ELSEVIER

Static Friction, Differential Algebraic Systems and Numerical Stability

Jian Chen[a], Alexander Schinner[b], Hans-Georg Matuttis[c]

[a]*University of Electronic Science and Technology of China, School of Mechatronics Engineering, 610054 Chengdu, P. R. China*
[b]*T-Systems GEI GmbH, Dachauerstr. 651, 80995 München, Germany*
[c]*The University of Electro-Communications, Department of Mechanical Engineering and Intelligent Systems, 182-8585 Tokyo, Japan*

Abstract

We show how Differential Algebraic Systems (Ordinary Differential Equations with algebraic constraints) in mechanics are affected by stability issues and we implement Lubich's projection method to reduce the error to practically zero. Then, we explain how the "numerically exact" implementation for static friction by Differential Algebraic Systems can be stabilized. We conclude by comparing the corresponding steps in the "Contact mechanics" introduced by Moreau. © 2010 Published by Elsevier Ltd.

Keywords: exact implementation of Coulomb friction, differential algebraic equations, granular many particle systems, stabilization by projection

1. Introduction

One of the most tedious problems in computational mechanics is the drift of the solution away from "hard constraints", e.g. a pendulum simulated as a two-dimensional constraint system de- or increasing its length. The various flavors of non-smooth mechanics[1, 2, 3, 4, 5, 6] also belong to this class of problems, though the occurring numerical drift and what can be done about it is hardly ever discussed outside the numerical analysis community. We will explain how the projection method by Lubich[7] can overcome such instabilities in the "numerically exact" sense, show how static friction can be formulated as a constraint problem and how the "numerically exact" treatment of such static Coulomb friction constraints can be achieved.

2. Constraints in Mechanics

Newton's equation of motion $F = ma = m\ddot{x}$ can be represented as a system of coupled ordinary differential equations (ODEs) of first order

$$\dot{v} = F/m \qquad (1)$$
$$\dot{x} = v. \qquad (2)$$

With $y = [v, x]^T$ and $f(y, t) = [F/m, v]^T$ we get

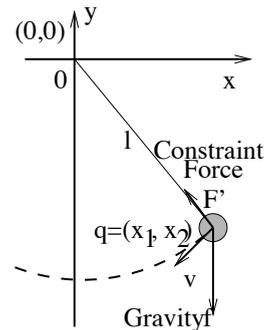$$\dot{y} = f(y, t), \qquad (3)$$

Figure 1: Pendulum as constraint system.

which is (apart from the use of prime instead of dot) the way to represent ODEs in numerical analysis textbooks. All variables for the velocities v, positions x and the "right-hand-side" f in the following are used in the sense of vectors.

Constraint systems in mechanics[1, 8] which in the field of numerical analysis are usually referred to as Differential Algebraic Systems (DAEs), are not only governed by Newton's equation of motion, $\vec{F} = m\vec{a}$, but additionally by equations for geometrical constraints $g(q)$, which act on a subset q of the coordinates and velocities y, and which e.g. fix the distance between rigid bodies. A typical example is the pendulum with point mass $m = 1$ in Fig. 1, where the particle is constrained along a circular trajectory. The geometric constraint for a pendulum of unit length which swings around the origin (see Fig. 1) is given by

$$g(q) = x_1^2 + x_2^2 - l = 0, \tag{4}$$

on the coordinates $q = [x_1; x_2]$, and we will set $l = 1$, which means that q is on a circular orbit with unit radius. Differentiating eq. (4) with respect to time yields

$$\dot{g}(q) = q^T \cdot \dot{q} = x^T \cdot v = 0. \tag{5}$$

By differentiating eq. (5), the constraint equation for the acceleration becomes

$$\ddot{g}(q) = \dot{q}^T \cdot \dot{q} + q^T \cdot \ddot{q}^T = 0. \tag{6}$$

Substituting $\ddot{q}$ in eq. (6) by $\ddot{q} = \frac{F + F'^T}{m}$, with gravity F and constraint force F', the equation is simplified to first order

$$\ddot{g}(q) = \frac{(F + F')^T}{m} q + \dot{q}^T \cdot \dot{q} = 0, \tag{7}$$

where the constraint force F' is still unknown. In order to solve for the acceleration $\ddot{g}(q)$, we need an additional condition for the unknown constraint force F', which has to satisfy eq. (7) and the scalar equation

$$F'^T \cdot q = -F^T \cdot x - m\dot{x}^T \cdot \dot{x}, \tag{8}$$

(a) Trajectory of the single pendulum as constraint system

(b) Numerical "drift" in the radius.

Figure 2: Motion and radius of the single pendulum with timestep for CRK with $dt = 0.1$.

as well as the principle of virtual work, e.g. the constraint force F' does not change the energy of the system, which means that the kinetic energy

$$T = \frac{m}{2}\dot{q}^T \cdot \dot{q} \tag{9}$$

has a time derivative which is is independent of F' :

$$\dot{T} = m\ddot{q}^T \cdot \dot{x} = mF^T \cdot \dot{q} + mF'^T \cdot \dot{q} = 0. \tag{10}$$

Due to eq. (5) F' is orthogonal to $\dot{q}$ and parallel to q, so that

$$F' = \lambda x \tag{11}$$

with Lagrange multiplier λ. Actually, eq. (11) could have been inferred already on physical argument, but the more formal derivation helps to understand the more complicated examples in the later sections. Actually, it is also essential to chose the set of initial conditions $x_1(0), x_2(0), v_1(0), v_2(0)$ "consistently", which means for our case so, that the coordinates are orthogonal to the velocities, $[x_1(0), x_2(0)] \cdot [v_1(0), v_2(0)]^T = 0$, else the solutions diverge exponentially. Substituting eq. (11) into eq. (8) gives
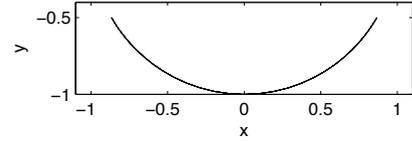
$$\lambda = \frac{-F^T \cdot x - m\dot{x}^T \cdot \dot{x}}{x^T \cdot x}. \tag{12}$$

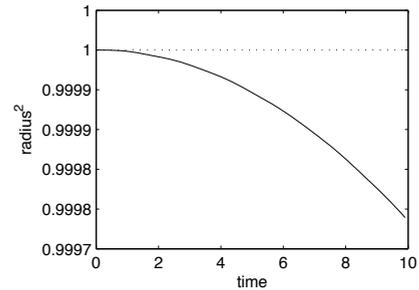The pendulum is then described by the following coupled differential equations

$$\dot{x} \quad = \quad v \tag{13}$$

$$\dot{v} \quad = \quad \frac{F + F'(\lambda)}{m}, \tag{14}$$

of first order, where F' is given by eq. (11) and the constraint value of λ is given by eq. (12). For detailed discussion on the pendulum and constraint systems with more degrees of freedom see Witkin[8]. Because it is inconvenient to apply the principle of virtual work for mechanical systems with parts moving on arbitrary trajectories, it is better to

formulate the general equations for constrained mechanic systems as

$$\dot{q} = v \tag{15}$$

$$M(q)\dot{v} = f(q, v) - G^T(q)\lambda \tag{16}$$

$$0 = g(q) \tag{17}$$

(see Hairer[9], *p.*464) where q is the vector of the coordinates, $g(q)$ is the set of geometric constraints and $G(q) = \frac{\partial g}{\partial q}$ its gradient with respect to the particle coordinates. Differentiating the algebraic equation for $g(q) = 0$ gives

$$\begin{pmatrix} M(q) & G^T(q) \\ G(q) & 0 \end{pmatrix} \begin{pmatrix} \dot{v} \\ \lambda \end{pmatrix} = \begin{pmatrix} f(q, v) \\ -g_{qq}(q)(v, v) \end{pmatrix} \tag{18}$$

as equation for the Lagrange multipliers λ. This kind of equations has very universal application possibilities. For example, the Finite-Element-Formulation of the incompressible Navier-Stokes equation can be rewritten in such a form, in which case Lagrange-Parameters turn out to be the pressures in each element[10]. Solving λ from this system of linear equations, the equation of the system becomes a system of first order differential equations which can be solved numerically:

$$\dot{q} = v \tag{19}$$

$$M(q)\dot{v} = f(q, v) - G^T(q)\lambda \tag{20}$$
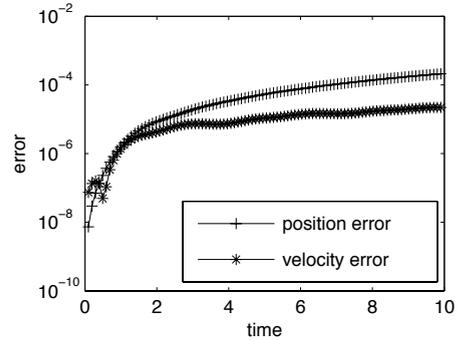
2.1. Numerical drift in constraint systems

When we compute the trajectory with the classical Runge-Kutta method (for a, as we hope, readable account of the zoo of numerical methods used in numerically solving ODEs see Matuttis et al.[11]) of fourth order in Fig. 2(a), the radius of the single pendulum decays in Fig. 2(b) within about 3 periods, the

(a) Direct implementation.

(b) With projection

Figure 3: Absolute errors for position and velocity for the pendulum in the direct implementation in (a) and after the implementation of the projection method in (b) (for the missing points the error is exactly 0).

pendulum spirals inward. The problem is not the size of the error itself, which could be reduced by the choice of a smaller timestep, but its exponential increase, so that for sufficient long times, the solution will always be inaccurate. This numerical drift occurs due to the finite time-step. The reasoning for coordinates also applies to velocities: The weighted average of velocities tangential to a circle at different points is not necessarily tangential itself. The position error $\varepsilon_p = |x^2 + y^2 - l^2|$ and the velocity error $\varepsilon_v = [x \, y] \cdot [v_x \, v_y]^T$ are drawn in Fig. 3(a).

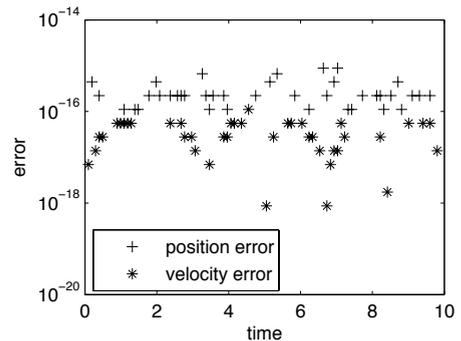### 2.2. Stabilization of the Drift

The numeric errors should be removed in some way. Historically, the most widely used stabilization to annihilate the drift away from the constraint is due to Baumgarte[12], who arranged the constraint C and its time derivatives $\dot{C}$ and $\ddot{C}$ formally as a harmonic oscillator equation

$$\ddot{C} = -k_s C - k_d \dot{C}. \tag{21}$$

The (unphysical) constants k_s, k_d can be chosen[13, 14, 15] so that the resulting ODE-system of the original problem and the harmonic oscillator in eq. (21) becomes stiff. That means that the time-scale of the constraint is chosen much higher than the time-scale of the original problems, which leads to a "stiff" ordinary differential equation, and for the solution "stiff solvers" like Gear-Predictor-Corrector[16]. Instead of Baumgarte-stabilization, we use the parameter-free "stabilization by projection" due to Lubich[7], where the numerical deviated solution is being "projected" (in the sense of the projection of a vector onto another one) back onto the correct solution manifold, in the case of one-step methods after the weighted average, e.g. for the classical Runge-Kutta method of fourth order.

Simply illustrated for the pendulum, assume that approximate values (x_i, y_i, vx_i, vy_i) at time t_i satisfy the geometric constraints, which means the errors ε_p and ε_v, equal zero. At the next timestep t_{i+1}, we get approximate values $(\widehat{x}_{i+1}, \widehat{y}_{i+1}, \widehat{vx}_{i+1}, \widehat{vy}_{i+1})$. Due to the numerical error, the geometric constraints are not satisfied anymore: $\varepsilon_p = \hat{x}_{i+1}^2 + \hat{y}_{i+1}^2 - 1 \neq 0$ and $\varepsilon_v = \hat{x}_{i+1} \cdot \widehat{vx}_{i+1} + \hat{y}_{i+1} \cdot \widehat{vy}_{i+1} \neq 0$. Since the the length of the rope is a constant constraint, the position projection can be achieved easily as

$$(x_{i+1}, y_{i+1}) = \frac{(\hat{x}_{i+1}, \hat{y}_{i+1})}{|\hat{x}_{i+1}, \hat{y}_{i+1}|} \cdot l.$$

When using a projection of coordinates, the function F in eq. (2) in general is a complicated nonlinear function of the coordinates, and the numerical solution (in common use for the RATTLE- and the and SHAKE-algorithm[17]) is quite inconvenient for e.g. non-linear molecules. It turns out that no nonlinear-system must be solved. In Fig. 3(a), the position error seems to be larger than the velocity error, but a short fuzzy error analysis convinces one of the the the opposite: from eq. (12), we have $\dot{x} = v$, so for their respective errors $\Delta x, \Delta v$, we can assume $\Delta \ddot{x} \approx \Delta v$. For a finite timestep Δ, we obtain in highest order $\Delta t \Delta x \approx \Delta v$, so the error $\Delta x = \Delta v / \Delta t$ is nothing else than the velocity error, magnified by division by the timestep. This tells us that choosing a very small timestep may not always be favorable and that we can reduce the error if we use the projection on the velocity alone. For the satisfaction of the velocity constraint, (vx_{i+1}, vy_{i+1}) can be projected $(\widehat{vx}_{i+1}, \widehat{vy}_{i+1})$ back to the real velocity direction, where the velocity vector is orthogonal to the position vector. From Fig. 3(b) it can be seen that the errors are reduced to numerical irrelevance. General formulae can be found in Ref.[9] for position projection, $p. 471$ and Ref.[7].

$$M(\widetilde{q}_n)(q_n - \widetilde{q}_n) + G^T(\widetilde{q}_n)\mu = 0 \tag{22}$$
$$g(q_n) = 0 \tag{23}$$

and for velocity projection,

$$M(q_n)(u_n - \widetilde{u}_n) + G^T(q_n)\mu = 0 \tag{24}$$
$$G(q_n)u_n = 0 \tag{25}$$

We have also performed numerical experiments for the double pendulum (not shown due to the lack of space) with the direct implementation of Newton's equations of motions and stabilization and compared them to an implementation of the Hamiltonian formulation. Lamentably, even with stabilization, the accuracy was considerably worse than for the Hamiltonian formulation, which is not available for problems with Coulomb friction. In that case, velocity stabilization is the best way to reduce the computational cost and improve the numerical accuracy.

3. Modeling Friction via DAEs

The constraint system in the first two sections results in Differential Algebraic Equations with bilateral constraints ("="), which could be solved by transformation to normal coordinates as an ODE, without the stability issues discussed, so one might ask oneself: "So what?". In this section it will turn out that static friction leads to DAEs with unilateral constraints ("≤") which cannot be solved via transformation techniques, so the tedious direct solution must be sought.

3.1. The physical character of friction

Coulomb/ dry friction between surfaces of solids can either exist as static or as dynamic friction, which are hugely different in character. Mathematically, dynamic friction is a function,

$$F_{\text{fric}} = -\mu F_{\text{norm}} \frac{v}{|v|}, \tag{26}$$

while static friction is described

$$-\mu F_{\text{norm}} \leq F_{\text{fric}} \leq \mu F_{\text{norm}}. \tag{27}$$
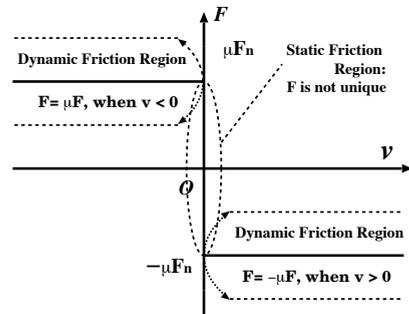
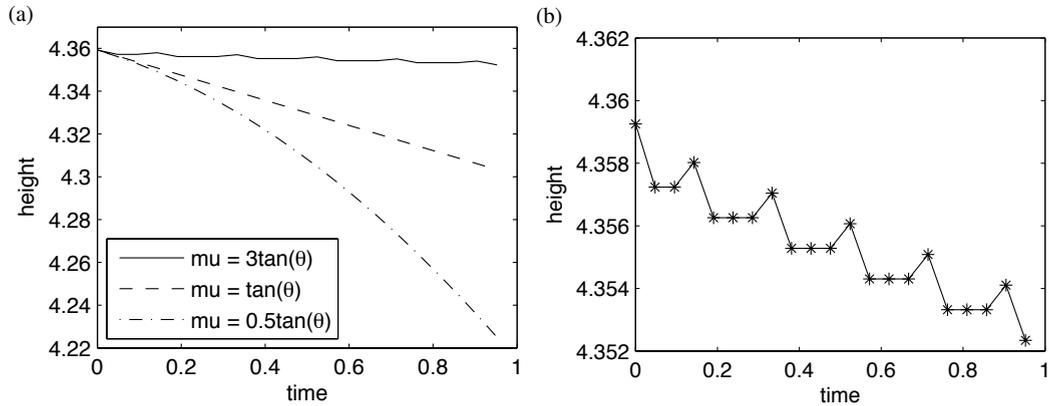

Figure 4: Friction regimes.

Figure 5: Effect of three different friction coefficients, above, below and at the critical angle on the trajectories in for implementation of dynamic friction for the static case with a classical Runge-Kutta-method of fourth order with $dt = 0.1$ in (a) and magnification of the unphysical trajectory for $\mu = 3\tan(\theta)$ in (b).

This relation is often called a unilateral constraint (with a "≤" instead of a "="). Physically, the dynamic friction dissipates energy during the relative motion of the surfaces, in contrast to static friction, which constrains the surfaces relative to each other without any energy loss. As shown in the Fig. 4, when the velocity $v \neq 0$, the value of friction is uniquely determined by eq. (26), however when the *velocity* = 0, the friction force could have any value within the bounds of the inequality (27), but is in fact unique, when it compensates all other external forces so that $v = 0$. In the following, we use a single friction coefficient μ for both static and dynamic friction, the equations in the following can easily be generalized for different coefficients. Though orthodox physics claims that the static friction coefficient should be larger then the dynamic friction, newer engineering books give only a single friction coefficient for a wide range of materials (e.g. ceramics[18, 19]). Where different coefficients are given[20], these stem from different references, so that the equivalence of the material and laboratory conditions seems doubtful, and tables in different books usually contradict each other (e.g. static friction for steel on steel in Kuchling p. 584 $\mu = 0.12$, in Bhushan and Gupta section 2.11 $\mu = 0.7$ . . . . For Polymers, in general, the static friction coefficient it even smaller then the dynamic friction coefficient[21]. For the problem of reliably tabulation friction coefficients, see Rabinowicz[22].

3.2. Unsatisfying Approaches

One naive way is to neglect the difference between static and dynamic friction and use eq. (26) also for the computation of static friction. For a particle on an inclined slope as shown in Fig. 6, where θ is the critical angle, the results in Fig. 5 indicate that when the friction coefficient $\mu \leq \tan(\theta)$, the numerical solution make sense physically, however, when $\mu > \tan(\theta)$, the solution is physically meaningless. Another way[23] is to model is to increment the tangential force F_i^t (zero at the beginning of the contact) between contacting particles in every timestep dt

Figure 6: Possible cased for friction in the block on a slope.

proportional to their tangential velocity v_t and a "tangential spring constant" k_t,

$$\tilde{F}_i^t = F_{i-1}^t - k_t v_t dt, \tag{28}$$

and to truncate this force if the maximal tangential force exceeds the dynamic friction force μF_i^t for a normal force F_i^n, according to

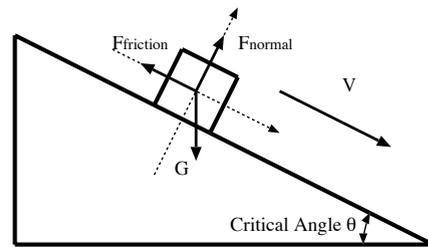$$F_i^t = \min(|\tilde{F}_i^t|, |\mu F_i^n|) \cdot \frac{\tilde{F}_i^t}{|\tilde{F}_i^t|}. \tag{29}$$

Below the threshold eq. (29), equation (28) is the time-derivative of the equation for the undamped harmonic oscillator $dF/dt = -kdx/dt$, so a particle on an inclined plane does not come to rest but continues to swing around the contact point. For the sake of verisimilitude, a second term e.g. with $\tilde{k}_t v_t$ has to be included which damps out these unphysical oscillation. Though the constants $k_t, \tilde{k}_t$ have a physically meaningful interpretation as the onset speed of friction, such an approach is unsatisfying in any application other than the statistical physics of granular media[24], because it does not suffer from the numerical instabilities we will discuss later in the text and the many-body interactions annihilate the noise fluctuations and the simulation results become independent of $k_t, \tilde{k}_t$ for reasonable parameter choices. Nevertheless, an approach which increases the degrees of freedom in a strongly vibrated medium may lead to unforeseen results if the fluctuations arbitrarily load or discharge energy from the "springs" in eq. (28).
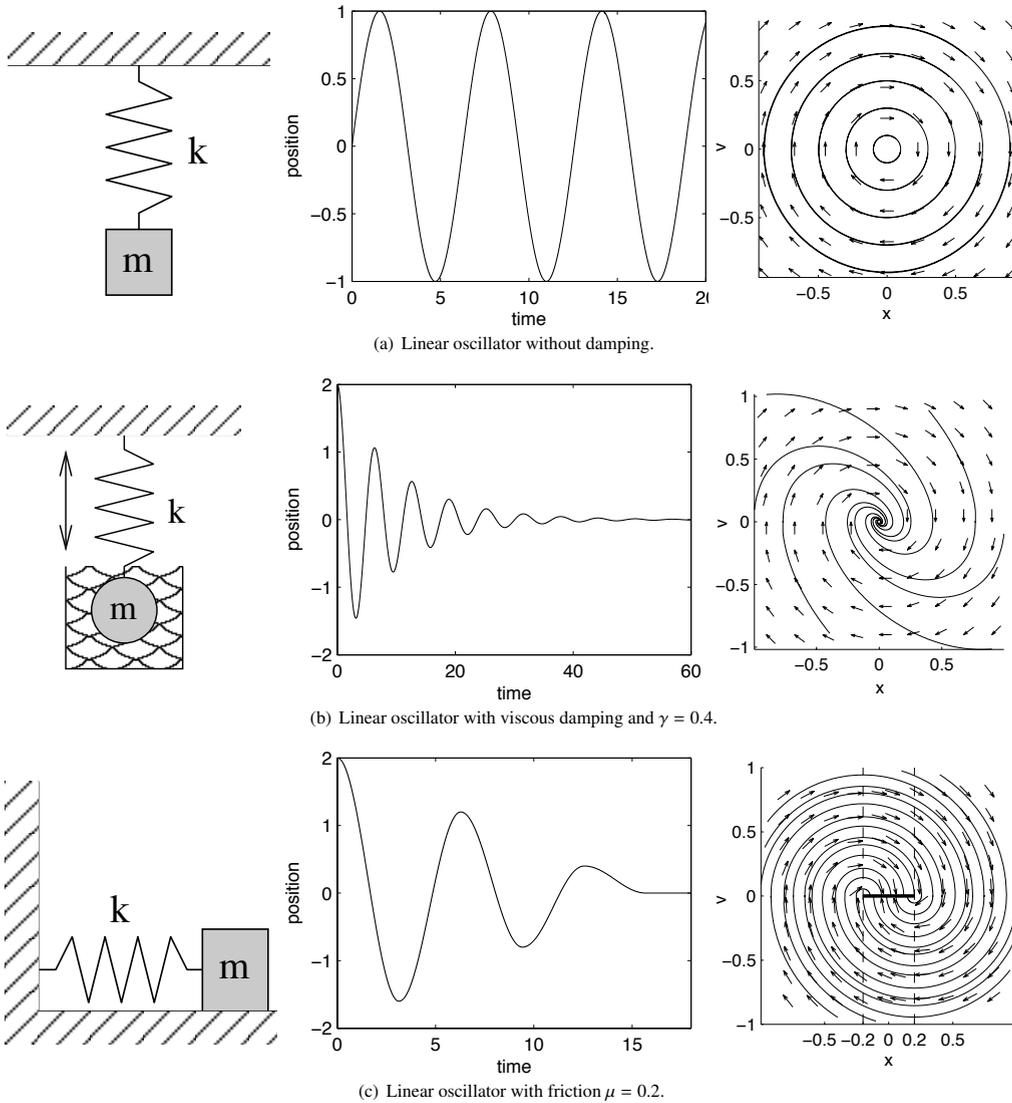
(a) Linear oscillator without damping.

(b) Linear oscillator with viscous damping and $\gamma = 0.4$.

(c) Linear oscillator with friction $\mu = 0.2$.

Figure 7: Linear oscillator without and with damping.

3.3. Static friction in phase-space

To implement friction in "many-body-dynamics"[25, 26, 27] "numerically exact" two fundamental problems must be solved. The first is how to detect whether the fiction is dynamic or static (velocity reversal is not sufficient, as we will see later), and if the condition for static friction is fulfilled, the second problem is to find the physically unique value for static friction. To define the condition for static friction, it is revealing to compare the portrait of the phase-space (the plot of the elongation versus the velocity, for different initial conditions) for a simple harmonic oscillator with and without static friction (for convenience, with unit mass $m = 1$). For the free harmonic oscillator

$$\ddot{x} = -kx, \quad k = 1, \tag{30}$$

in Fig. 7(a), the trajectories in phase-space are ellipses. Actually, for this energy-conserving ("symplectic") system it would be better to use a symplectic integrator, i.e. one for which the deviation of the energy from the exact solution is bounded, like e.g. the Störmer[28]/Verlet[29] of second order, or related higher order derivatives[30], which can be derived by the Suzuki-Trotter-decomposition[31] (for an overview on symplectic methods, see Hairer[32]). For the harmonic oscillator with viscous damping

$$\ddot{x} = -kx - \gamma v, k = 1, \gamma = 0.8 \tag{31}$$

in Fig. 7(b), the trajectories in phase space approach the origin in exponentially tightening spirals. For the harmonic oscillator damped by dry/ Coulomb friction

$$\ddot{x} = -kx - \mu \frac{v}{|v|}, k = 1, \mu = 0.2 \tag{32}$$

the trajectories don't spiral into a single fix point, but into a "fix line" between the points $(x, v) = (\mu/k, 0)$ and $(x, v) = (-\mu/k, 0)$ (thick line in Fig. 7(c), right.). This fix-line can be detected by the sign of the parameter $a = -\text{sign}(v)x - \mu$, which changes sign for dynamic dry friction (beyond $x < -\mu/k$ and $x > \mu/k$) when the line at $v = 0$ is crossed, but not in the case of static friction, where for $-\mu/k \le x \le \mu/k$ it is negative (Fig. 8). Above and below the fix line, which corresponds to static friction, the flow of the ODE (direction of the arrows in Fig. 7(c), right) points towards the line, while for dynamic friction, the flow crosses the axis at $v = 0$ without a change of the direction. Therefore, the task for identifying static friction lies in a proper generalization of a, which we will show in the next section.

3.4. Static Friction as a Constraint

As seen above, static friction, a constraint of motion, can not be treated as dynamical friction, an energy dissipation mechanism. Since we can not get an unique value from the inequality relation eq. (27) and can not substitute eq. (26), we have to determine the value of the static friction from the physical constraint, as we done before in the pendulum case. The physical meaning of "static friction" is in fact a constraint which holds two bodies at rest with respect to each other. In the following, we follow the elaboration of the formalism according to Hairer[9], *p*.464. First we separate the phase-space of the differential algebraic equation into the one with positive and the one with negative velocities according to the switching function $g(y)$, so that we have the two possible forces

$$\dot{y} = \begin{cases} f_I(y) = (F_{ext.} - \mu F_{nor.})/m & \text{if} \quad g(y) > 0 \\ f_{II}(y) = (F_{ext.} + \mu F_{nor.})/m & \text{if} \quad g(y) < 0 \end{cases} \tag{33}$$

where y represent the velocity, and the switching-function $g(y) = \text{sign}(y)$ indicates the direction of velocity. f is discontinuous on the surface $S = \{y; g(y) = 0\}$. We now look for the "true" value for $f(y)$ for $g(y) = 0$ in the "convex hull" of f_I and f_{II}, which is just the linear combination

$$\dot{y} = f(y, \lambda) = (1 - \lambda)f_I(y) + \lambda f_{II}(y), \tag{34}$$

of f_I and f_{II}, see Ref. [25], *p*.199. The value of λ is determined by $g(y) = 0$.

By differentiating $g(y) = 0$, we obtain:

$$\nabla g(y)\dot{y} = \nabla g(y)f(y,\lambda). \qquad (35)$$

Setting

$$a_I = \langle \nabla g(y), f_I(y) \rangle \qquad (36)$$
$$a_{II} = -\langle \nabla g(y), f_{II}(y) \rangle, \qquad (37)$$

where the gradient ∇ in $1D$ reduces to ± 1. Then eq. (35) becomes $(1 - \lambda)a_I(y) - \lambda a_{II}(y) = 0$, the value of λ is obtained by

$$\lambda = \frac{a_I(y)}{a_I(y) + a_{II}(y)}, \qquad (38)$$

which means nothing else that the highly nonlinear, non-smooth problem for the jump in the friction in Fig. 4 has a solution which can be computed from

Figure 8: Sign of the parameter $a = -\text{sign}(v)x - \mu$ for the harmonic oscillator with static friction $\mu = 0.2$.

equations linear in λ, no Newton-Raphson-Procedure or other iterative computation is necessary. Since the static friction can be solved uniquely by a geometrical constraint, which as the kinetic energy of the contacting bodies with respect to each other, the remaining problem is to find a suitable strategy to detect whether the fiction is dynamic or static. a_I and a_{II} is being using as the criterion in our detection strategy. In the friction case

$$a_I = (F_{external} - \mu F_{\text{norm}})/m \qquad (39)$$
$$a_{II} = -(F_{external} + \mu F_{\text{norm}})/m \qquad (40)$$

(Ref. [25] $p.196 - p.200$) for the acceleration in positive direction a_I and in opposite direction a_{II} we have the following four possibilities:

1. $a_I > 0$, $a_{II} < 0$: the acceleration of the system is in the positive direction, the integration process (to calculate velocity) follows a_I so there is velocity reversal for the dynamic friction, no static friction.
2. $a_I < 0$, $a_{II} > 0$: the acceleration of system is in the negative direction, the integration process follows the a_{II}, dynamic friction only.
3. $a_I < 0$, $a_{II} < 0$: the acceleration of system is neither positive nor negative, which means 0, static friction domain.
4. $a_I > 0$, $a_{II} > 0$: the acceleration of system is in both directions. This is physically meaningless and may happen due to unphysical chosen initial conditions, which in analytical mechanics occur in systems associated with "Painleve paradoxes".

Results of the implementation of the projection method in the "block-slope" case can be seen in Fig. 9. Because the switching function $g(y)$ is also a constraint, if $g(y) = 0$ is fulfilled, we set the velocity identical to 0, which is the reason why the one-dimensional problem does not suffer from numerical instabilities. For higher dimension, as switching function and constraint $g(y)$, the relative velocity between the mechanical constituents will be used.

3.5. Many-body problem and two dimensions

We treat the many-body problem with friction for the cannonball-stacking in two dimensions (Fig. 10) neglecting the rotation of the particles. The particles are "soft", which means that the elastic deformation is taken into consideration via the use of a Young modulus. In this example, we use "linear potentials" (though Hertzian contacts are more physical, see Wolf[33] for a more elaborate discussion). The standard-potentials between grains are usually tailored according to an oscillator equation

$$F = -\alpha_1 \xi_1^{\beta} - \alpha_2 \dot{\xi}_2^{\beta}, \qquad (41)$$

where ξ is the overlap, $\dot{\xi}$ is the change of the overlap and $\alpha_1, \alpha_2, \beta_1, \beta_1$ are chosen by the researchers discretion. Irrespective of the choice of constants, the time dependence for an idealized collision of duration T_c will look more or less like the one in Fig. 11, so the damping-force introduces jumps into the force progression.
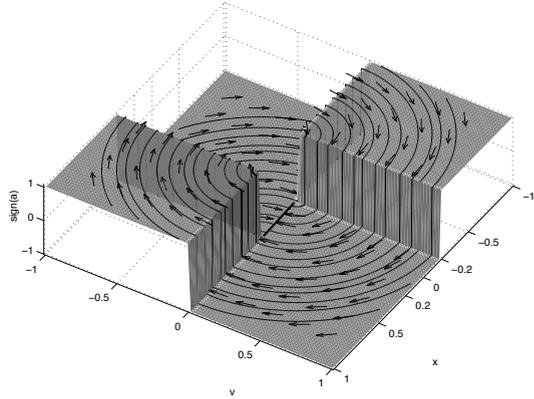
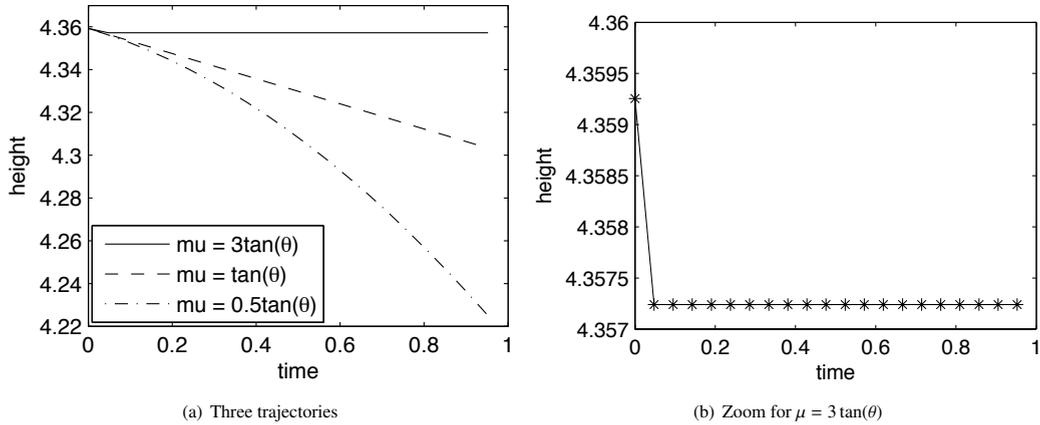(a) Three trajectories

(b) Zoom for $\mu = 3\tan(\theta)$

Figure 9: Block on a slope simulated with the DAE-formalism eq. (38) distinguishing dynamic and static friction, below the critical angle ($\mu = 3\tan(\theta)$), the body is constrained by static friction.
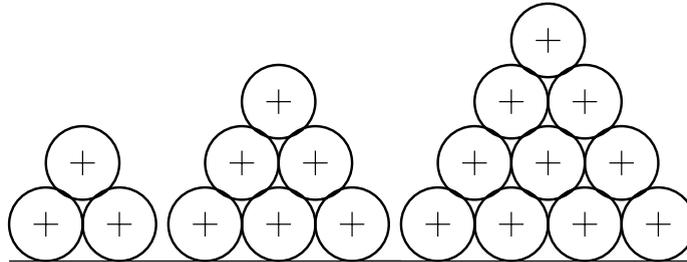

Figure 10: Cannonball-stacking in two dimensions for three, six and ten cannon-balls as an example of many-body problem.

Though the cohesive part (shading in Fig. 11) can be eliminated by appropriate if-conditions[24], the jump at the initial closing of the contact leads to numerical instabilities for explicit integrators (built from a Taylor-series of the force progression), e.g. explicit Runge-Kutta/one-step methods, and the Verlet-family (Verlet, Velocity-Verlet, Leapfrog)[17]. Therefore, we will use an implicit multi-step method (BDF/Gear predictor corrector of 5th order[16, 17].) in this section. The stabilization-step is inserted after the correction step. The usage of the projection formalism in multi-step methods is in principle mathematically questionable: The conditions following eqs. (39,40) assume that the sign of $a_I, a_I I$ does not change during a timestep, which can be fulfilled by reducing the timestep for one-step methods. For multi-step methods which resort to the use of values from former timesteps, it is in principle possible that each iteration step has a different kinetic condition (change from static to dynamic friction and back). In our experience, for actual physical problems concerned this has not been a limitation.

Figure 11: Typical time-dependence of inter-granular forces tailored according to the oscillator-equation.

3.6. Contact matrix

For changing particle contacts, the dimensions of the $G(q)$ matrix in eq. (20), change during the integration time steps. When a_I and a_{II} from eq. (37,36) are used as the criteria to check whether the friction is in the dynamic or static region, the gradient ∇ in 2D is not ± 1 anymore, but for n contacts, see Fig. 12, become (in two dimensions) sparse matrices with $2n$ entries:
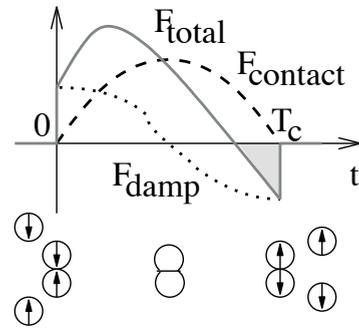
$$G_1(q) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \sin(\theta_{12}) & -\cos(\theta_{12}) & -\sin(\theta_{12}) & \cos(\theta_{12}) & 0 & 0 \\ \sin(\theta_{13}) & -\cos(\theta_{13}) & 0 & 0 & -\sin(\theta_{13}) & \cos(\theta_{13}) \end{pmatrix}, \tag{42}$$

$$G_2(q) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \sin(\theta_{12}) & -\cos(\theta_{12}) & -\sin(\theta_{12}) & \cos(\theta_{12}) & 0 & 0 \\ \sin(\theta_{13}) & -\cos(\theta_{13}) & 0 & 0 & -\sin(\theta_{13}) & \cos(\theta_{13}) \\ 0 & 0 & \sin(\theta_{23}) & -\cos(\theta_{23}) & -\sin(\theta_{23}) & \cos(\theta_{23}) \end{pmatrix}. \tag{43}$$

Nevertheless, when the method is implemented in a straightforward way, the decay of the kinetic energy in Fig. 13(a) shows that that the particles don't "stick": The constraint is only fulfilled up to the error order of the time integrator. The relative velocities are not zero, but e.g. for a $O(tau^2)$ method, they are $O(tau^3)$ etc. Nevertheless, projecting the solution onto the exact constraints with the stabilization method by Lubich[7] leads to a perfectly static configuration for the many particle case Fig. 13(b). as it did for the single particle case.
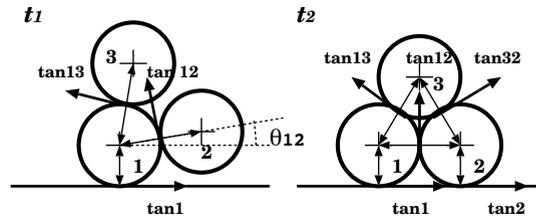

Figure 12: Contact situation for the stacking of three cannon balls, the one on the left described by eq. (42), the one on the right described by eq. (43)

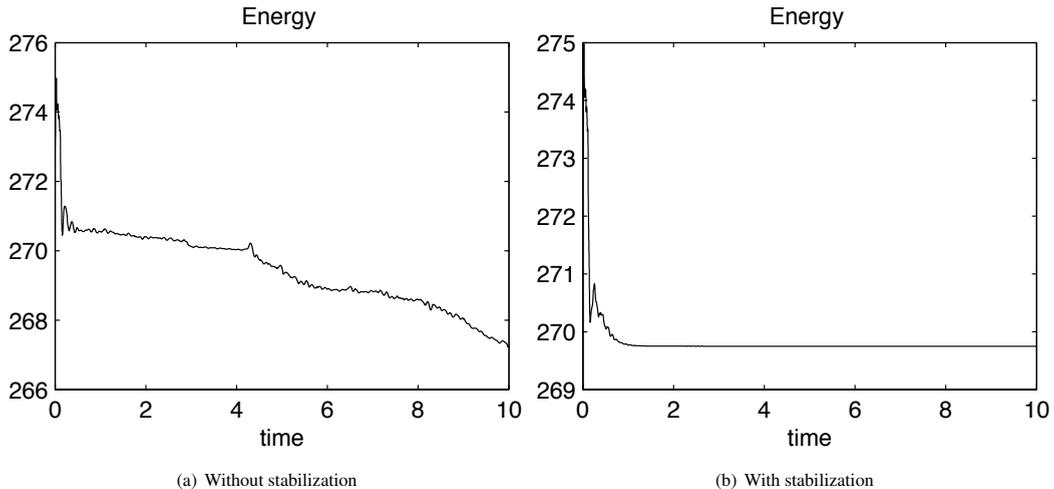(a) Without stabilization

(b) With stabilization

Figure 13: Typical energy for a stacking of ten cannonballs ($\mu = 0.5$) with friction modeled with DAE without (left) and with stabilization by projection (right).

4. Summary and perspectives

We have shown how the DAE-formalism can be used to model friction between elastic particles, so that the friction force is reacting to the already existent elastic forces between the particles. A "straightforward" numerical implementation will not lead to resting contacts, but a "numerical stabilization" is necessary. In this approach, the modeling of the elastic particles is clearly separated from the numerics: a variety of integrators can be used, and stabilization can be included. In the contact mechanics approach by Moreau for rigid particles, the "sweeping process" does not separate the computation of the tangential and normal forces any more, but the ideas outlines in this paper

are the same: the solution of a DAE instead of ODEs with model force laws. The event-driven collision dynamics can be implemented in a soft-particle simulation if Runge-Kutta-integrators are used which can be stopped and restarted at the particle collision times via the technique of "dense output"[25]. Because of this "smooth" transition between DEM, contact mechanics and the event-driven method, it is not necessary to present the methods as totally different entities like in Wolf[33]. When comparing the relative merits of "contact mechanics" of rigid bodies and elastic particle simulation, is should not be forgotten that the sound velocity for rigid bodies is infinite, no matter what the contact situation is; this may be inconvenient to model experimental results which depend on the sound velocity and contact situation, like in Shourbagy et al.[34].

[1] Jens Wittenburg. *Dynamics of Systems of rigid bodies*. Teubner, 1977.
[2] F. Pfeiffer and C. Glocker. *Multibody Dynamics with unilateral contacts*. Wiley Interscience, New York, 1997.
[3] Edda Eich-Soellner and Claus Führer. *Numerical Methods in Multibody Dynamics*. B.G. Teubner, 1998.
[4] J. J. Moreau. *Unilateral contact and dry friction in finite freedom dynamics*, pages 1–82. Number 302 in CISM Courses and Lectures. Springer, 1988.
[5] J. J. Moreau. An expression of classical dynamics. *Ann.Inst.H.Poincaré Anal.Non Lin'eaire*, XXX(6):1–48, 1989.
[6] B. Brogliato, editor. *Impacts in Mechanical Systems. Analysis and Modelling*, volume 551 of *Lecture Notes in Physics*. Springer, 2000.
[7] C. Lubich. *On projected Runge-Kutta methods for differential-algebraic equations*. BIT, 1991.
[8] A. Witkin. Constrained dynamics. In Michael Kass, editor, *SIGGRAPH 93: An Introduction to physical Modeling*, pages G1–G12. International Conference on Computer Graphics and Interactive Techniques, 1993. electronic version recently accessible under `http://www.cs.cmu.edu/~baraff/pbm/particles.pdf`
[9] E. Hairer and G. Wanner. *Solving Ordinary Differential Equation I: NonStiff Problems Second Revised Edition*. Springer, 1996.
[10] P. M. Gresho and R. L. Sani. *Incompressible Flow and The Finite Elemeny Method, Isothermal Laminar Flow* vol 2. John Wiley & Sons Ltd, 2000.
[11] H.-G. Matuttis and J. Chen. Numerical Solution of Ordinary Differential Equations for Modeling in Socio-Econo-Physics, a friendly introduction, in: *Lectures on Socio- and Econophysics*, eds. J. Schneider and C. Hirtreiter, Springer, 2008.
[12] J. Baumgarte. *Stabilization of constraints and integrals of motion in dynamical systems*. Comp. Meth. Appl. Mech. Eng., 1972.
[13] R. Barzel and A. H. Barr. A modeling system based on dynamic constraints. In *Computer Graphics*, pages 179–188. ACM SIGGRAPH, 1988.
[14] J. Platt and A. Barr. Constraint methods for flexible models. In *Computer Graphics*, pages 279–288. ACM SIGGRAPH, 1988.
[15] A. Witkin, M. Gleicher, and W. Welch. Interactive dynamics. In *Computer Graphics*, pages 11–21. ACM SIGGRAPH, 1990.
[16] C. W. Gear. *Numerical initial value problems in ordinary differential equations*. Prentice-Hall, Englewood Cliffs, NJ, 1971.
[17] M. P. Allen, and D. J. Tildesley, *Computer Simulation of Liquids*. Oxford University Press, (1989)
[18] E. R. Booser, editor. *Tribology Data Handbook*, pages 551–560. CRC, 1997.
[19] M. F. Ashby and D. R. H. Jones. *Engineering Materials*. Pergamon Press, 1997.
[20] E. R. Booser, editor. *CRC Handbook of Lubrication*, volume II, pages 46–47. CRC, 1997.
[21] E. R. Booser, editor. *Tribology Data Handbook*, chapter 52, pages 436–436. CRC, 1997.
[22] Rabinowics. *Fraiction and Wear of Materials*. Wiley Interscience, 2.nd edition, 1995.
[23] P. A. Cundall and O. D. L. Strack. A discrete numerical model for granular assemblies. *Géotechnique*, 29(1):47–65, 1979.
[24] H.-G. Matuttis. Simulations of the pressure distribution under a two dimensional heap of polygonal particles. *Granular Matter*, 1(2):83–91, 1998.
[25] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equation II: Stiff and Differential Algebraic Problems Second Revised Edition*. Springer, 1996.
[26] F. Pfeiffer and C. Glocker. *Multibody Dynamics with Unilateral Contacts*. John Wiley & Sons, Inc., 1996.
[27] E. Eich-Soellner and C. Führer. *Numerical Methods in Multibody Dynamics*. B.G. Teubner Stuttgart, 1998.
[28] C. Störmer. *Méthodes d'intégration numérique des équations différentielles ordinaires*. In *C. R. congr. intern. math.*, pages 243–257, Strasbourg, 1921.
[29] Loup Verlet. Computer "experiments" on classical fluids. I. Thermodynamical Properties of Lennard-Jones molecules. *Phys. Rev.*, 159(1):98–103, 1967.
[30] S. K. Gray, D. W. Noid, and B. G. Sumpter. Symplectic integrators for large scale molecular dynamics simulations: A comparison of several explicit methods. *J. Chem. Phys.*, 101(5):4062–4072, 1994.
[31] M.Suzuki: Comm. Math. Phys. **51** (1976) 183, ibid. **57** (1977) 193; Prog. Theor. Phys. **58** (1977) 1377.
[32] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Intergration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, 2002.
[33] Dietrich E. Wolf. *Modeling and Computer Simulation fo Granular Media*, pages 64–95. Springer, 1995.
[34] S. A. M. El Shourbagy, S. Okeda and H.-G. Matuttis. *Acoustic of sound propagation in Granular Materials in one, two and three dimensions*, to appear in J. Phys. Soc. Jpn. 77, 3, 2008