Distributed Numerical Calculations of Wear in the Wheel-Rail Contact

Kurt Frischmuth and Dirk Langemann

Universität Rostock, FB Mathematik, D-18051 Rostock, Germany

Abstract. We study numerical aspects of wear problems in railway mechanics. A short overview of the needed components – a dynamical vehicle/track model, contact geometry and contact mechanics – is given to demonstrate the complexity of the problem. An algorithm for the integration of the domain evolution equation describing wear in the long time scale is presented and discussed. Solutions for simple choices of the model components are studied. Finally, ways to couple the most advanced available models via internet programming are demonstrated on the basis of a vehicle model due to Meinke in a co-simulation with our own geometry program.

1 Introduction

We describe wear in the framework of *domain evolution problems* (cf Sethian, [20], and Fig. 1). Related problems are e.g. curvature driven flows, Stefan

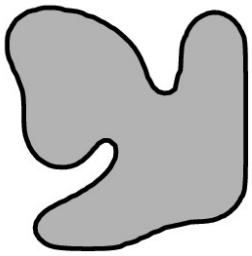

Fig. 1. Example of a domain (used as an extreme test case for assessing integration methods). Under curvature driven motion, $\mathcal{F} = -\kappa$, where κ denotes the curvature of the boundary, the domain should become convex and vanish to a point. Poor numerical methods, however, lead to artificial oscillations of the boundary which in the case of railway mechanics may look like corrugations

problem or the osmotic cell problem. In our case, the speed of the boundary (contact surface) is determined by a *wear law*, the latter using data from a dynamical model as input.

For this paper, the wear process is understood as a pure geometrical change of the body under consideration, i.e. as a shift of the boundary in direction of the inner normal. For now, we do not allow for changes inside the material. Our assumptions lead to the formulation

$$\dot{x}(\tau, t) = \mathcal{F}\big(x(\cdot, t), t\big)\boldsymbol{n}\,. \tag{1}$$

Here x is a point on the actual position of the body's surface at time t. The variable τ is used to parameterize the surface.
The speed $\mathcal{F}$ at which the wear surface retreats is determined by a so-called *speed law*. In general, it is not defined by a formula – as it is in the case of curvature driven flows, which we use frequently for testing numerical procedures.
In railway mechanics, to obtain the speed $\mathcal{F}$ at a given position on the actual wear surface, we need to perform a simulation of the motion of the vehicle. During this simulation, given the present surface geometry, the intensity of power dissipation, and hence the wear intensity are determined. Obviously, this is a numerically very expensive procedure, and the result depends on the driving conditions of the simulated vehicle.
The time integration of (1) can be described by the following feedback loop, cf. Fig. 2, in which dynamics, geometry and contact mechanics are coupled together. The numerical solution of a wear problem requires hence two inte-

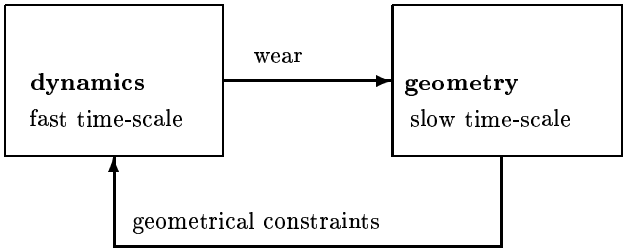

Fig. 2. Knothe's feedbackloop (simplified for autonomous problem)

gration procedures. In an outer loop, solving integration of (1) with respect to the so-called long time scale (or slow time scale), the surface is updated. In an inner loop, simulation with respect to the fast time scale has to be performed. This procedure is – from the point of view of the outer loop – an auxiliary problem, from which we obtain the wear intensity which has the character of a probabilistic density. Here qualitative aspects of the vehicle model, like periodicity, quasi-periodicity or ergodicity, play an important role, [18].

To illustrate the time scales, consider a wheelset rolling at 200 km/h. Under normal conditions, 50 million rotations remove a layer of about 100 µm ... 1 mm. Hence a single-atomic layer of material is abrased in one second, in the same time we observe several recurrences of the lateral motion, 16 revolutions and about 100 oscillations of the normal contact. A reasonable step for the outer time integration would be the order of hours – which would still require about 1000 (quite expensive) steps. For the inner integration, a time step around a millisecond is required to obtain a reasonable accuracy. The ratio is hence of the order of 10^6.

2 Components

In this section we collect some of the ingredients needed to build models capable of describing wear distributed over the circumference of a railway wheel. These components may be combined in different ways to create models for simulations. Our minimal collection may be extended by supplements from other groups.

2.1 Nonlinear Dynamics

We assume that the driving motion is governed by a system of nonlinear equations of motion of the general form

$$m\big(q(t)\big)\ddot{q}(t) = f\big(t, q(t), \dot{q}(t), r(t)\big)\ . \tag{2}$$

Here we denote by $q \in \mathbb{R}^d$ generalized coordinates, t time, $\dot{}$ time derivative, m the mass matrix and and f generalized forces. External control is expressed

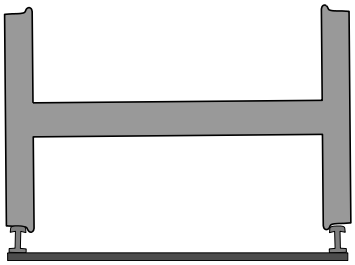

Fig. 3. Wheelset on rails. The evolving surfaces result in complicated relations between lateral shift, elevation and contact point position. For an elastic axle, such relations (dependent on yaw and, in general, also on revolution angle) hold for each of the wheels. For a rigid wheelset, also the rolling angle is determined by a holomorphic constraint

by $r(t)$, which may be a force or a prescribed travelling speed.
The dynamical behavior of a single wheelset, cf. Fig. 3, is already very complex. For low speed, we have a stable trivial solution. Above a critical speed, there is a stable limit cycle. For further increasing speed, there is phase doubling and transition to chaos, cf. for instance [11] and [22].
For the purpose of wear calculation, the dynamical model has to be adapted to the changing geometry in the contact region. This will be discussed in the next two subsections.

2.2 Rolling Contact and Dry Friction

A major component of the dynamical system introduced in the previous subsection is the force vector f on the right-hand side of the equations of motion (2), in particular, the contact forces are difficult to model. For the calculation of friction forces, we have the standard options of the Vermeulen-Johnson law [10], or the Fastsim [14,21] and Contact [13] algorithms. For the long time simulations we need to perform in order to obtain accurate wear speeds, we

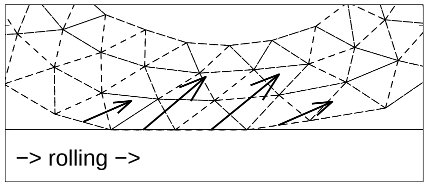

Fig. 4. Disk wheel rolling, accelerating moment and counteracting force F_C. Sparsely discretized 2D-model of a tyre

prefer simpler friction laws in form of algebraic expressions, cf. also [2]. Even a rough approximate solution, cf. Fig. 4, of the underlying Signorini problem, basing on influence functions obtained within the Zastrau/Nackenhorst project, leads to excessive computation times, and is at this moment not possible to carry out for our purposes. In Fig. 5 we present results for the

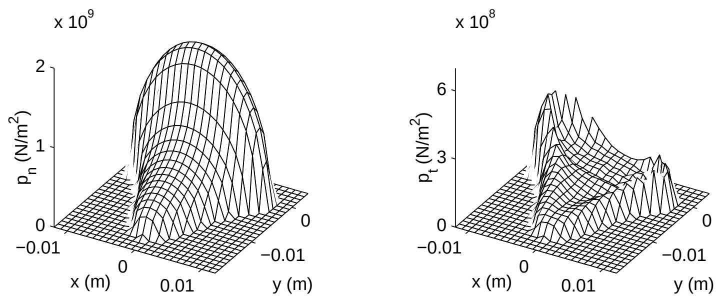

Fig. 5. Contact forces. **(a)** normal pressure. **(b)** tangential forces

tractions in the contact zone between wheel and rail for a large normal force. This local distribution of the friction force over the *contact patch* determines how the *dissipated power* has to be mapped to the wear surfaces. In the next subsection, we discuss how to find the location of the *contact point* around which said contact patch is located.

2.3 Geometry

A very sensitive aspect of modeling rolling of realistic wheels on rails is the geometry of the contact, in particular the effective calculation of the geometrical points of contact. Those points are needed for the dynamical model – the calculation of friction forces – and also for the wear model – for assigning the frictional power at a given instant to a place on the contact evolving surface.

For given mbs-coordinates of the bodies, let τ_i^b be the ith surface coordinate of the bth body, $b = 1, 2$, accordingly $n^b = (n_i^b)_{i=1..3}$ the (unique) outer

normal unit vectors to those surfaces x^1, x^2. Then necessary conditions for contact may by formulated as

$$x^1(\tau^1) + \lambda^1 n^1(\tau^1) = x^2(\tau^2)\,, \tag{3}$$
$$x^2(\tau^2) + \lambda^2 n^2(\tau^2) = x^1(\tau^1)\,. \tag{4}$$

This system implies that the normal directions coincide, and that shifting the bodies rigidly in that common normal direction, brings the points $x^1(\tau^1)$ and $x^2(\tau^2)$ to the same spot. This system a is 6×6 nonlinear problem, and for general non-convex bodies hard to solve.
For perfect wheels (axial-symmetric) and rails (prismatic), the system can be simplified considerably, cf. [1]. The solution can be determined from a scalar minimization problem, cf. Fig. 6. Even so, due to the non-convexity, a

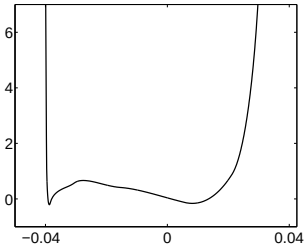

Fig. 6. Cut through the distance function. The minimization of the distance function between wheel and rail surfaces is a non-convex problem, difficult for the existence of more than one local minimum. The distance function depends continuously on the mbs-coordinate q, the minimizer, however, may jump

very accurate starting value for the minimizer has to be found before a fast iterative method (Newton-type) can be successfully applied.
For worn surfaces, reduction to a scalar problem is in general not possible. An implementation of a reasonably quick solver for (3)-(4) is due to Hänler, [9].
For a prismatic rail, however, the simplification still works. Considerable reduction of numerical effort comes from the fact that the first component of the rail's normal vanishes everywhere.
Let us label the wheel by superscript 1, rail by 2, axial coordinate on the wheel by τ_1^1, angular by τ_2^1, and define a function $\tau^1(\tau_1^1) = (\tau_1^1, \tau_2^1(\tau_1^1))$ by the condition

$$\chi(n^1(\tau_1^1, \tau_2^1(\tau_1^1)), q)\boldsymbol{e}_1 = 0\,. \tag{5}$$

Here $\boldsymbol{e}_1$ points in the direction along the rail, $\boldsymbol{e}_3$ is vertical, and $\chi(x, q)$ is the actual position of wheel particle x under mbs-coordinates q. Then the parameter of the contact point on the wheel surface is defined via

$$\tau_{1\,c}^1(q) = \operatorname{argmin}\big(\chi(x^1(\tau^1(\tau_1^1)), q) - x^2(x_2^2(\chi(\tau^1(\tau_1^1), q)))\big)\boldsymbol{e}_3\,. \tag{6}$$

The contact point is hence $x^1(\tau^1(\tau_{1\,c}^1))$. Here we assumed that for the rail surface, we have a parametrization of the form $x^2 = x^2(x_2^2)$, i.e. τ_1^2 is along the rail and does not enter the equation, $\tau_2^2 \equiv x_2^2$.

It is obvious that a fast evaluation of the above contact conditions is essential. However, there is no point in tuning procedures for the original profiles, e.g. S1002/UIC 60, since the profiles change essentially during wear simulation. To make this clear, compare the location of the contact point in dependence on lateral shift, once calculated for virgin profiles, once for slightly worn ones, Fig. 7. Note that the contact geometry is the main interface point where the

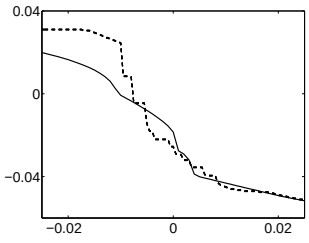

Fig. 7. The axial position of the point of geometrical contact on the wheel surface, for vanishing yaw and roll angles. The continuous line is for the virgin profile pair S1002/UIC60. The dashed line corresponds to slightly worn wheel profile, rail profile still unchanged

feedback loop closes, Fig. 2. Dynamical models which use force models basing on frozen profiles can't work beyond the mere onset of wear.

2.4 Wear Laws

Due to a simple assumption, frequently quoted as Archard's law, cf. [5], for abrasive wear $\mathcal{F}$ depends on the frictional power density dissipated at $x(\tau, t)$. Hence the term $\mathcal{F}$ in (1) is obtained by running the dynamical model (2), and averaging a function of the dissipated power, which is then mapped to the surface of the body in order to obtain a wear distribution.
For example, [19] uses a piecewise constant function for mild and strong abrasion, respectively.
For a comprehensive discussion of factors contributing to wear, we refer to [16].
Observations that reshaped rails wear faster than new ones motivated assumptions about *state variables* assigned to the surface [6].
For this paper, we want to focus on numerical aspects of wear simulation, rather than on sophistication of the wear model. Thus we assume proportionality between worn mass and dissipated energy, which results finally in an equation of the form

$$\mathcal{F} = -\beta s \sigma_t \tag{7}$$

where s is the local creepage, σ_t tangential stress, and β a positive constant. For calculations on the long time scale, on the right-hand side the mean value over a sufficiently long interval of the fast time scale has to be taken.
In terms of Meinder's model, our assumption means that during the exploitation of a given vehicle, we do not change regimes between mild and strong abrasion.

Given this assumption, by matching data on wear of ICE-wheels, the value of the single wear constant β can be found to be of the order 10^{-12} if standard units are used. We prefer, however, to scale the long time scale to 1, and hence we may assume $\beta = 5 \cdot 10^{-6}$.

3 Models

It was Brommundt's original idea, cf. [4] to study evolution of wear in the long time scale on the basis of extremely simple dynamical models in the short time scale. This allowed the application of asymptotic expansions and gave first results very quickly. Despite the fact that we have at our disposal a palette of much more advanced vehicle models, we chose for this paper to follow this approach. It allows us to compare the effects of different analytical and numerical methods on one and the same model. We will see that poor approximations may contribute more to the output than the model itself.
As a **benchmark** model, we propose a rolling disk, driven by a constant moment against a quadratic drag force. The equations of motion for the three degrees of freedom (longitudinal and vertical displacement, angle of rotation) are given by

$$m\ddot{q}_1 = F_R - c|\dot{q}_1|\dot{q}_1 \,, \tag{8}$$
$$m\ddot{q}_2 = F_G + mg + F_N \,, \tag{9}$$
$$J\ddot{q}_3 = M + r(\tau)F_R \tag{10}$$

with

$$\tau = -q_3 \,, \tag{11}$$
$$s = \dot{q}_1 + \dot{q}_3 r(\tau) \,, \tag{12}$$
$$F_N = k\big(r(\tau) - q_2\big)^{3/2} - d\dot{q}_2 \,, \tag{13}$$
$$F_R = -\mu F_N s \,. \tag{14}$$

Equation (11) plays the role of a *contact geometry module*, (12-14) substitute

k	$= 1.6742 \cdot 10^9\,\mathrm{N/m^{\frac{2}{3}}}$
d	$= 100\,\mathrm{Ns/m}$
μ	$= 0.2\ \mathrm{s/m}$
m	$= 300\ \mathrm{kg}$
J	$= 37.5\,\mathrm{kgm^2}$
c	$= 0.5\,\mathrm{Ns^2/m^2}$
F_G	$= -50000\,\mathrm{N}$
g	$= -9.81\,\mathrm{m/s^2}$
M	$= -225\,\mathrm{Nm}$

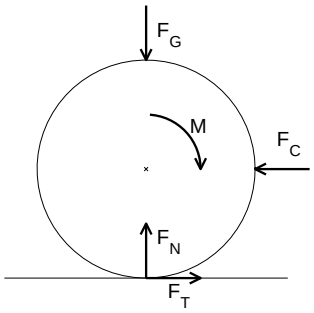

Fig. 8. Disk wheel as a benchmark model

a *contact force model*, the term containing the constant c in (8) simulates the *control motion*. The constants given in Fig. 8 lead to a steady state with longitudinal speed of around 30 m/s, and a frictional power around 19 W. For given radius function $r(\cdot)$, the mean value of frictional power tends to a limit, multiplication with the wear constant β gives finally the evolution speed $\mathcal{F}$:

$$r_t(\tau, t) = -\beta \overline{s F_R} \tag{15}$$

$$\beta = 5 \cdot 10^{-6} \tag{16}$$

where overline stands for taking the temporal mean value, and time is measured in a characteristic time for the life cycle of a wheel. For this case, the approximation of the outer normal by the position vector in (15) is obviously acceptable.

The initial geometry imperfection – a single trough – is defined by B–splines, the data can be downloaded from our website.[1]
In Brommundt's original paper, the dynamical model was completely linear, and the normal force was assumed to be constant. The only nonlinear relation was (7). The out-of-roundness is modeled by a sequence of Fourier coefficients. Under these assumptions, every harmonic excites an oscillation of slip and friction force with the same frequency, higher harmonics are caused by the non-linear abrasion hypothesis.
Similar results were obtained by use of approximations to $\mathcal{F}$ in terms of Taylor coefficients of the actual wear surface, [6]. Those lead to evolution equations of the general form

$$r_t = F(u, \nabla r, \nabla^2 r, ...) \tag{17}$$

which for suitable functions F yield speed dependent growth or decay of wavyness of the surface.

4 Numerics

For more realistic models of wear in railway mechanics, we have typically two layers of numerical calculations. In an inner loop, the equations of motion of the dynamical model (vehicle, track) have to be solved. Along with the solution, data essential for wear have to be collected and processed. We call this most essential ingredient of a coupled wear model the *wear collecting algorithm*. From this inner procedure we obtain the speed function $\mathcal{F}$ in dependence on the momentary geometry (together with the control motion). Now, the speed function is passed to the outer time-integration loop in which the wear patterns finally evolve.

[1] $r = 0.5\,\mathrm{m}$ everywhere except on an arch of length $0.2\,\mathrm{m}$ where it is reduced by $75\,\mu\mathrm{m}$. Details on `http://alf.math.uni-rostock.de/~kurt/DFG/benchmark/`

A central problem here is accuracy. It turns out, that already for very simple models, it is hard to obtain an accurate speed function. One limiting factor is computation time, because we need trajectories of the dynamical system with rather high precision, but also for a reasonably long time span.
Once the problem of obtaining a good approximation for $\mathcal{F}$, the outer integration loop is straightforward. Since evaluations of $\mathcal{F}$ are extremely costly, we do not recommend too much sophistication. However, as examples show, the common practice of simply applying Euler's explicit method is not the best choice.
Here we need some denotations which we introduce below.

4.1 Discretization

Discretization of the functional variable x and separation leads to an ODE problem of the form

$$\dot{y} = g(t, y, u)\,, \tag{18}$$
$$\dot{u} = h(t, y, u)\,. \tag{19}$$

Here we denote by $y \in \mathbb{R}^{2d}$ the pair of vectors $(q, \dot{q})$ describing the dynamical state of the driving model, while $u \in \mathbb{R}^n$ is the vector of nodal values of a suitable presentation of the change in surface positions on a given grid.
The functions g and h on the right-hand sides are derived from the forces f and the speed function $\mathcal{F}$, cf. (2) and (7), respectively.
Now, instead of solving the system (18)–(19) in a straightforward way on a given time interval $[0, T]$ with huge T (not desirable), or solving a modified system with h replaced by ah, a – big, on $[0, T/a]$ instead (common, but dangerous, cf. [17]), we apply the algorithm from [7] for integrating the coupled system.

step 1: On the time interval $[t_i, t_i + \Delta t_i]$ (which is very long in terms of the y-variables but very short in terms of the u-variables) we integrate the y-equations with frozen u

$$\bar{y}(t) = y(t_i) + \int_{t_i}^{t} g(t, y(t), u(t_i))\, dt\,. \tag{20}$$

Thus, this low-dimensional part of the system is uncoupled and can be integrated by any suitable method.

step 2: An approximation for $u(t_{i+1})$, $t_{i+1} = t_i + \Delta t_i$, is calculated by

$$u(t_{i+1}) = u(t_i) + \int_{t_i}^{t_{i+1}} h(t, \bar{y}(t), u(t_i))\, dt \tag{21}$$

where we use $\bar{y}$ instead of the exact solution.

For the efficiency and accuracy of this algorithm, it is crucial how we approximate the integral on the right-hand side. Using the property that

$$\Delta t_i^{-1} \int_{t_i}^{t_i+\Delta t_i} h(t, \bar{y}(t), u(t_i))\, dt \tag{22}$$

approaches a limit very quickly, we may substitute the integral over the interval $[t_i, t_i + \Delta t_i]$ by

$$\frac{\Delta t_i}{\Delta t_{sim}} \int_{t_i}^{t_i+\Delta t_{sim}} h(t, \bar{y}(t), u(t_i))\, dt\,, \tag{23}$$

where Δt_{sim} is a reasonable time span for simulation of the y-system. The latter is chosen by monitoring the limit of the integral mean value, terminating when changes become negligible.
In the sequel, given the interpretation of (18) and (19), we will refer to step 1 of the algorithm as *wear collection*, and to step 2 as *geometry updating*.

4.2 Distributed Calculations

The algorithm described in the previous subsection is not well suited for *parallel computations*, it is essentially *sequential*. Nonetheless, for several practical reasons, it is sensitive to perform some routines on different computers. In fact, often more advanced models are not easy to port from their original environment, hence it is far easier to couple them while leaving them where they were developed. (Usually, they are still being developed.)
Thus we agreed with other groups of the DFG Priority Programme 1015 (Meinke/Meinders, Stuttgart, [19] and Popp/Kaiser, Hannover, [12]) on coupling our models by the most simple possible interface – which is on the other hand the safest to work with. Anticipating that all partial models have a considerable cost in terms of CPU-time, it is clear that time for the exchange of data is not the limiting factor. All results from one model are written on ASCII text files and then transported by FTP to the remote machine. Timing is done by waiting for needed input files. For all tests, losses due to communication time were minimal.
We implemented distributed calculations with Meinke/Meinders the following way. A (simple) vehicle model used as driving component is integrated on one CPU (in Stuttgart). Along the trajectories dissipated power is calculated and projected onto the surface grid by a method using geometrical data on the contact geometry. The resulting wear intensity is sent to our server, where the surface is updated, new contact geometry data are calculated and sent to (the meantime idle) machine in Stuttgart.
To make this loop really closed, the dynamical model has to depend on the contact geometry. In our case, this is the case due to dependencies of forces

and moments on the position of the contact point and the curvatures of the profiles there. Further, the wear collection is very sensitive to the contact point position, cf. Fig. 7.
It has to be mentioned that for the simulation of realistic wear processes the number of outer loop time steps is not excessive, it rarely exceeds 100.
Total time needed for the file transfer is measured in seconds, while the overall computation time is – even for simple components – the order of hours. Consequently, we have practically no loss of efficiency if compared with running the simulation on one computer. What we gain is that we save the effort to get always the actual version of the software developed in the other group (or groups), together with all the formal problems that arise from that (e.g. obtaining licences).

4.3 Testing the Speed Function

Before approaching the very time consuming integration of the shape evolution equation (outer loop), it is sensitive to evaluate and discuss the speed function $\mathcal{F}$, i.e. to test step 1 of the algorithm, wear collection, without geometry update. We perform all tests on the initial radius described in Sec. 3.
There are several sources of errors (from our own experience), which we want to point out. All have catastrophal impact on the results.
The typical errors come from (i) too small accuracy of the time integration of (18), (ii) bad discretization of the manifold x and (iii) too small Δt_{sim} in the algorithm.
For (i) we remark that matlab's built-in solvers, with standard accuracy options, perform poorly on the benchmark problem. In order to get a feeling for the model, it is helpful to monitor the vertical motion, presented by Fig. 9. The trajectory converges to the presented cycle, the 'shortcut' caused

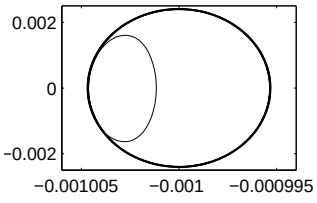

Fig. 9. Integrating the benchmark model. Vertical phase as signature of the dynamical model

by the initial imperfection is taken once in eight rounds. As illustration for (ii), we compare the correct result, confirmed by several methods, with erroneous wear speeds, Fig. 10. The source of the error here is a typical effect for trigonometric polynomials – on localized defects, like a trough, they tend to oscillations and miss the extrema. From (c) we infer that even the best approximation by a trigonometric polynomial is not suited for reliable wear simulations. A comparison with models without dynamical response in ver-

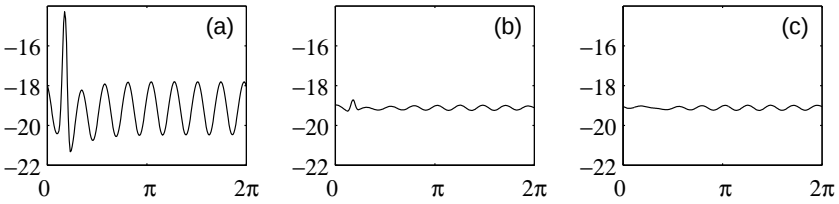

Fig. 10. **(a)** wear intensity for benchmark model, **(b)** same with Fourier approximated radius, **(c)** Fourier approximation of (b)

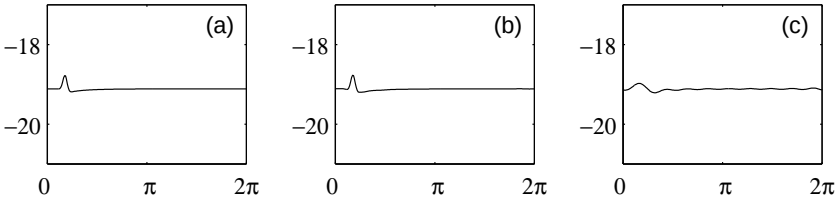

Fig. 11. **(a)** wear intensity in model with frozen normal force, **(b)** same with Fourier approximated radius, **(c)** Fourier approximation of (b)

tical direction – i.e. with normal frozen force, equal to gravity plus external load, shows large differences. In Figs 10 and 11, we used the same order of trigonometric polynomials $N = 10$ as in [4], i.e. 21 coefficients. Our results show the importance of the vertical components of the dynamical model, and discourage the use of trigonometric polynomials.
Let us visualize the third source of errors in a modified version of the benchmark model – we superimpose a lateral motion according to a linear model

$$m\ddot{q}_4 = -k_4 q_4 \qquad \text{with } k_4 = 4147\,\mathrm{N/m} \tag{24}$$

A typical run of the algorithm collecting data on the wear distribution yields the following sequence of figures, Fig. 12 It is obvious from Fig. 12 that too

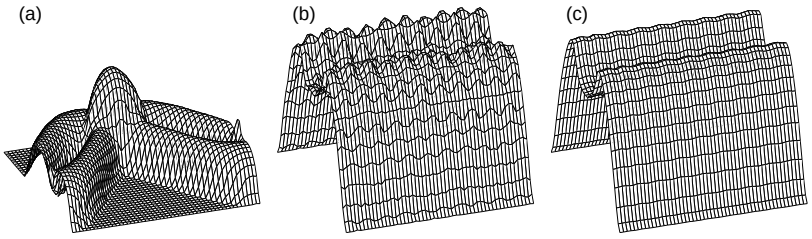

Fig. 12. Stages of the wear collection. **(a)** early. **(b)** medium. **(c)** final

quick termination of the wear collection may lead to possibly most spectac-

ular, but very wrong results. Note that for Fig. 12 (a) a different scale had to be used.

5 Results

In this section we present several results obtained for the integration of the surface evolution equation (1) with the speed function $\mathcal{F}$ calculated by the algorithm from Sec. 4. As a rule, dynamical simulations (inner loop) were performed on a SUN workstation with 1.2 GHz processor, geometry and graphics on a PC with and AMD Athlon processor at 1.6 GHz, data files were exchanged by was done by FTP.
The examples relate to the questions in how far wear patterns are determined by initial imperfections, and whether they stay in place or travel.
For the benchmark model from Sec. 3, a time interval of length 4.0 (corresponds to ca. 1000 h) gives reasonable wear patterns. The results are pretty robust with respect to parameters of the numerical method, e.g. the time step of the outer loop or the use of predictor/corrector steps.
We obtain always a pattern with 8 maxima and minima, as indicated by Fig. 10 (a). With the evolution, the wear speed tends almost to a sinusoidal distribution.
For higher speed, the number of minima decreases. However, the variation of the radius may grow considerably faster, so that we get lift off very quickly, [3]. This violates the model assumptions, hence we stop simulations at the first occurrence of zero normal force. For a three times higher speed, we obtain 3 humps, but life time (time to first bumping of the wheel) decreases 8 times, cf. Fig. 13. We observe further that the frictional power is phase-

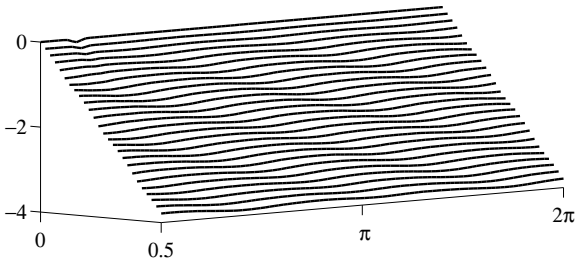

Fig. 13. Change of radius (in mm) vs angle and time (in 10^6 s), for a moment $M = -2000$ Nm. For the original benchmark, we observe a pattern of order 8. The higher moment results in a speed of ca. 320 km/h and only 3 humps

shifted with respect to the radius, hence the wear pattern is not fixed but moves around against rolling direction.
This effect may not be observed for cases where the variation growth too fast. Than lift off takes place before the shift of the pattern becomes significant. Further, a preexisting imperfection, like a flat spot, 'pins' the pattern to the surface until it is levelled out.

We have to point out that for this presentation, we chose a driving wheel. The mentioned effects occur likewise for a wheel coupled elastically to a perfect partner wheel on a wheelset, cf. [4].
In conclude this section by some remarks on randomizing some of the model parameters.

5.1 Randomized Control

Following a hint from a referees, we have performed calculations with randomized data. Among others, we have run a vehicle over randomized ground. For a random geometric error imposed on the vertical position of the rail head, we obtain – in accordance with a lemma from [17] – that the results coincide with those for a slightly changed wear law. For a given out-of-the-

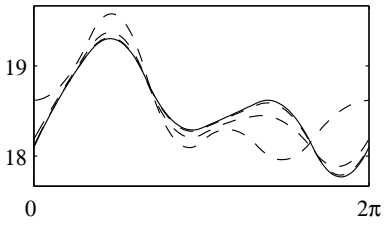

Fig. 14. 1D wear simulation, disk wheel model on wavy ground. The *solid line* shows the power dissipation (W) for an ideal track, the *dotted lines* indicate mean values over increasing rolling times (1,10,100 (s)) on randomized track

round wheel, on perfect ground, we would obtain the creepage distribution indicated in Fig. 14, solid line.
On wavy ground, we obtain a position (= time) dependent creepage. However, the mean value curves converge (with running time going to infinity) to the output of the undisturbed ground case (dotted curves). In this case, we have an *averaging effect* as should be expected on the basis of [17].
Analogously, we introduced a random lateral force f_4 acting on a damped variant of the lateral dynamic (24)

$$m\ddot{q}_4 = -k_4 q_4 - d_4 \dot{q}_4 + f_4(t) \qquad \text{with } d_4 = 514\,\text{Ns/m}. \tag{25}$$

The parameters of the the random force distribution f_4 determine the shape of the final geometry, Fig. 15. Note that the damping has considerable influence on the distribution along the wheel axis.

6 Conclusions

It is as desirable as unrealistic to have the most advanced models of all sub-systems involved in the feedback-loop of system dynamics and long-term behavior, and run them in one single program or package. Instead, we use simplifications where possible, and couple components of other groups via

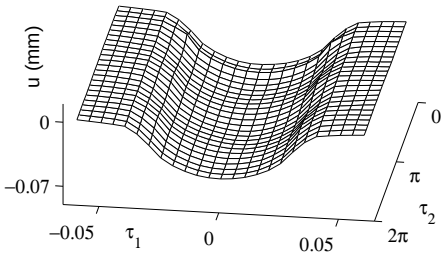

Fig. 15. 2D extension of benchmark problem. The force $f_4(t)$ is a time series of of random impulses with zero mean and a deviation of 3000 N

internet programming. It was shown in Sec. 3 that for studying polygonalisation, it is essential to include vertical dynamics in a sensible way. That way results depend reasonably on data like initial surface deviations and speed of control motion, and allow a consistent interpretation.
From the point of view of long-term changes, models of vehicles, track and subgrade are components that determine the right-hand side of an evolution problem for the domain occupied by the wearing parts. Proper methods of integration for domain evolution problems are as important as fast methods for solving the equations of motion of the dynamical systems. Benchmark problems designed for testing the long-term integration have been formulated. For closing the feedback loop, the dynamical system has to respond to changes of geometry.
In this project, we have mainly studied wear patterns on wheels; patterns on rails have been omitted to keep this paper within the given limits. For the same reason, temperature effects on friction and wear laws have not been discussed, [15]. In the future, roughness, the real contact areas and local forces, and in turn the changes of surface texture, should be modeled and included into the feedback loop.

7 Acknowledgements

We appreciate the great efforts of the creators and coordinators of the Priority Programme 1015. In particular, we enjoyed the very constructive atmosphere and the intensive discussions at the regular meetings of the vehicle group. Finally, we have to thank the reviewers for their time and a couple of really helpful suggestions.

References

1. Arnold, M., Netter, H. (1998) Approximation of contact geometry in the dynamical simulation of wheel-rail systems. Mathematical and Computer Modelling of Dynamical Systems **4**, 162–184

2. Bogacz, R., Frischmuth, K. (2001) Slip waves on curved track. In: Proceedings of the 7^{th} German-Polish Workshop on Dynamical Problems in Mechanical Systems, Osieki
3. Bogacz, R., Kowalska, Z. (2001) Computer simulation of the interaction between a wheel and a corrugated rail. Eur. J. Mech. A/Solids **20**, 673–684
4. Brommundt, E. (1996) A simple mechanism for the polygonalization of railway wheels by wear. Mechanics Research Communications Basic and Applied
5. Chudzikiewicz, A. (2000) Zużycie kół kolejowych. In: Proceedings of the 7^{th} Conference on Computer Simulation in Research and Development, Kościelisko, September 14–26
6. Frischmuth, K. (2000) Wear models with internal state parameters. Machine Dynamics Problems, **24**, No. 1, 79–86
7. Frischmuth, K. (2000) Temporal evolution of wear profiles. TransComp, Zakopane
8. Frischmuth, K. (2001) Contact, motion and wear in railway mechanics. Journal of Theoretical and Applied Mechanics, 3, **39**
9. Hänler, M. (1998) Private communication.
10. Johnson, K. L. (1985) Contact mechanics. University Press, Cambridge
11. Kaas-Petersen, Ch. (1986) Chaos in a Railway Bogie. Acta Mechanica **61**, 89–107
12. Kaiser, I., Popp, K. (2000) The behaviour of a railway bogie in the mid-frequency range. Euromech Colloquium 409, Hannover, March 6-9
13. Kalker, J. J. (1990) Three-Dimensional Elastic Bodies in Rolling Contact. Vol. **2** of Solid Mechanics and its Applications, Kluwer Academic Publisher, Dordrecht
14. Kik, W., Piotrowski, J. (1996) A fast, approximate method to calculate normal load at contact between wheel and rail and creep forces during rolling. In: Zobory, I. (Editor): Proc. of the 2nd MiniConference on Contact Mechanics and Wear of Rail/Wheel Systems, TU Budapest, 52–61
15. Knothe, K. (2001) Probleme der Berechnung von Rollreibungskräften. Vortrag im FK Numerik, Universität Rostock, 7. November
16. Kragelski, I. V. (1965) Friction and Wear. Butterworth Washington DC
17. Langemann, D. (1999) Numerische Analyse abrasiv verschleißender mechanischer Systeme. Fortschritt-Berichte 12.**392**, VDI-Verlag Düsseldorf
18. Langemann, D. (2000) Numerical analysis of wear processes. submitted to SIAM Journal of Applied Mathematics
19. Meinders, T., Meinke, P. (2000) Development of polygonalized wheels and their effects on the dynamical system. Euromech Colloquium 409, Hannover, March 6-9
20. Sethian, J. A. (1996) Level set methods. Cambridge University Press, Cambridge
21. Shen, Z. Y., Hedrick, J. K., Elkins, J. A. (1983) A Comparison of Alternative Creep Force Models for Rail Vehicle Dynamic Analysis. In: The Dynamics of Vehicle on Road and on Track, Proc. of 8th IASVD Symposium held at Massachusets Institute of Technology, Cambridge, MA, August 15–19
22. True, H. (1993) Dynamics of a rolling wheelset. Applied Mechanics Reviews, **46**, No. 7