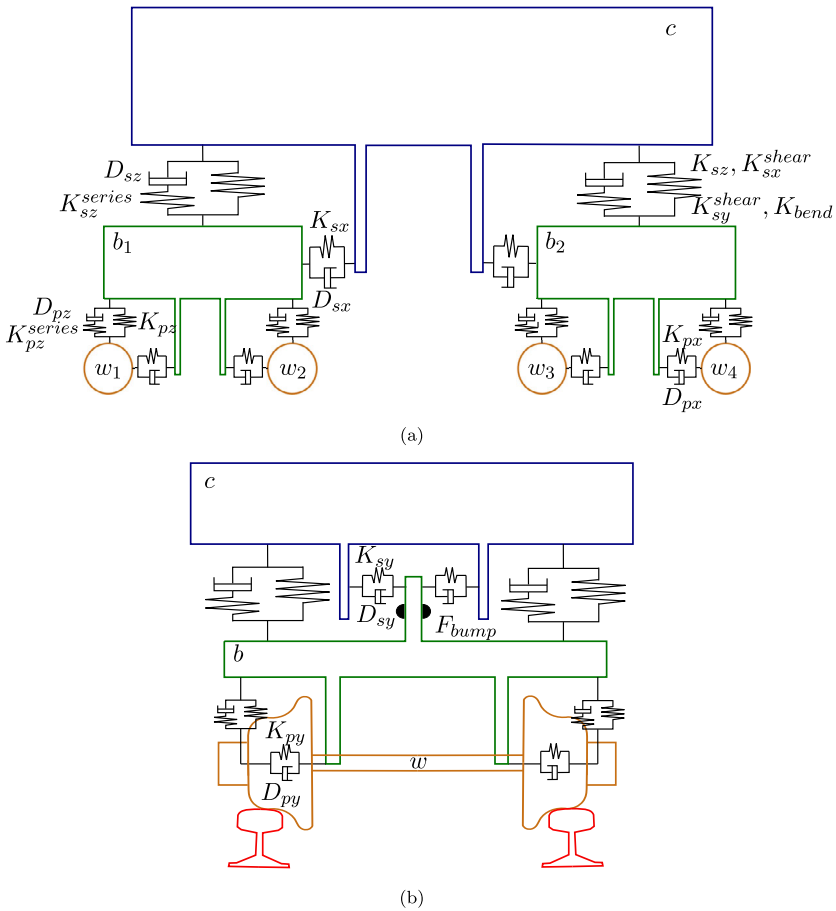


Fig. 13 Three-dimensional Manchester wagon: (a) side view; (b) front view

Table 3 Hertzian stiffness and damping constant for flange contact with *LUT* method

Model	Parameters				
Stiffness K^{hertz} (N/m ^{1.5})	$7.075 \cdot 10^{13}$	$1 \cdot 10^{12}$	$1 \cdot 10^{11}$	$1 \cdot 10^{10}$	$1 \cdot 10^9$
Damping D^{damp} (N·s/m ²)	$7.075 \cdot 10^{11}$	$1 \cdot 10^{10}$	$1 \cdot 10^9$	$1 \cdot 10^8$	$1 \cdot 10^7$

Track irregularities are generated using analytical expressions of the power spectral density functions (PSD) [35]. The alignment, vertical profile, gauge variation, and cross level are shown in Fig. 15.

5.2 Validation

In this section, the 3D multibody model of the railway vehicle is implemented in commercial simulation software, Universal Mechanism (UM) [36] to validate that implemented in Matlab software. For both the *KEC* and *LUT* methods in Matlab software and UM commer-

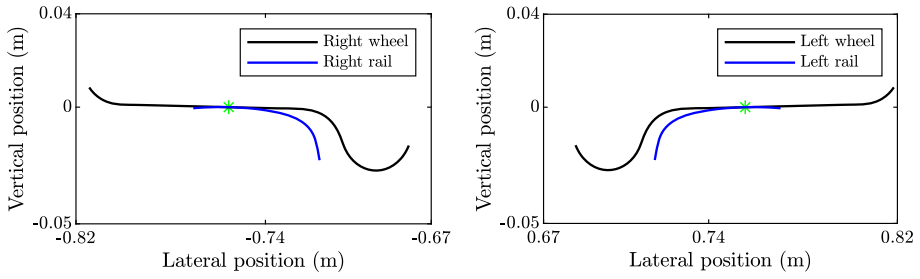


Fig. 14 Real wheel/rail profile combination that shows two-point contact scenario and LB-140-area rail profile with S1002 wheel profile

Table 4 Main parameters of suspension elements in the simulation

Suspension	Parameter	Number of elements
Secondary vertical stiffness K_{sz}	430 kN/m	2/bogie
Secondary longitudinal shear stiffness K_{sx}^{shear}	160 kN/m	2/bogie
Secondary lateral shear stiffness K_{sy}^{shear}	160 kN/m	2/bogie
Secondary bending stiffness K_{bend}	10.5 kN/rad	2/bogie
Secondary vertical damper rate D_{sz}	4 kN·s/m	2/bogie
Secondary vertical damper series stiffness K_{sz}^{series}	1000 kN/m	2/bogie
Secondary longitudinal stiffness K_{sx}	5000 kN/m	1/bogie
Secondary longitudinal damper rate D_{sx}	25 kN·s/m	1/bogie
Secondary lateral stiffness K_{sy}	6000 kN/m	2/bogie
Secondary lateral damper rate D_{sy}	32 kN·s/m	2/bogie
Primary vertical stiffness K_{pz}	1220 kN/m	4/bogie
Primary vertical damper rate D_{pz}	4 kN·s/m	4/bogie
Primary vertical damper series stiffness K_{pz}^{series}	1000 kN/m	4/bogie
Primary longitudinal stiffness K_{px}	31391 kN/m	4/bogie
Primary longitudinal damper rate D_{px}	15 kN·s/m	4/bogie
Primary lateral stiffness K_{py}	3884 kN/m	4/bogie
Primary lateral damper rate D_{py}	2 kN·s/m	4/bogie
Bumpstop	Δl_{bump}	0 25 30 35 40 45 50 55 60 65/mm
	F_{bump}	0 0 0.6 1.76 3.73 6.87 11.58 17.17 29.2 230.0/kN

cial simulation software, the fixed-step-size integrator ABM with time step size $\Delta t = 1$ ms is used for all approaches. In addition, the Hertzian contact stiffness at the flange for the LUT method is $K^{hertz} = 1 \cdot 10^{10}$ N/m^{1.5}. A numerical comparison of three different case studies is presented: (1) single wheelset on a curved track without track irregularities, (2) the Manchester vehicle on a curved track without track irregularities, and (3) the Manchester vehicle on a curved track with track irregularities.

Single wheelset on a curved track without track irregularities

In the first case, the single wheelset is simulated at a constant forward velocity of $V = 10$ m/s (36 km/h) on a 120 m track without irregularities constructed of the following three

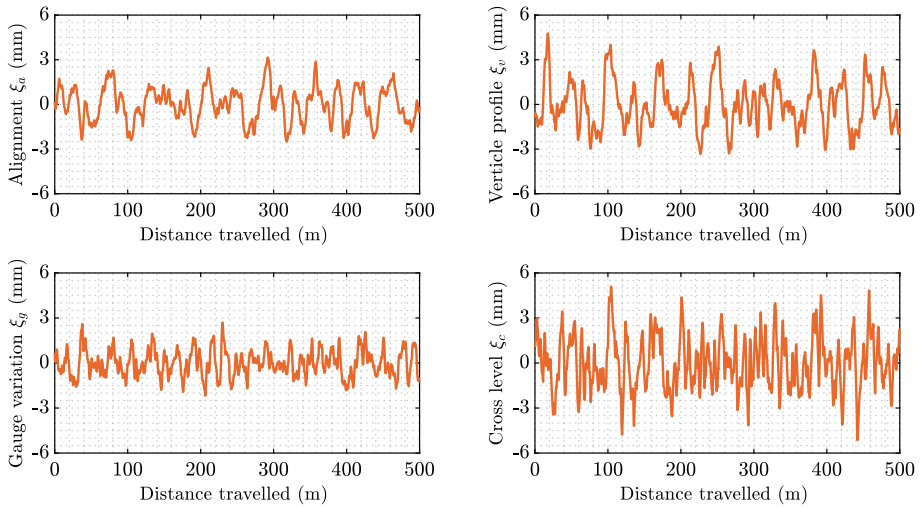


Fig. 15 Track irregularities

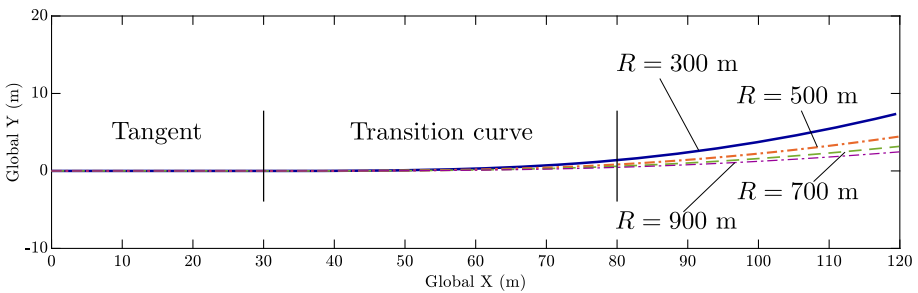


Fig. 16 Track geometries with 120 m traveled distance. Lines perpendicular to the track geometry show the transition points of the track segments

segments: a 30 m tangent, a 50 m transition, and a 40 m left curve segment with different values of curve radius as shown in Fig. 16.

Figure 17 shows the comparison of the lateral displacement and yaw angle of the wheelset with the *KEC* and *LUT* methods and *UM*. The achievement of the steady curving can be observed when the vehicle enters into transition curve (see grey area). In addition, with different values of curve radius, the traveled distance where steady curving starts is different. The steady curving results are quantified in Table 5. The results of lateral displacement and yaw angle show good agreement between both of the examined methods and *UM*, except for the yaw angle, which is about 2 mrad bigger when using *UM* on the radius $R = 300$ m curve track. That is mainly because the planar contact approach is implemented into the *KEC* and *LUT* methods, but in *UM*, the 3D elastic contact approach is utilized for the tread and flange contacts.

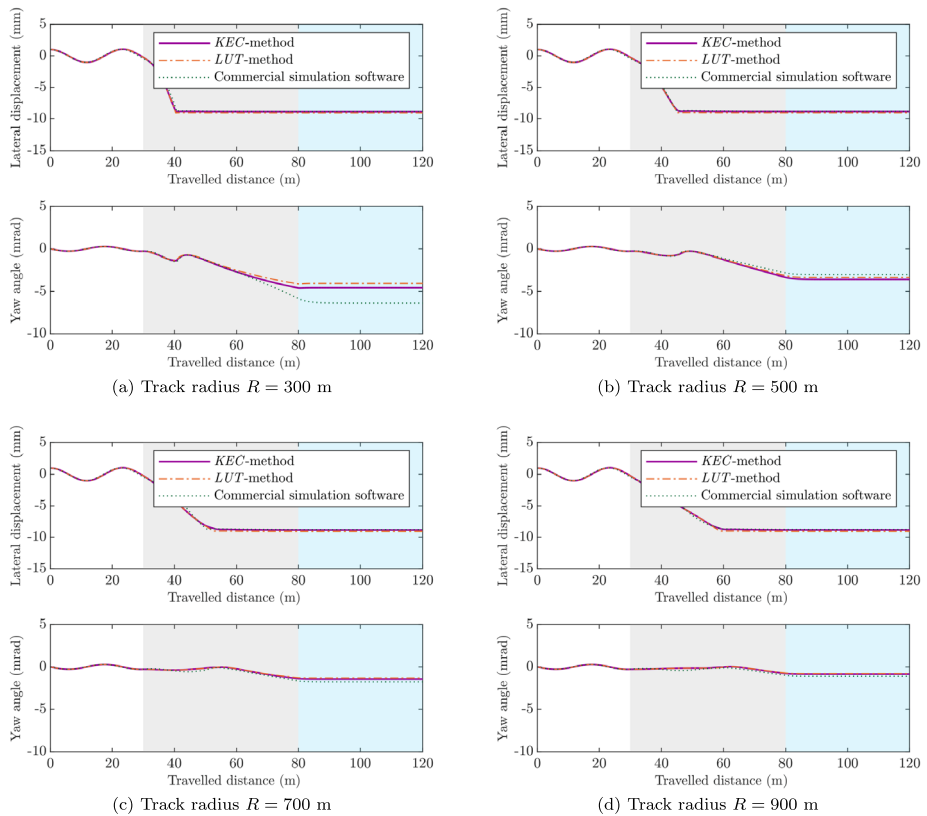


Fig. 17 Comparison of the lateral displacement and yaw angle of the single wheelset vehicle as it negotiates different radius tracks with constant forward velocity $V = 10$ m/s. The white area represents the tangent segment, the grey area represents the transition curve segment, and the blue area represents the left curve segment

Table 5 Vehicle coordinates during steady curving, when a single wheelset vehicle negotiates different radius tracks with constant forward velocity $V = 10$ m/s

	Radius $R = 300$ m		Radius $R = 500$ m		Radius $R = 700$ m		Radius $R = 900$ m	
	y^{wh} (mm)	ψ^{wh} (mrad)	y^{wh} (mm)	ψ^{wh} (mrad)	y^{wh} (mm)	ψ^{wh} (mrad)	y^{wh} (mm)	ψ^{wh} (mrad)
KEC	-8.8569	-4.5566	-8.8303	-3.5689	-8.8476	-1.4441	-8.8374	-0.8383
LUT	-9.0468	-4.0368	-9.0324	-3.3389	-9.0401	-1.3149	-9.0353	-0.7775
UM	-8.8266	-6.3745	-8.7994	-3.0023	-8.7829	-1.7665	-8.7538	-1.0950

Manchester wagon on a curved track without track irregularities

In the second case, the Manchester wagon is simulated at a constant forward velocity of $V = 20$ m/s (72 km/h), on a 500 m track without irregularities formed by the following three segments: a 100 m tangent, a 50 m transition, and a 350 m left curve of $R = 500$ m radius segment (see Fig. 18).

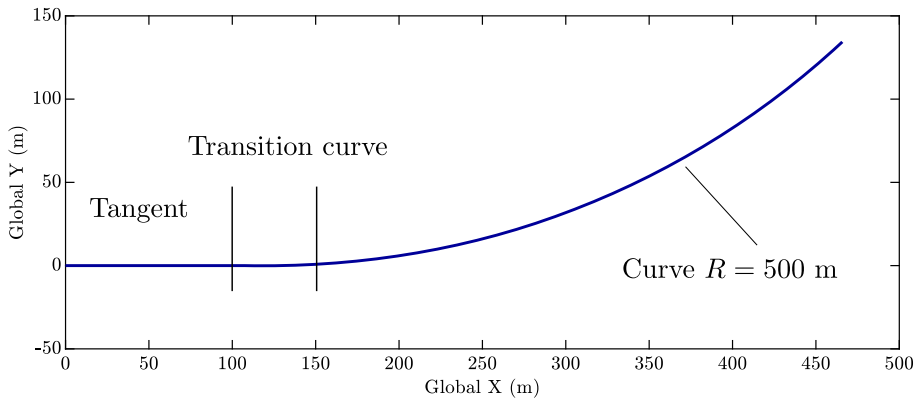


Fig. 18 Track geometries on the 500 m traveled distance with an $R = 500$ m radius curved track

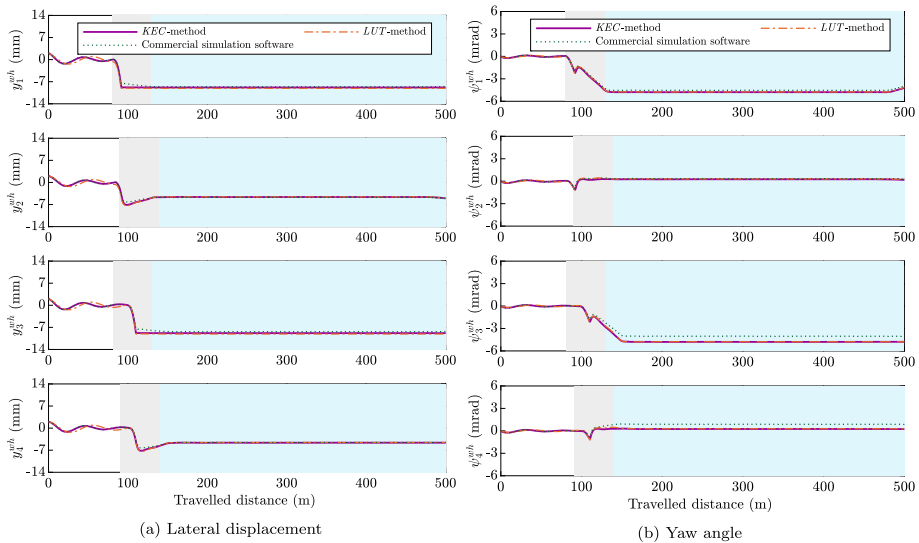


Fig. 19 Comparison of kinematics of four wheelsets when the vehicle negotiates the $R = 500$ m radius track without track irregularities with constant forward velocity $V = 20$ m/s

Figures 19 and 20 show a comparison of lateral displacements and yaw angles for the vehicle wheelsets, bogies, and car body, when using the *KEC* and *LUT* methods and UM. Table 6 lists the quantities of the steady curving results. Again, the results of lateral displacement and yaw angle show good agreement. A steady curving motion is achieved when the vehicle enters into the transition curve. According to the lateral displacement of the four wheelsets, the front wheelsets of both bogies can experience flange contact when negotiating the curve (blue area in the figure), and the rear wheelsets do not.

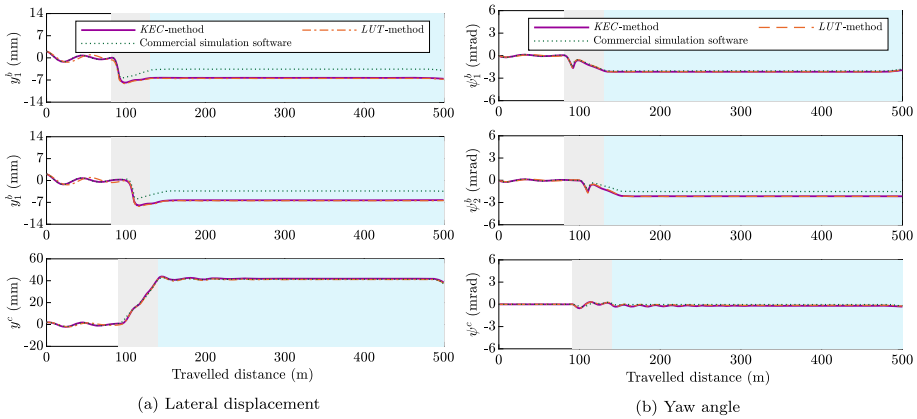


Fig. 20 Comparison of kinematics of bogies and car body – the information is the same as that shown by Fig. 19

Table 6 Vehicle coordinates during steady curving when a Manchester wagon negotiates an $R = 500$ m radius curved track without irregularities with constant forward velocity $V = 20$ m/s

		Front wh., front bog.	Rear wh., front bog.	Front wh. rear bog.	Rear wh., rear bog.	Front bog.	Rear bogie	Car
y^{wh} (mm)	KEC	-8.8291	-4.6144	-8.8294	-4.6242	-6.3153	-6.3191	41.8464
	LUT	-9.0966	-4.5718	-9.0970	-4.5469	-6.4958	-6.4676	41.0944
	UM	-8.6057	-4.7402	-8.3450	-4.5669	-3.5763	-3.3553	41.3972
ψ^{wh} (mrad)	KEC	-4.7822	0.2586	-4.7911	0.2495	-2.1481	-2.1570	-0.1997
	LUT	-4.7474	0.3042	-4.7752	0.2799	-2.1474	-2.1730	-0.1200
	UM	-4.5372	0.3430	-4.0380	0.8763	-2.0372	-1.5205	-0.0276

Manchester wagon on a curved track with track irregularities

In the last case, the Manchester wagon is simulated at a constant forward velocity of $V = 20$ m/s (72 km/h) on a 500 m track (see Fig. 18) with irregularities. Figures 21 and 22 show a comparison of the lateral displacement and yaw angle for the vehicle wheelsets, bogies, and car body when using the KEC and LUT methods and UM. The results of lateral displacement and yaw angle are almost identical with the results on the irregular track. According to the lateral displacement of the two wheelsets, steady curving motion is not achieved because of the track irregularities.

5.3 Computational efficiency

In this case, the Manchester wagon is simulated at a constant forward velocity of $V = 20$ m/s (72 km/h), on the same 500 m track as shown in Fig. 18 with irregularities. The computational efficiency of the LUT method is compared to that of the KEC method using different integration methods. The comparison was carried out on an Intel Core i5 laptop with a 2.5 GHz CPU and Matlab 2018b. The Mex functions written in C or Fortran were not used to improve computational efficiency in Matlab. To determine the computational time,

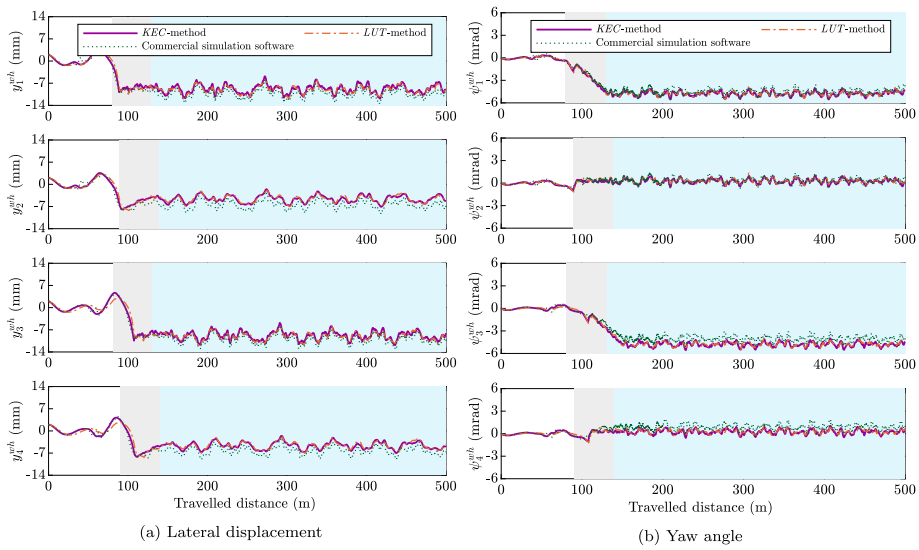


Fig. 21 Comparison of kinematics of four wheelsets when the vehicle negotiates the $R = 500$ m radius track with track irregularities and constant forward velocity $V = 20$ m/s

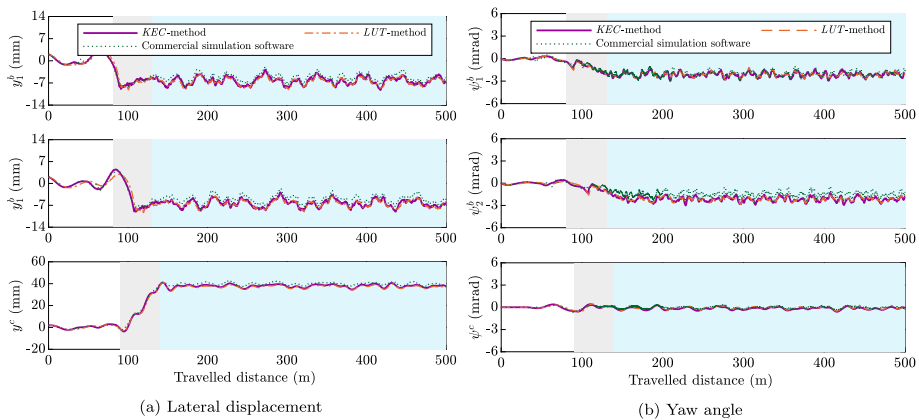


Fig. 22 Comparison of kinematics of bogies and car body, the information is same as in Fig. 21

each case is simulated five times, and the calculated simulation time was as the average value of the five runs. The computation times for the five runs for each case are listed in Appendix B. Two fixed-step-size integrators, explicit RK4 and predictor–corrector ABM, and two variable-step-size Matlab built-in function integrators, explicit *ode45* and implicit *ode15s*, were applied to compare computational efficiency.

According to the Hertz contact theory [25], the flange Hertzian stiffness for the wheel–rail profile combination shown in Fig. 14 is $K^{hertz} = 7.075 \cdot 10^{13} \text{ N/m}^{1.5}$ with the Poisson’s ratio $\nu = 0.28$ and Young’s modulus $E = 2.1 \cdot 10^{11} \text{ N/m}^2$. Since the location of the flange contact points remains the same when the two-point contact scenario occurs, the constant flange contact stiffness parameters are considered for the hybrid method in the con-

Table 7 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on a 500 m track using *LUT* method with different integrators – the Hertzian stiffness at the flange for *LUT* method is $7.075 \cdot 10^{13}$ N/m^{1.5}

Integrators	Tol	<i>LUT</i> method	
		CPU time (s)	Func- eval-
<i>ode15s</i>	$1 \cdot 10^{-7}$	11192.74	1125730
<i>ode45</i>	$1 \cdot 10^{-8}$	16909.05	2171100
RK4		–	–
ABM		–	–

tact lookup table. Computational costs and function evaluations of the vehicle simulation with flange Hertzian stiffness $K^{hertz} = 7.075 \cdot 10^{13}$ N/m^{1.5} for the *LUT* method are listed in Table 7. With step size $\Delta t = 1$ ms, RK4 and ABM failed to converge at same time as when flange occurs. In addition, the variable-step-size explicit integrator *ode45* only succeeds when the absolute and relative tolerance is smaller than $1 \cdot 10^{-8}$ and the variable-step-size implicit integrator *ode15s* only succeeds when the absolute and relative tolerance is smaller than $1 \cdot 10^{-7}$. This is because having the high-magnitude Hertzian parameter, $K^{hertz} = 7.075 \cdot 10^{13}$ N/m^{1.5}, results in a stiff system of ordinary differential equations, and some variations of the equation significantly impact the solution. If the time step and/or tolerances are too large, the first impact between the wheel flange and rail head may result in a substantial penetration depth, which leads to a numerical instability and failure of the time integration. Therefore, a large number of time steps are needed to solve the problem, which results in high computational cost.

Under the assumption that the contacting bodies behave like infinite semispaces, the Hertzian stiffness of $K^{hertz} = 7.075 \cdot 10^{13}$ N/m^{1.5} might be higher than the real value. In fact, reducing the value of Hertzian stiffness is realistic because the influence of flange flexibility is not considered. In what follows, different values of flange contact stiffness will be presented. Computational cost and function evaluations of the vehicle simulation with different values of flange Hertzian stiffness for *LUT* method are compared against *KEC* method in Fig. 23. The computation time of five runs for each case are listed in Tables 12, 13, 14, 15, 16. When using variable-step-size integrators, *ode15s* and *ode45*, the *KEC* method is much superior to the *LUT* method with higher flange contact stiffness (such as $K^{hertz} = 1 \cdot 10^{11}$ N/m^{1.5} and $1 \cdot 10^{12}$ N/m^{1.5}). This is due to the large value of stiffness that leads to a stiff system of ordinary differential equations and increases the computational effort during the simulation. It is true that stiff integrators (*ode15s*) are more expensive. But in case of stiff systems, they lead to a stable solution that is not sometimes possible or take even more time in the case of nonstiff integrators (ABM and RK4). When working with lower flange stiffness $K^{hertz} = 1 \cdot 10^{10}$ N/m^{1.5} and $1 \cdot 10^9$ N/m^{1.5}, the *LUT* method shows better performance in computational efficiency.

5.4 Accuracy

Table 8 lists a comparison of maximum and average errors of lateral displacement between *LUT* and *KEC* methods when using different integrators, with the Manchester wagon simulated at a constant forward velocity of $V = 20$ m/s (72 km/h) on a 500 m track with irregularities. See Fig. 18. Different values of flange stiffness are considered at the wheel flange. The fixed step sizes are $\Delta t = 1$ ms for fixed-step-size integrators RK4 and ABM. The maximum step size is $\Delta t = 1$ ms for Matlab integrators *ode45* and *ode15s*. The absolute and

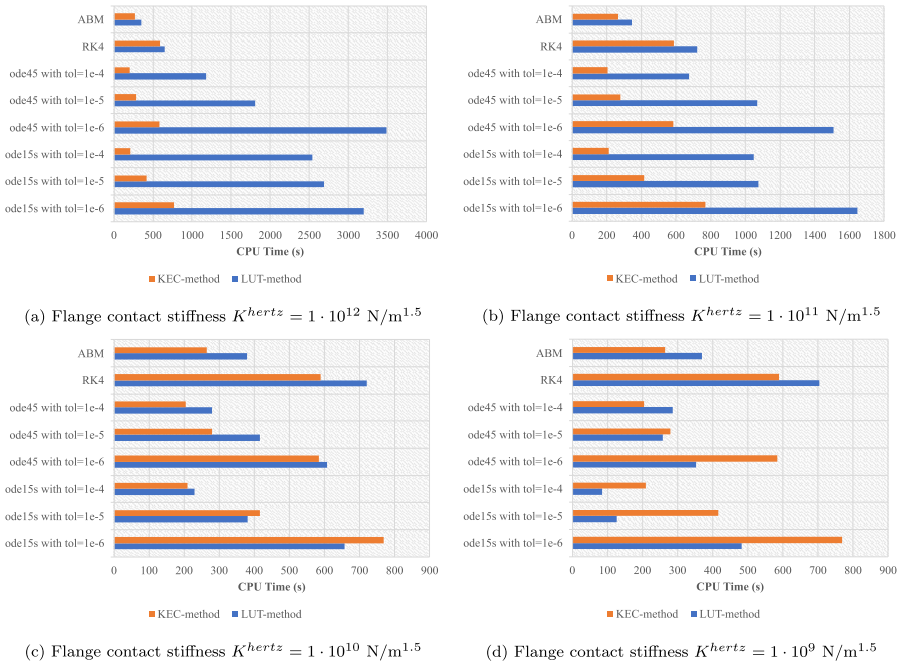


Fig. 23 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on a 500 m track by using both methods with different integrators

relative tolerance for *ode45* and *ode15s* is $1 \cdot 10^{-6}$. The error between both methods is computed as

$$Error = \frac{|y_{KEC}^{wh} - y_{LUT}^{wh}|}{\max(|y_{LUT}^{wh}|)}, \tag{33}$$

where y_{KEC}^{wh} is the lateral displacement for the *KEC* method within a 25 s simulation, and y_{LUT}^{wh} is for the *LUT* method. Some conclusions can be made based on the numerical results from Table 8:

- For all integrators, the lower flange stiffness for the *LUT* method (especially $K^{hertz} = 1 \cdot 10^9$ N/m^{1.5}) results in higher maximum and average absolute errors. This is due to the low flange stiffness that results in large flange to rail head indentations.
- Compared to the other integrators, the *ABM* errors are much higher when selecting a large value of the flange contact stiffness $K^{hertz} = 1 \cdot 10^{12}$ N/m^{1.5}. This is because constant time-step integrators are not appropriate for impact problems. As shown in Fig. 24, the flange indentation and normal contact forces become artificially large for the *ABM* integrator.

As shown in Fig. 25, the use of a low stiffness ($K^{hertz} = 1 \cdot 10^9$ N/m^{1.5}) helps to simulate the vehicle motion. However, low stiffness results in high indentations (around 0.75 mm) that can be considered as physically inadmissible. Even though the flange indentations are different for different flange contact stiffnesses K^{hertz} in Fig. 25, the resulting flange normal contact forces F_{hertz} are on average almost the same. However, the higher the stiffness, the higher the force oscillations.

Table 8 Maximum and average errors of lateral displacement between the *LUT* and *KEC* methods

Hertzian stiffness (N/m ^{1.5})	Integrators	Error	Front wh., Front bogie (%)	Rear wh., Front bogie (%)	Front wh., Rear bogie (%)	Rear wh., Rear bogie (%)	Front bogie (%)	Rear bogie (%)	Car (%)	
1 · 10 ¹²	<i>ode15s</i>	Max.	13.79	26.99	13.88	18.03	16.79	14.17	3.08	
		Ave.	3.48	7.06	3.28	4.63	4.24	3.47	0.92	
	<i>ode45</i>	Max.	13.76	27.01	13.95	18.05	16.81	14.18	3.08	
		Ave.	3.48	7.06	3.27	4.62	4.24	3.47	0.93	
	RK4	Max.	13.78	27.22	13.84	18.07	16.92	14.19	3.09	
		Ave.	3.48	7.10	3.27	4.63	4.26	3.47	0.92	
	ABM	Max.	34.78	26.20	47.54	30.77	29.02	29.34	3.53	
		Ave.	7.52	7.32	8.91	10.13	6.88	8.58	0.93	
	1 · 10 ¹¹	<i>ode15s</i>	Max.	13.92	27.37	14.11	18.05	17.09	14.25	3.14
			Ave.	3.47	7.28	3.27	4.69	4.32	3.48	0.94
<i>ode45</i>		Max.	13.87	27.37	14.12	18.06	17.08	14.24	3.14	
		Ave.	3.47	7.28	3.27	4.69	4.32	3.48	0.94	
RK4		Max.	13.94	27.36	14.07	18.09	17.09	14.28	3.14	
		Ave.	3.47	7.28	3.27	4.70	4.32	3.48	0.94	
ABM		Max.	13.94	27.44	14.14	18.15	17.13	14.30	3.13	
		Ave.	3.47	7.29	3.27	4.71	4.32	3.49	0.94	
1 · 10 ¹⁰		<i>ode15s</i>	Max.	14.28	29.27	14.30	20.05	18.62	15.85	3.29
			Ave.	3.50	8.34	3.26	5.54	4.85	3.82	0.95
	<i>ode45</i>	Max.	14.28	29.27	14.29	20.05	18.62	15.85	3.29	
		Ave.	3.50	8.34	3.26	5.54	4.85	3.82	0.95	
	RK4	Max.	14.28	29.27	14.29	20.05	18.62	15.85	3.29	
		Ave.	3.50	8.34	3.26	5.54	4.85	3.82	0.95	
	ABM	Max.	14.28	29.27	14.29	20.05	18.62	15.85	3.29	
		Ave.	3.50	8.34	3.26	5.54	4.85	3.82	0.95	
	1 · 10 ⁹	<i>ode15s</i>	Max.	20.24	36.82	20.48	27.40	24.85	22.06	4.06
			Ave.	5.58	13.90	5.45	10.30	8.67	7.01	1.15
<i>ode45</i>		Max.	20.24	36.82	20.49	27.40	24.85	22.06	4.06	
		Ave.	5.58	13.90	5.45	10.30	8.67	7.01	1.15	
RK4		Max.	20.24	36.82	20.49	27.40	24.85	22.06	4.06	
		Ave.	5.58	13.90	5.45	10.30	8.67	7.01	1.15	
ABM		Max.	20.24	36.82	20.49	27.40	24.85	22.06	4.06	
		Ave.	5.58	13.90	5.45	10.30	8.67	7.01	1.15	

5.5 The effect of the refinement for the *KEC* and *LUT* tables on the computational efficiency

The use of lookup table for both *LUT* and *KEC* approaches requires generating a pre-calculated table which tabulates the values of the different parameters and variables into

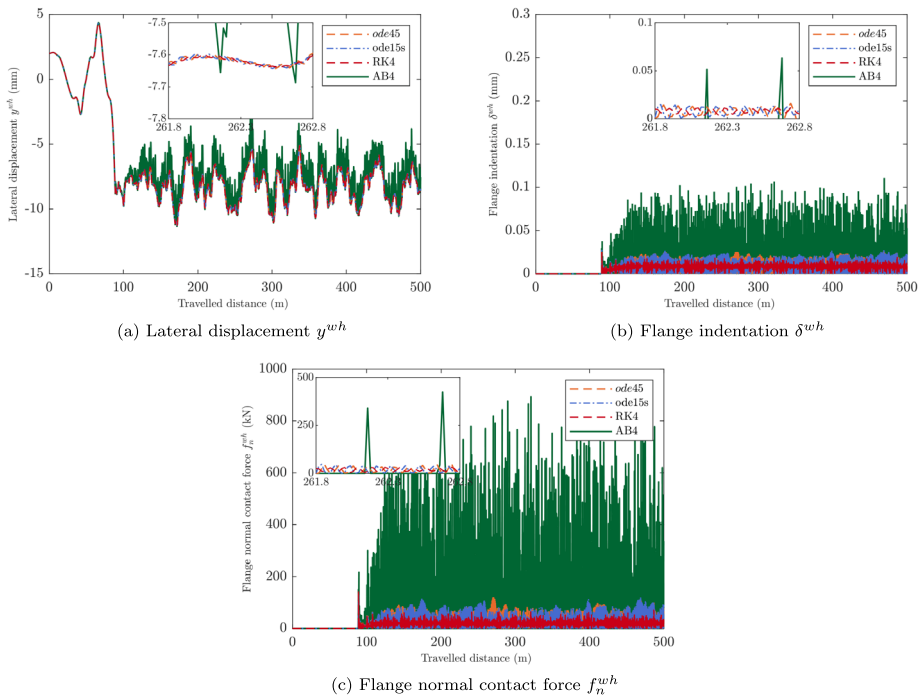


Fig. 24 Comparison of lateral displacement, flange indentation, and normal contact force with the *LUT* method when using ABM, RK4, *ode45* and *ode15s*. The fixed step sizes are $\Delta t = 1$ ms for fixed-step-size integrators RK4 and ABM. The maximum step size is $\Delta t = 1$ ms for the Matlab integrators *ode45* and *ode15s*. The absolute and relative tolerance for *ode45* and *ode15s* is $1 \cdot 10^{-6}$. Flange contact stiffness $K_{hertz} = 1 \cdot 10^{12}$ N/m^{1.5}

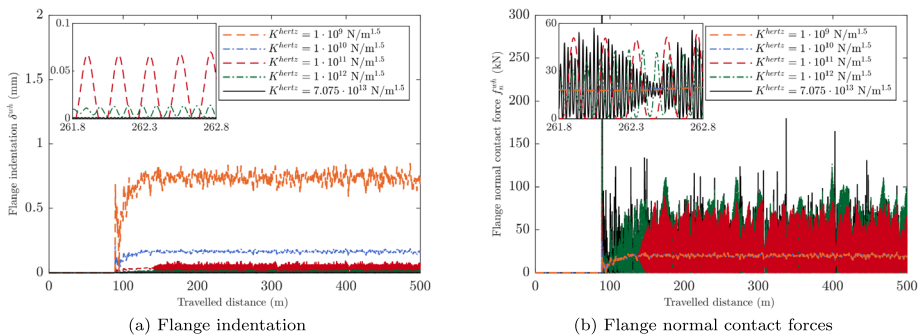


Fig. 25 Comparison of flange indentation and normal contact force with *LUT* method when using ABM with step size $\Delta t = 0.1$ ms. Different flange contact stiffnesses are considered

a set of nodal points. Later in the dynamic simulations, the required parameters or variables for the dynamic analysis are obtained by using linear interpolation between the stored data. The *KEC* table has only one entry, the lateral wheel parameter s^{lk} or s^{rk} . In addition, the *LUT* table has two entries, gauge variation irregularity ξ_g and wheelset lateral displacement y^w . In this section, each *KEC* table is refined by increasing the number of

Table 9 Notes of lookup table information for *KEC* tables before and after refinement

Before refinement	After refinement
<ul style="list-style-type: none"> • For each <i>KEC</i> table shown in Eq. (23), wheel lateral parameters s^{lk} and s^{rk} are discretized from -46.4 mm to $+46.4$ mm with constant steps $\Delta s^{lk} = 0.2$ mm and $\Delta s^{rk} = 0.2$ mm. • For each <i>KEC</i> table, the matrix size is 465×1 with Matlab storage of 4 kB each. 	<ul style="list-style-type: none"> • For each <i>KEC</i> table shown in Eq. (23), the wheel lateral parameters s^{lk} and s^{rk} are discretized from -46.4 mm to $+46.4$ mm with constant step $\Delta s^{lk} = 0.018$ mm and $\Delta s^{rk} = 0.018$ mm. • For each <i>KEC</i> table, the matrix size is 5001×1 with Matlab storage of 12 kB.

Table 10 Notes of lookup table information for both *LUT* and *KEC* tables

Before refinement	After refinement
<ul style="list-style-type: none"> • For each <i>LUT</i> table shown in Eq. (14), the gauge variation ξ_g is discretized from -5 mm to $+5$ mm with constant step $\Delta \xi_g = 1$ mm, and the wheelset lateral displacement y^{wi} is discretized from -18 mm to $+18$ mm with constant step $\Delta y^{wi} = 0.1$ mm. • For each <i>LUT</i> table, the matrix size is 361×11 with Matlab storage of 25 kB each. 	<ul style="list-style-type: none"> • For each <i>LUT</i> table shown in Eq. (14), the gauge variation ξ_g is discretized from -5 mm to $+5$ mm with constant step $\Delta \xi_g = 1$ mm, and the wheelset lateral displacement y^{wi} is discretized from -18 mm to $+18$ mm with constant step $\Delta y^{wi} = 0.05$ mm. • For each <i>LUT</i> table, the matrix size is 721×11 with Matlab storage of 46 kB.

points from 465×1 to 5001×1 as listed in Table 9. Furthermore, each *LUT* table is refined by increasing the number of points from 361×11 to 721×11 as listed in Table 10.

The refinement of lookup tables tends to increase the precalculation time and the computer storage while it is assumed to improve the accuracy of the interpolated values [15]. To verify if the refinement of lookup tables can improve the computational efficiency, Fig. 26 compares the computation cost by using *KEC* and *LUT* methods with the flange Hertzian stiffness $1 \cdot 10^{10}$ N/m^{1.5}. The computation time of five runs for each refinement case are listed in Tables 17–18. The results indicate that the refinement of both *KEC* and *LUT* tables can slightly improve the computational efficiency in most particular cases (14 out of 18).

Table 11 lists the computational costs and function evaluations of the vehicle simulation with flange Hertzian stiffness $K^{hertz} = 7.075 \cdot 10^{13}$ N/m^{1.5} for the *LUT* method after refinement. Compared to the result before refinement from Table 7, it can be concluded that the refinement only helps *ode15s* and *ode45* to smooth the numerical integration that increases the maximum tolerance to 10 times bigger.

6 Conclusion

Two constraint-based formulations for the wheel/rail contact simulation were studied in terms of their accuracy and computational efficiency based on the Manchester wagon 1 model. Two fixed-step-size integrators, RK4 and predictor–corrector-based ABM, and two

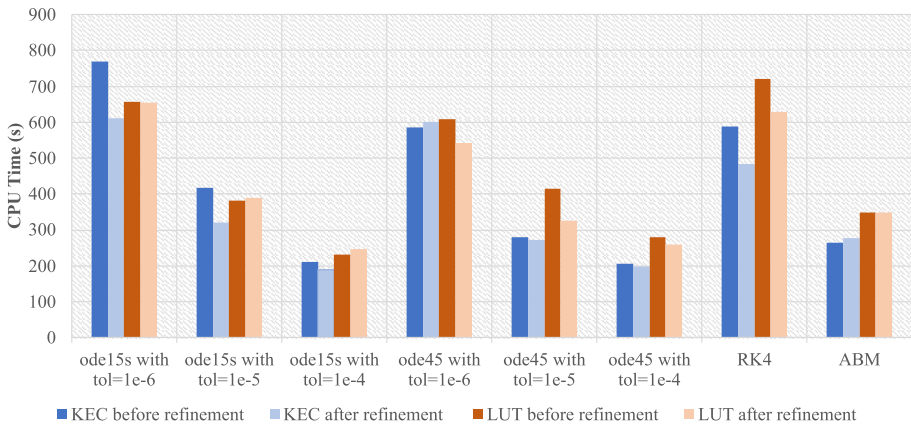


Fig. 26 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using both *KEC* and *LUT* methods with and without refinement

Table 11 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *LUT* method after refinement, the Hertzian stiffness at flange is $7.075 \cdot 10^{13}$ N/m^{1.5}

Integrators	Tol	<i>LUT</i> method	
		CPU time (s)	Func- eval-
<i>ode15s</i>	10^{-6}	9316.95 s	100394
<i>ode45</i>	10^{-7}	11137.77 s	1354320
RK4		–	–
ABM		–	–

variable-step-size Matlab integrators, *ode15s* and *ode45*, were used to carry out the comparison.

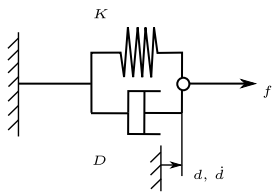
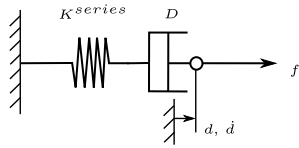
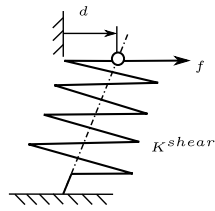
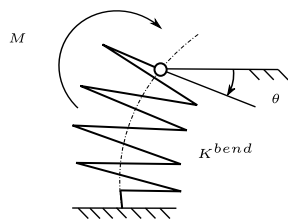
The studied contact models are the *KEC* and *LUT* methods. The differences between each approach can be summarized as follows:

1. Contact detection for the *LUT* method is based on an offline approach where the contact points are determined by interpolating a predefined lookup table. The *KEC* method is an online contact detection approach where the exact position of the contact points can be determined from the online solution of the *KEC* constraints.
2. The second difference is the computation of normal contact forces at the flange. For the *LUT* method, the normal contact forces are defined in terms of stiffness and damping ratio using Hertz's contact theory. This implies that computational efficiency and numerical stability will be affected by the large magnitude values of the stiffness parameters. However, with using the *KEC* method, the normal contact forces are determined through Lagrange multipliers that avoid the numerical disadvantages in the case of stability using high values of contact stiffnesses.

Numerical simulations of vehicle dynamics have been carried out in different case studies with and without track irregularities. Comparison between the kinematic results obtained by

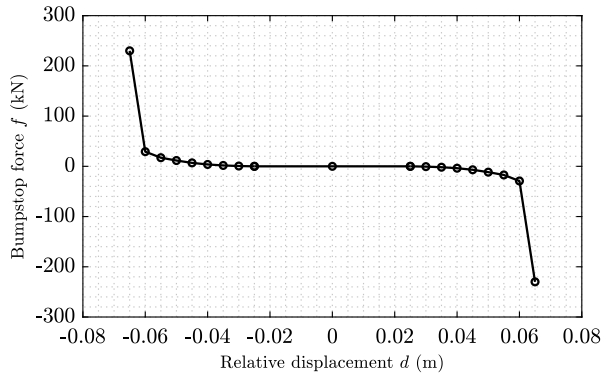
Matlab and the commercial simulation software UM has allowed a reliable validation of both the *LUT* and *KEC* methods. The performance of both approaches is compared in terms of efficiency and accuracy. The *KEC* method is faster (more computationally efficient) than the *LUT* method when using all the studied integrators (up to 20%). Also, as expected for the *LUT* method, results are more accurate when Hertzian stiffness is lower. The integrators *ode15s* and *ode45* showed better efficiency compared to the integrators RK4 and ABM, only when flange contact stiffness K^{hertz} is lower than $1 \cdot 10^{10} \text{ N/m}^{1.5}$. As for the *KEC* method, the integrators *ode15s* and *ode45* are faster than the fixed-step-size case under some specific tolerances.

Appendix A: Elements used for the suspension systems

Element name	Example	Calculation
Spring-damper in parallel		$f^{susp} = Kd + D\dot{d}$
Spring-damper in series		$\dot{f}^{susp}(t) + \frac{K^{series}}{D} f^{susp}(t) = K^{series} \dot{d}$ $f^{susp}(t=0) = f_0^{susp}$
Shear spring		$f^{susp} = K^{shear} d$
Bending spring		$M^{susp} = K^{bend} \theta$

The bumpstop forces f can be parameterized as a continuous functions (cubic splines in this work) of the relative displacement d , as shown in Fig. 27.

Fig. 27 The nonlinear stiffness for longitudinal bumpstop



Appendix B: Computational time for five runs

Table 12 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *LUT* method with Hertzian stiffness at flange $1 \cdot 10^9$ N/m^{1.5}

Integ-	Tol	CPU time 1st (s)	Func-eval-1st	CPU time 2nd (s)	Func-eval-2nd	CPU time 3rd (s)	Func-eval-3rd	CPU time 4th (s)	Func-eval-4th	CPU time 5th (s)	Func-eval-5th
<i>ode15s</i>	$1 \cdot 10^{-6}$	493.00	71273	487.82	71273	492.93	71273	475.32	71273	461.72	71273
	$1 \cdot 10^{-5}$	143.24	17928	127.81	17928	127.20	17928	118.06	17928	118.16	17928
	$1 \cdot 10^{-4}$	85.79	12405	82.95	12405	76.48	12405	79.29	12405	96.90	12405
<i>ode45</i>	$1 \cdot 10^{-6}$	344.74	48511	327.76	48511	374.03	48511	382.00	48511	339.08	48511
	$1 \cdot 10^{-5}$	282.84	33811	242.49	33811	245.72	33811	259.78	33811	257.95	33811
	$1 \cdot 10^{-4}$	324.93	37045	282.51	37045	273.45	37045	283.15	37045	267.22	37045
RK4		734.72	100000	688.87	100000	683.98	100000	692.36	100000	724.20	100000
ABM		408.18	50011	354.05	50011	368.91	50011	360.75	50011	353.15	50011

Table 13 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *LUT* method with Hertzian stiffness at flange $1 \cdot 10^{10}$ N/m^{1.5}

Integ-	Tol	CPU time 1st (s)	Func-eval-1st	CPU time 2nd (s)	Func-eval-2nd	CPU time 3rd (s)	Func-eval-3rd	CPU time 4th (s)	Func-eval-4th	CPU time 5th (s)	Func-eval-5th
<i>ode15s</i>	$1 \cdot 10^{-6}$	653.89	85968	610.55	85968	686.72	85968	653.18	85968	679.43	85968
	$1 \cdot 10^{-5}$	372.46	51736	391.47	51736	385.11	51736	389.24	51736	367.10	51736
	$1 \cdot 10^{-4}$	226.51	32614	216.81	32614	216.61	32614	266.12	32614	225.54	32614

Table 13 (Continued)

Integ-	Tol	CPU time 1st (s)	Func-eval-1st	CPU time 2nd (s)	Func-eval-2nd	CPU time 3rd (s)	Func-eval-3rd	CPU time 4th (s)	Func-eval-4th	CPU time 5th (s)	Func-eval-5th
ode45	$1 \cdot 10^{-6}$	615.99	77011	637.96	77011	624.77	77011	582.01	77011	578.58	77011
	$1 \cdot 10^{-5}$	386.61	49579	442.25	49579	410.13	49579	369.58	49579	467.99	49579
	$1 \cdot 10^{-4}$	271.93	38155	258.03	38155	264.98	38155	286.16	38155	315.09	38155
RK4		698.43	100000	694.88	100000	744.67	100000	735.31	100000	729.37	100000
ABM		401.47	50011	398.92	50011	337.01	50011	364.40	50011	399.35	50011

Table 14 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the LUT method with Hertzian stiffness at flange $1 \cdot 10^{11}$ N/m^{1.5}

Integ-	Tol	CPU time 1st (s)	Func-eval-1st	CPU time 2nd (s)	Func-eval-2nd	CPU time 3rd (s)	Func-eval-3rd	CPU time 4th (s)	Func-eval-4th	CPU time 5th (s)	Func-eval-5th
ode15s	$1 \cdot 10^{-6}$	1728.10	244144	1740.00	244144	1600.56	244144	1579.49	244144	1591.98	244144
	$1 \cdot 10^{-5}$	1202.83	161772	1073.86	161772	1033.54	161772	1005.55	161772	1056.18	161772
	$1 \cdot 10^{-4}$	1051.29	147154	1046.50	147154	985.35	147154	1103.24	147154	1062.70	147154
ode45	$1 \cdot 10^{-6}$	1591.40	203887	1426.20	203887	1429.11	203887	1675.47	203887	1413.85	203887
	$1 \cdot 10^{-5}$	1210.50	136729	1105.41	136729	928.60	136729	962.59	136729	1141.31	136729
	$1 \cdot 10^{-4}$	673.39	91519	686.47	91519	683.42	91519	669.41	91519	660.82	91519
RK4		697.31	100000	823.04	100000	698.23	100000	707.15	100000	688.72	100000
ABM		361.99	50011	349.87	50011	363.83	50011	352.48	50011	307.80	50011

Table 15 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the LUT method with Hertzian stiffness at flange $1 \cdot 10^{12}$ N/m^{1.5}

Integ-	Tol	CPU time 1st (s)	Func-eval-1st	CPU time 2nd (s)	Func-eval-2nd	CPU time 3rd (s)	Func-eval-3rd	CPU time 4th (s)	Func-eval-4th	CPU time 5th (s)	Func-eval-5th
ode15s	$1 \cdot 10^{-6}$	3373.78	453170	3330.28	453170	2879.76	453170	3262.34	453170	3116.91	453170
	$1 \cdot 10^{-5}$	2741.32	380241	2849.83	380241	2632.71	380241	2596.08	380241	2604.17	380241
	$1 \cdot 10^{-4}$	2392.28	337975	2278.72	337975	2499.16	337975	2856.17	337975	2667.23	337975
ode45	$1 \cdot 10^{-6}$	2781.03	370171	3030.14	370171	3315.77	370171	3688.98	370171	4606.37	370171
	$1 \cdot 10^{-5}$	1697.65	236533	1779.44	236533	1699.05	236533	1991.37	236533	1875.88	236533
	$1 \cdot 10^{-4}$	1099.14	158281	1068.69	158281	1242.76	158281	1314.16	158281	1162.69	158281
RK4		675.77	100000	657.59	100000	593.76	100000	621.64	100000	686.72	100000
ABM		302.10	50011	319.09	50011	350.46	50011	387.78	50011	395.54	50011

Table 16 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *KEC* method with different integrators

Integ-	Tol	CPU time 1st (s)	Func- eval- 1st	CPU time 2nd (s)	Func- eval- 2nd	CPU time 3rd (s)	Func- eval- 3rd	CPU time 4th (s)	Func- eval- 4th	CPU time 5th (s)	Func- eval- 5th
<i>ode15s</i>	$1 \cdot 10^{-6}$	700.25	123172	792.50	123172	730.71	123172	757.94	123172	862.18	123172
	$1 \cdot 10^{-5}$	422.75	67333	425.98	67333	405.24	67333	360.47	67333	465.68	67333
	$1 \cdot 10^{-4}$	236.68	39941	202.67	39941	203.95	39941	204.57	39941	204.55	39941
<i>ode45</i>	$1 \cdot 10^{-6}$	644.04	112201	582.96	112201	568.18	112201	561.73	112201	566.01	112201
	$1 \cdot 10^{-5}$	297.70	52927	271.51	52927	262.21	52927	263.17	52927	307.22	52927
	$1 \cdot 10^{-4}$	217.60	38719	217.53	38719	203.34	38719	193.46	38719	191.67	38719
RK4		590.29	100000	613.67	100000	565.60	100000	591.42	100000	584.59	100000
ABM		269.76	50011	246.61	50011	253.45	50011	293.95	50011	262.63	50011

Table 17 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *KEC* method with different integrators after refinement

Integ-	Tol	CPU time 1st (s)	Func- eval- 1st	CPU time 2nd (s)	Func- eval- 2nd	CPU time 3rd (s)	Func- eval- 3rd	CPU time 4th (s)	Func- eval- 4th	CPU time 5th (s)	Func- eval- 5th
<i>ode15s</i>	$1 \cdot 10^{-6}$	611.46	124788	608.69	124788	602.46	124788	603.86	124788	631.08	124788
	$1 \cdot 10^{-5}$	318.63	66208	331.12	66208	321.29	66208	318.57	66208	318.35	66208
	$1 \cdot 10^{-4}$	180.91	37596	181.90	37596	188.10	37596	200.18	37596	196.00	37596
<i>ode45</i>	$1 \cdot 10^{-6}$	610.96	113005	603.39	113005	602.96	113005	606.14	113005	579.53	113005
	$1 \cdot 10^{-5}$	297.06	52747	264.96	52747	263.59	52747	266.39	52747	265.84	52747
	$1 \cdot 10^{-4}$	207.29	38287	204.34	38287	192.58	38287	194.58	38287	196.09	38287
RK4		484.88	100000	485.44	100000	472.32	100000	480.49	100000	489.68	100000
ABM		281.09	50011	292.30	50011	268.26	50011	270.96	50011	277.85	50011

Table 18 Computation cost for track radius $R = 500$ m with forward velocity $V = 20$ m/s on 500 m track by using the *LUT* method with Hertzian stiffness at flange $1 \cdot 10^{10}$ N/m^{1.5} after refinement

Integ-	Tol	CPU time 1st (s)	Func- eval- 1st	CPU time 2nd (s)	Func- eval- 2nd	CPU time 3rd (s)	Func- eval- 3rd	CPU time 4th (s)	Func- eval- 4th	CPU time 5th (s)	Func- eval- 5th
<i>ode15s</i>	$1 \cdot 10^{-6}$	632.44	86595	645.81	86595	665.05	86595	639.87	86595	687.02	86595
	$1 \cdot 10^{-5}$	349.64	49964	413.11	49964	390.06	49964	351.50	49964	447.04	49964
	$1 \cdot 10^{-4}$	252.74	32594	247.04	32594	221.31	32594	255.72	32594	253.14	32594

Table 18 (Continued)

Integ-	Tol	CPU time 1st (s)	Func- eval- 1st	CPU time 2nd (s)	Func- eval- 2nd	CPU time 3rd (s)	Func- eval- 3rd	CPU time 4th (s)	Func- eval- 4th	CPU time 5th (s)	Func- eval- 5th
<i>ode45</i>	$1 \cdot 10^{-6}$	566.49	77323	537.21	77323	528.05	77323	537.56	77323	542.45	77323
	$1 \cdot 10^{-5}$	346.01	49315	317.53	49315	305.15	49315	309.19	49315	351.40	49315
	$1 \cdot 10^{-4}$	250.43	38137	241.77	38137	274.68	38137	265.75	38137	265.79	38137
RK4		606.72	100000	597.32	100000	607.46	100000	644.40	100000	683.03	100000
ABM		342.94	50011	314.21	50011	335.81	50011	393.37	50011	356.16	50011

Acknowledgements The authors thank for the support given by Business of Finland under the SmartTram-LUT project with reference 6292/31/2018. In addition, the authors thank for the help of the Andalusian Regional Government, in Spain, under the PAIDI 2020 program with project reference P18-RT-1772. All this support is gratefully acknowledged.

Funding Note Open Access funding provided by LUT University (previously Lappeenranta University of Technology (LUT)).

Declarations

Conflicts of interest The authors declare that there is no conflict of interest to this work.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Bruni, S., Meijaard, J.P., Rill, G., Schwab, A.L.: State-of-the-art and challenges of railway and road vehicle dynamics with multibody dynamics approaches. *Multibody Syst. Dyn.* **29**(1), 1–32 (2020)
2. Ferreira, J., Flores, P., Marques, F.: On the utilization of simplified methodologies for the wheel–rail contact. In: *International Symposium on Multibody Systems and Mechatronics*. Springer, Cham (2021)
3. Iwnick, S.: Manchester benchmarks for rail vehicle simulation. *Veh. Syst. Dyn.* **30**(3–4), 295–313 (1998)
4. Bezin, Y., Pålsson, B.A.: Multibody simulation benchmark for dynamic vehicle–track interaction in switches and crossings: modelling description and simulation tasks. *Veh. Syst. Dyn.* 1–16 (2021). <https://doi.org/10.1080/00423114.2021.1942079>
5. Shabana, A.A., Zaazaa, K.E., Escalona, J.L., Sany, J.R.: Development of elastic force model for wheel/rail contact problems. *J. Sound Vib.* **269**(1–2), 295–325 (2004)
6. Magalhães, H., Marques, F., Liu, B., Antunes, P., Pombo, J., Flores, P., Ambrósio, J., Piotrowski, J., Bruni, S.: Implementation of a non-Hertzian contact model for railway dynamic application. *Multibody Syst. Dyn.* **48**(1), 41–78 (2020)
7. Marques, F., Magalhães, H., Pombo, J., Ambrósio, J., Flores, P.: A three-dimensional approach for contact detection between realistic wheel and rail surfaces for improved railway dynamic analysis. *Mech. Mach. Theory* **149**, 103825 (2020)

8. Liu, B., Bruni, S.: Comparison of wheel–rail contact models in the context of multibody system simulation: Hertzian versus non-Hertzian. *Veh. Syst. Dyn.* 1–21 (2020). <https://doi.org/10.1080/00423114.2020.1847297>
9. Sugiyama, H., Suda, Y.: On the contact search algorithms for wheel/rail contact problems. *J. Comput. Nonlinear Dyn.* **4**(4), 041001 (2009)
10. Sugiyama, H., Araki, K., Suda, Y.: On-line and off-line wheel/rail contact algorithm in the analysis of multibody railroad vehicle systems. *J. Mech. Sci. Technol.* **23**(4), 991–996 (2009)
11. Sugiyama, H., Sekiguchi, T., Matsumura, R., Yamashita, S., Suda, Y.: Wheel/rail contact dynamics in turnout negotiations with combined nodal and non-conformal contact approach. *Multibody Syst. Dyn.* **27**, 55–74 (2012)
12. Escalona, J.L., Aceituno, J.F.: Multibody simulation of railway vehicles with contact lookup tables. *Int. J. Mech. Sci.* **155**, 571–582 (2019)
13. Piotrowski, J., Liu, B., Bruni, S.: The Kalker book of tables for non-Hertzian contact of wheel and rail. *Veh. Syst. Dyn.* **55**(6), 875–901 (2017)
14. Piotrowski, J., Bruni, S., Liu, B., Gialleonardo, E.D.: A fast method for determination of creep forces in non-Hertzian contact of wheel and rail base on a book of tables. *Multibody Syst. Dyn.* **45**, 169–184 (2019)
15. Marques, F., Magalhães, H., Liu, B., Pombo, J., Flores, P., Ambrósio, J., Piotrowski, J., Bruni, S.: On the generation of enhanced lookup tables for wheel–rail contact models. *Wear* **434–435**, 202993 (2019)
16. Escalona, J.L., Aceituno, J.F., Urda, P., Balling, O.: Railway multibody simulation with the knife-edge-equivalent wheel–rail contact constraint equations. *Multibody Syst. Dyn.* **48**(4), 373–402 (2020)
17. Aceituno, J.F., Urda, P., Briales, E., Escalona, J.L.: Analysis of the two-point wheel–rail contact scenario using the knife-edge-equivalent contact constraint method. *Mech. Mach. Theory* **148**, 103803 (2020)
18. Muñoz, S., Aceituno, J.F., Urda, P., Escalona, J.L.: Multibody model of railway vehicles with weakly coupled vertical and lateral dynamics. *Mech. Syst. Signal Process.* **115**, 570–592 (2019)
19. Schupp, G., Weidemann, C., Mauer, L.: Modelling the contact between wheel and rail within multibody system simulation. *Veh. Syst. Dyn.* **41**(5), 349–364 (2004)
20. Meli, E., Magheri, S., Malvezzi, M.: Development and implementation of a differential elastic wheel–rail contact model for multibody applications. *Veh. Syst. Dyn.* **49**(6), 969–1001 (2011)
21. Zhai, W.: Two simple fast integration methods for large-scale dynamic problems in engineering. *Int. J. Numer. Methods Eng.* **39**(24), 4199–4214 (1996)
22. Baeza, L., Roda, A., Nielsen, J.C.: Railway vehicle/track interaction analysis using a modal substructuring approach. *J. Sound Vib.* **293**(1–2), 112–124 (2016)
23. Arnold, M., Fuchs, A., Führer, C.: Efficient corrector iteration for DAE time integration in multibody dynamics. *Comput. Methods Appl. Mech. Eng.* **195**(50–51), 6958–6973 (2006)
24. Escalona, J.L., Yu, X., Aceituno, J.F.: Wheel–rail contact simulation with lookup tables and KEC profiles: a comparative study. *Multibody Syst. Dyn.* **52**(4), 339–375 (2021). <https://doi.org/10.1007/s11044-020-09773-7>
25. Shabana, A.A., Zaazaa, K.E., Sugiyama, H.: *Railroad Vehicle Dynamics: A Computational Approach*. CRC Press, Boca Raton (2007)
26. Popp, K., Schiehlen, W.: *Ground Vehicle Dynamics*. Springer, Berlin (2010)
27. Goldsmith, W.: *Impact: The Theory and Physical Behaviour of Colliding Solids*. Edward Arnold Ltd., London, England (1960)
28. Flores, P., Ambrósio, J.: On the contact detection for contact–impact analysis in multibody systems. *Multibody Syst. Dyn.* **24**(1), 103–122 (2010)
29. Polach, O.: A fast wheel–rail forces calculation computer code, *Veh. Syst. Dyn.* **33** (Proceedings of the 16th IAVSD Symposium held in Pretoria, South Africa, August 30 – September 3, 1999) 728–739 (1999). <https://doi.org/10.1080/00423114.1999.12063125>
30. Polach, O.: Creep forces in simulations of traction vehicles running on adhesion limit. *Wear* **258**(7–8), 992–1000 (2005) <https://doi.org/10.1016/j.wear.2004.03.046>
31. Shabana, A.A.: *Computational Dynamics*. John Wiley & Sons, Inc, United Kingdom (2009)
32. Ashino, R., Nagase, M., Vaillancourt, R.: Behind and beyond the Matlab ODE suite. *Comput. Math. Appl.* **40**(4–5), 491–512 (2000)
33. Shampine, L.F., Reichelt, M.W.: The Matlab ODE suite. *SIAM J. Sci. Comput.* **18**(1), 1–12 (1997)
34. Burgermeister, B., Arnold, M., Esterl, B.: DAE time integration for real-time applications in multi-body dynamics. *ZAMM–J. Appl. Math. Mech./Z. Angew. Math. Mech.: Appl. Math. Mech.* **86**(10), 759–771 (2006)
35. Claus, H., Schiehlen, W.: Modeling and simulation of railway bogie structural vibrations. *Veh. Syst. Dyn.* **29**, 538–552 (1998)
36. Pogorelov, D.Y.: Simulation of rail vehicle dynamics with universal mechanism software. In: *Rail Vehicle Dynamics and Associated Problems*, pp. 13–58. Silesian University of Technology, Gliwice (2005)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Xinxin Yu¹  · Javier F. Aceituno² · Emil Kurvinen³ · Marko K. Matikainen¹ · Pasi Korkealaakso⁴ · Asko Rouvinen⁴ · Dezhi Jiang¹ · José L. Escalona⁵ · Aki Mikkola¹

✉ X. Yu

xinxin.yu@lut.fi

J.F. Aceituno

jaceitun@ujaen.es

E. Kurvinen

emil.kurvinen@oulu.fi

M.K. Matikainen

marko.matikainen@lut.fi

P. Korkealaakso

pasi.korkealaakso@mevea.com

A. Rouvinen

asko.rouvinen@mevea.com

D. Jiang

dezhi.jiang@student.lut.fi

J.L. Escalona

escalona@us.es

A. Mikkola

aki.mikkola@lut.fi

¹ Dept. of Mechanical Engineering, LUT University, Lappeenranta, Finland

² Dept. of Mechanical and Mining Engineering, University of Jaén, Jaén, Spain

³ Dept. of Materials and Mechanical Engineering, University of Oulu, Oulu, Finland

⁴ Mevea Ltd., Lappeenranta, Finland

⁵ Dept. of Mechanical and Manufacturing Engineering, University of Seville, Seville, Spain