Nonsmooth Contact Dynamics for the Large–Scale Simulation of Granular Material

Jan Kleinert[a,b], Bernd Simeon[c], Klaus Dreßler[a]

[a]Fraunhofer ITWM, Fraunhofer Platz 1, 67663 Kaiserslautern, Germany
[b]Fraunhofer SCAI, Schloss Birlinghofen, 53754 Sankt Augustin, Germany
[c]FB Mathematik, TU Kaiserslautern, 67663 Kaiserslautern, Germany

Abstract

For the large-scale simulation of granular material, the Nonsmooth Contact Dynamics Method (NSCD) is examined. First, the equations of motion of nonsmooth mechanical systems are introduced and classified as a differential variational inequality that has a structure similar to Differential-Algebraic Equations (DAEs). Using a Galerkin projection in time, we derive nonsmooth extensions of the *SHAKE* and *RATTLE* schemes. A matrix–free Interior Point Method (IPM) is used for the complementarity problems that need to be solved in each time step. We demonstrate that in this way, the NSCD approach yields highly accurate results and is competitive compared to the Discrete Element Method (DEM).

Keywords: Granular Material, Nonsmooth Contact Dynamics, Friction, Interior Point Methods, Conical Optimization, Timestepping Scheme

1. Introduction

Granular material such as powders, pellets, sand and gravel are present in numerous engineering applications. The main loads acting on an excavator or wheel loader, for instance, result from the interaction of the shovel with the ground. A simulation framework to predict these loads is a desirable tool but still represents a great challenge today. Currently, two methods are mainly used: the classical Discrete Element Method (DEM) [1] and Non-Smooth Contact Dynamics (NSCD) [2, 3]. The DEM has shown its potential to deliver loads – the so-called draft forces – that are in agreement with experimental data [4]. But a simulation of a few seconds may take up days to weeks of computing time. On the other hand, NSCD provides a class of

Preprint submitted to Journal of Comp. and Applied Math. December 22, 2015

methods that are substantially faster and more stable [5], but due to their low accuracy, they fail to reproduce realistic draft forces.

In NSCD a large complementarity problem must be solved in every time step. The survey [6] demonstrates that NSCD, used in combination with the Projected Gauß Jacobi (PGJ) method to solve the complementarity problems, delivers visually pleasing results in very short computing time. Yet it requires many PGJ iterations to achieve an accurate force result from a digging scenario. In fact, it turns out that PGJ is by orders of magnitude slower than DEM in this case.

This shortcoming of NSCD implies two tasks. The first task is to analyze the mathematical model behind NSCD, and the second one is to replace the PGJ method by a more accurate solver for the complementarity problems. We present a concise and mathematically sound problem formulation that is a generalization of the Differential Algebraic Equations (DAEs) in multibody dynamics [7]. The constraint forces turn here into Lagrangian multipliers in the form of differential measures. In addition, we discretize the equations in time as extensions of the well–known *SHAKE* and *RATTLE* integrators [8].

The main obstacle is then to solve the large-scale complementarity problems accurately and reasonably fast in each time step, which replaces Newton's method in smooth multibody dynamics. Noteworthy contributions in this direction are [9], where the authors employ a semi–smooth Newton method; [10] where the authors consider quadratic programming and Interior Point Methods (IPM) for linear complementarity problems; [5] where Krylov subspace methods are generalized for the solution of variational inequalities; and finally, the PhD thesis [11], which compares active–set strategies, accelerated gradient–descent methods and IPMs. We propose the use of a conical IPM. It exploits the Jordan–algebraic structure of $\mathbb{R}^3$ and the symmetric cones associated with Jordan algebras to tackle a large number of frictional contacts. By combination with a conjugate gradient method and tailored preconditioning, the IPM becomes matrix-free and remarkably efficient.

This contribution continues the work [12] and the thesis [13]. For brevitiy, basic concepts such as Jordan algebras and Lebesgue–Stieltjes measures are not included but can be found in [13]. In Sect. 2, the equations of motion for nonsmooth mechanical systems subject to inequality constraints are discussed while Sect. 3 shows that a Petrov–Galerkin discretization in time yields nonsmooth pendants of the well–known *SHAKE* and *RATTLE* schemes. Sect. 4 contains a brief description of the IPM that is in each time step, and numerical results are presented in Sect. 5.

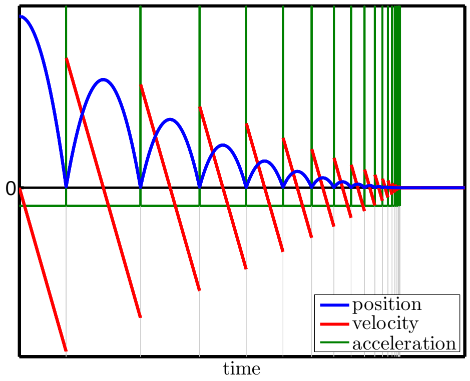


Figure 1: Position, velocity and acceleration of a bouncing ball

2. Nonsmooth Mechanical Systems

The equations of motion for a constrained mechanical system are usually written as a DAE [7]

$$\begin{aligned} M(\mathbf{q})\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{q},\dot{\mathbf{q}},t) - G(\mathbf{q},t)^T\boldsymbol{\lambda}, \\ \mathbf{0} &= \mathbf{g}(\mathbf{q},t). \end{aligned} \tag{1}$$

Here, $\mathbf{q}(t)$ denotes the position of the system, $M(\mathbf{q})$ is the mass matrix, $\mathbf{f}(\mathbf{q},\dot{\mathbf{q}},t)$ are force terms, $\mathbf{g}(\mathbf{q},t)$ are the constraints acting on the system, $G(\mathbf{q},t)$ is the constraint Jacobian and $\boldsymbol{\lambda}(t)$ are the Lagrangian multipliers associated to the constraints. If the Delassus matrix

$$\tilde{N} = G(\mathbf{q},t)M(\mathbf{q})G(\mathbf{q},t)^T$$

is invertible, it holds

$$\mathbf{q} \in \mathcal{C}^2\left([t_0,t_1]\right) \quad \text{and} \quad \boldsymbol{\lambda} \in \mathcal{C}^0\left([t_0,t_1]\right).$$

Moreover, (1) is then a DAE of index 3.

The presence of contact dynamics, however, involves a loss of regularity for all unknowns in (1). The example of a bouncing ball illustrates this, see Figure 1. A perfectly rigid ball accelerates under gravity towards the ground. At the time of impact, the ball's velocity must change instantaneously from a negative value v^- to a positive value $-e \cdot v^-$, where e is the coefficient of restitution. Thus, the position of the ball cannot be in $\mathcal{C}^2\left([t_0,t_1]\right)$. In fact it is not even in $\mathcal{C}^1$, hence the term nonsmooth. As a consequence, such nonsmooth systems cannot be formulated as a DAE anymore. Instead, a weak form based on Differential Variational Inclusions (DVI) is required where only the net motion within any time interval is described.

Consider a mechanical system consisting of m rigid bodies with six degrees of freedom each. Let the position of every rigid body be described by an absolutely continuous function in time, i.e. $\mathbf{q} \in \mathrm{AC}^{6m}([t_0, t_1])$. Let $\mathbf{v} = \dot{\mathbf{q}}$ denote the velocity of the system and $M(\mathbf{q}) \in \mathrm{AC}^{6m\times 6m}([t_0, t_1])$ the mass matrix. The Lagrangian function

$$L(\mathbf{q}, \mathbf{v}) = T(\mathbf{q}, \mathbf{v}) - V(\mathbf{q}) = \frac{1}{2}\mathbf{v}^T M(\mathbf{q})\mathbf{v} - V(\mathbf{q}), \tag{2}$$

is the difference of kinetic energy $T(\mathbf{q}, \mathbf{v})$ and potential energy $V(\mathbf{q})$.

Leaving inequality constraints aside, a nonsmooth version of the Euler-Lagrange equations can be derived from Hamilton's principle of stationary action, see [13]. It reads

$$\int_{t_0}^{t_1} \boldsymbol{\varphi}^T \left(\frac{\partial L}{\partial q}\right) \mathrm{d}t = \int_{t_0}^{t_1} \boldsymbol{\varphi}^T \,\mathrm{d}\left(\frac{\partial L}{\partial \mathbf{v}}\right) \qquad \forall \boldsymbol{\varphi} \in \mathrm{AC}^{6m}([t_0, t_1]). \tag{3}$$

Here, $\mathrm{d}\left(\partial L/\partial \mathbf{v}\right)$ is the Lebesque–Stieltjes measure of the function of bounded variation $\mathbf{p} = \partial L/\partial \mathbf{v}$. The measure can be identified with the distributional derivative of $\mathbf{p}$. It is not possible to derive the classical Euler Lagrange Equations if the momentum $\mathbf{p}$ is not a differentiable function in time everywhere. A short notation for Equation (3) is given by

$$\mathrm{d}\left(\frac{\partial L}{\partial \mathbf{v}}\right) = \frac{\partial L}{\partial q}\,\mathrm{d}t.$$

A convenient way to deal with inequality constraints is via conical inclusions. A subset $K \in \mathbb{R}^N$ is called a convex cone if for all $\alpha, \beta \geq 0$ and $\mathbf{x}, \mathbf{y} \in K$ it holds

$$\alpha\mathbf{x} + \beta\mathbf{y} \in K.$$

To motivate the use of conical inclusions, consider the following examples. *(i)* In every vector space the set $K = \{\mathbf{0}\}$ is a convex cone and it holds

$$\mathbf{g}(\mathbf{q}, t) \in K \quad \Leftrightarrow \quad \mathbf{g}(\mathbf{q}, t) = \mathbf{0}.$$

Therefore, equality constraints can be written as a conical inclusion.
(ii) For $K = \mathbb{R}_+ \subset \mathbb{R}$ it holds

$$g(\mathbf{q}, t) \in K \quad \Leftrightarrow \quad \mathbf{g}(\mathbf{q}, t) \geq 0.$$

Inequality constraints can hence be written as conical inclusions.
(iii) The Coulomb Friction cone

$$K_\mu = \left\{ \begin{bmatrix} \lambda_\mathbf{n} \\ \boldsymbol{\lambda}_\mathbf{t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2 \;\middle|\; \mu\lambda_\mathbf{n} \geq \|\boldsymbol{\lambda}_\mathbf{t}\| \right\}$$

with frictional coefficient μ is a convex cone. In general, friction cannot be interpreted as a constraint, but it is possible to approximate Coulomb friction using a conical constraint within short intervals $[t, t+\epsilon]$, see [13, Section 4.2.2]. This model is based on the constraint

$$\boldsymbol{g}(\mathbf{q}, t) = \begin{bmatrix} \phi_\mathbf{n} + \mu \int_t^{t+\epsilon} \|\dot{\boldsymbol{\phi}}_\mathbf{t}\| \, \mathrm{d}t \\ \boldsymbol{\phi}_\mathbf{t} \end{bmatrix} \in K_\mu^*,$$

where

$$K_\mu^* = \left\{ \begin{bmatrix} \nu_\mathbf{n} \\ \boldsymbol{\nu}_\mathbf{t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2 \;\middle|\; \nu_\mathbf{n} \geq \mu\|\boldsymbol{\nu}_\mathbf{t}\| \right\}, \quad \boldsymbol{\phi} = \begin{bmatrix} \phi_\mathbf{n} \\ \boldsymbol{\phi}_\mathbf{t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2$$

are the dual cone of K_μ and the normal and tangential contact displacement of two bodies, respectively. This approximation extends the Coulomb friction model of DeSaxcé and Feng [14]. In [13] it is shown that the introduced error is small in low order time integration methods. The great benefit is that friction can be treated together with other constraints in a unified framework.

Theorem 1. *Let $K \subset \mathbb{R}^N$ be a convex cone. Assume the following assumptions hold.*

(a) $\mathbf{q} \in \mathrm{AC}^{6m}([t_0, t_1])$.
(b) $\mathbf{g} : \mathbb{R}^{6m} \times [t_0, t_1] \to \mathbb{R}^N$ *is a continuously differentiable function.*
(c) $\mathbf{g}(\mathbf{q}(t), t) \in K$ *for all* $t \in [t_0, t_1]$.
(d) *It holds for every* $t \in [t_0, t_1]$

$$\operatorname{im} G(\mathbf{q}(t), t) - K_{\mathbf{g}(\mathbf{q}(t),t)} = \mathbb{R}^N$$

where

$$G(\mathbf{q}(t), t) := \frac{\partial \mathbf{g}(\mathbf{q}(t), t)}{\partial \mathbf{q}} \in \mathbb{R}^{N \times 6m}$$

is the constraint Jacobian and

$$K_{\mathbf{g}(\mathbf{q}(t),t)} = K + span\,(\mathbf{g}(\mathbf{q}(t),t)).$$

Then there exists a Lebesgue–Stieltjes measure $\mathrm{d}\boldsymbol{\lambda}$ *with* $\mathrm{d}\boldsymbol{\lambda}(I) \in K^*$ *for all subintervals* $I \subset [t_0, t_1]$ *such that*

$$\int_{t_0}^{t_1} \boldsymbol{\varphi}^T \,\mathrm{d}\left(\frac{\partial L}{\partial \mathbf{v}}\right) = \int_{t_0}^{t_1} \boldsymbol{\varphi}^T \left(\frac{\partial L}{\partial \mathbf{q}}\right) \mathrm{d}t + \int_{t_0}^{t_1} \boldsymbol{\varphi}^T G(\mathbf{q},t)^T \,\mathrm{d}\boldsymbol{\lambda} \tag{4}$$

for all $\boldsymbol{\varphi} \in \mathrm{AC}^{6m}([t_0,t_1])$ *and*

$$0 = \int_{t_0}^{t_1} \mathbf{g}(\mathbf{q}(t),t)^T \,\mathrm{d}\boldsymbol{\lambda}. \tag{5}$$

For the proof we refer to [13, Theorems 3.1 and 3.2]. The measure $\mathrm{d}\boldsymbol{\lambda}$ is the equivalent of a Lagrangian multiplier. It can be identified with the reaction force associated to the constraint. Moreover, (4) is a Measure Differential Equation (MDE) and (5) a Cone Complementarity Problem (CCP), yielding in total a Differential Variational Inequality (DVI) [15]. A compact notation for (4) and (5) reads

$$\begin{aligned} &\mathrm{d}\left(\frac{\partial L}{\partial \mathbf{v}}\right) = \frac{\partial L}{\partial \mathbf{q}}\mathrm{d}t + G(\mathbf{q},t)^T \,\mathrm{d}\boldsymbol{\lambda} \\ &K \ni \mathbf{g}(\mathbf{q},t) \quad \perp \quad \mathrm{d}\boldsymbol{\lambda} \in K^*. \end{aligned}$$

Finally, by inserting (2) and using $\mathbf{f}(\mathbf{q},\mathbf{v},t) := -\nabla V(\mathbf{q}) - \frac{1}{2}\mathbf{v}\frac{\partial M(\mathbf{q})}{\partial \mathbf{q}}\mathbf{v}$, one obtains the equations of motion

$$M(\mathbf{q})\,\mathrm{d}\mathbf{v} = \mathbf{f}(\mathbf{q},\mathbf{v},t)\,\mathrm{d}t + G(\mathbf{q},t)^T \,\mathrm{d}\boldsymbol{\lambda} \tag{6a}$$

$$K \ni \mathbf{g}(\mathbf{q},t) \quad \perp \quad \mathrm{d}\boldsymbol{\lambda} \in K^*. \tag{6b}$$

Since $\mathbf{g}$ is a continuously differentiable function and $\mathbf{q}$ is absolutely continuous, the function $\mathbf{g}(\mathbf{q}(\cdot),\cdot)$ is an absolutely continuous function. According to Theorem 1 the Lagrangian multipliers are from the dual space of absolutely continuous functions. The dual space of absolutely continuous functions consists of signed Radon measures, i.e., distributions [16, 17]. Therefore impulsive forces are automatically included in this formulation simply due to the assumption of absolute continuity of positions.

The complementarity conditions frequently encountered when dealing with nonsmooth mechanical systems are a direct corollary from Theorem 1. Consider the case, where all constraints are inequality constraints, i.e. the cone K is the positive orthant $K = \mathbb{R}^N_+$. It holds $K^* = K$, and therefore the reaction forces must be positive. The complementarity in (5) means that if one component of $\mathbf{g}(\mathbf{q}, t)$ is positive in a time interval, the corresponding component in $\mathrm{d}\boldsymbol{\lambda}$ must be zero and vice versa.

Finally, for equality constraints, i.e. $K = \{\mathbf{0}\}$, it holds $K^* = \mathbb{R}^N$ and the complementarity condition (5) reduces to $\mathbf{g}(\mathbf{q}, t) = \mathbf{0}$. If $\mathbf{q}$ is continuously differentiable and $\mathbf{v}$ is absolutely continuous, it holds $\mathrm{d}\mathbf{v} = \dot{\mathbf{v}}\,\mathrm{d}t = \ddot{\mathbf{q}}\,\mathrm{d}t$ and the DAE (1) can be derived from the DVI (6) by using the fundamental lemma of the calculus of variations. In this sense, (6) is a generalization of (1) that is able to deal with nonsmooth impacts and inequality constraints.

3. Time Discretizaton

Clearly, Taylor expansion of $\mathbf{q}(t + \Delta t)$ and $\mathbf{v}(t + \Delta t)$ is inappropriate for deriving numerical methods in the nonsmooth case. The DVI (6) is a variational formulation in a function space setting and resembles the weak form of a partial differential equation where the spacial descretization is achieved by the Galerkin projection. A similar approach is used here in time rather than space, based on [18]. As starting point, consider a one dimensional problem where $q : [t_0, t_1] \to \mathbb{R}$ is absolutely continuous and $v = \dot{q} : [t_0, t_1] \to \mathbb{R}$ of bounded variation. The MDE (6a) then reads

$$\int_{t_0}^{t_1} \varphi M(q)\,\mathrm{d}v = \int_{t_0}^{t_1} \varphi f\,\mathrm{d}t + \int_{t_0}^{t_1} \phi G(q,t)^T\,\mathrm{d}\lambda \;\text{ for all } \varphi \in \mathcal{D}_0, \tag{7}$$

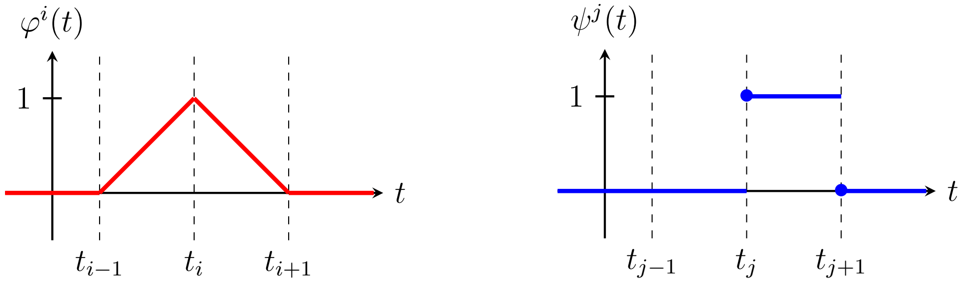
where

$$\mathcal{D}_0 = \{\; q \in \mathrm{AC}[t_0, t_1] \;\mid\; \dot{q} \in \mathrm{SBV}(t_0, t_1) \text{ and } q(t_0) = q(t_1) = 0 \;\}$$

and $\lambda, v \in \mathrm{SBV}(t_0, t_1)$. The function space $\mathrm{SBV}(t_0, t_1)$ denotes the space of functions of bounded variation without a Cantor part, see [13] for details. The idea is to satisfy (7) in finite dimensional subspaces of $\mathcal{D}_0$ and $\mathrm{SBV}(t_0, t_1)$.

Let $\{[t_i, t_{i+1}], 0 \leq i \leq s\}$ be a partition of $[t_0, t_1]$ and assume, for simplicity, that the step size $h_j = t_{j+1} - t_j$ is constant, i.e. $h := h_k = h_j$ for all $j, k \in \{0, ..., s\}$. An intuitive choice for a finite dimensional subspace of $\mathcal{D}_0 \subset \mathrm{AC}[t_0, t_1]$ is the set of continuous, piecewise linear functions

$$A = \{\;\; \varphi \in \mathcal{D}_0 \text{ continuous, piecewise linear in } [t_j, t_{j+1}] \;\;\} \subset \mathcal{D}_0.$$



(a) The basis function $\varphi^i(t)$ of the space $A \subset \mathcal{D}_0$

(b) The basis function $\psi^j(t)$ of the space $B \subset \mathrm{SBV}[t_0, t_1)$

Figure 2: Basis functions for finite dimensional subspaces of $\mathcal{D}_0$ and $\mathrm{SBV}[t_0, t_1)$

A basis for A is given by $\mathcal{B}_A = \{ \quad \varphi^i \in A, \quad i = 1, ..., s \quad \}$, where the φ^is are hat–functions,

$$\varphi^i(t) = \begin{cases} (t - t_i)/h & \text{if } t \in [t_{i-1}, t_i], \\ (t_{i+1} - t)/h & \text{if } t \in [t_i, t_{i+1}], \\ 0 & \text{else,} \end{cases}$$

see Fig. 2(a). The specific choice for a basis is arbitrary, but it makes sense to choose functions that are easy to integrate and have small support. A simple finite dimensional subspace of $\mathrm{SBV}[t_0, t_1)$ is the space of piecewise constant functions. These functions are allowed to have jumps at the interval boundaries. Consider the space

$$B = \{ \quad \psi \in \mathrm{SBV}[t_0, t_1) \text{ piecewise constant in } [t_j, t_{j+1}) \quad \} \subset \mathrm{SBV}[t_0, t_1)$$

with basis

$$\mathcal{B}_B = \{ \quad \psi^j \in B, \quad j = 0, ..., s \quad \}.$$

The basis functions are given by

$$\psi^j(t) = \begin{cases} 1 & \text{if } t \in [t_j, t_{j+1}), \\ 0 & \text{else,} \end{cases} \tag{8}$$

see Figure 2(b).

We can write $v, \lambda \in B$ as a linear combination of functions from $\mathcal{B}_B$,

$$v = \sum_{j=0}^{s} \bar{v}^j \psi^j \quad \text{and} \quad \lambda = \sum_{j=0}^{s} \bar{\lambda}^j \psi^j$$

with coefficients $\bar{v}^j, \bar{\lambda}^j \in \mathbb{R}$. By construction, $\bar{v}^i$ corresponds to constant velocity in the interval $[t_i, t_{i+1})$. The same holds for $\bar{\lambda}_i$ and λ. To indicate this, we write

$$v = \sum_{j=0}^{s} v^{j+1/2}\psi^j \quad \text{and} \quad \lambda = \sum_{j=0}^{s} \lambda^{j+1/2}\psi^j, \tag{9}$$

where $v^{j+1/2}$ and $\lambda^{j+1/2}$ denote the constant values of v and λ in the interval $[t_j, t_{j+1})$, respectively.

Next, introduce the abbreviation $q^i := q(t_i)$ and substitute

$$k^i := \int_0^T \varphi^i f \,\mathrm{d}t \tag{10}$$

for the force term. Usually, the external forces are known in advance, and the integral can be evaluated using appropriate quadrature methods.

To satisfy the MDE (7) in finite dimensions, it suffices to test it against all $\varphi^i \in \mathcal{B}_A$. Insertion of (9) and (10) into (7) yields

$$\begin{aligned} &\int_0^T \varphi^i M(q)\,\mathrm{d}\left(\sum_{j=0}^{s} v^{j+1/2}\psi^j\right) = k^i + \int_0^T \varphi^i G(q,t)^T\,\mathrm{d}\left(\sum_{k=0}^{s} \lambda^{k+1/2}\psi^k\right) \\ \Leftrightarrow\ &\sum_{j=0}^{s} v^{j+1/2}\int_0^T \varphi^i M(q)\,\mathrm{d}\psi^j = k^i + \sum_{k=0}^{s} \lambda^{k+1/2}\int_0^T \varphi^i G(q,t)^T\,\mathrm{d}\psi^k \end{aligned} \tag{11}$$

for all $\varphi^i \in \mathcal{B}_A$. Here, the Lebesgue–Stieltjes measure of ψ^j is given by the difference of two point measures,

$$\mathrm{d}\psi^j = \delta_{t_j} - \delta_{t_{j+1}}, \qquad \delta_t(I) := \begin{cases} 1, & \text{if } t \in I \\ 0, & \text{else.} \end{cases}.$$

Finally, by evaluating the integrals in (11) and inserting λ from (9) into the CCP $0 = \int_0^T g(q(t),t)\,\mathrm{d}\lambda$, one obtains the DVI in finite dimension

$$M(q^j)(v^{j+1/2} - v^{j-1/2}) = k^j + G(q^j, t_j)^T\gamma^j, \tag{12a}$$

$$K \ni g(q^j, t_j) \quad \perp \quad \gamma^j \in K^* \tag{12b}$$

for all $j = 1, ..., s$. Here,

$$\gamma^j = \lambda^{j+1/2} - \lambda^{j-1/2} \in K^*$$

appears as a new unknown. From a physical point of view, γ^j can is the reaction impulse of the constraint at time t_j. The equations of motion (12) hold in the same way for a rigid body system with m rigid bodies in three dimensions, i.e., for trajectories $\mathbf{q} : [0, T] \to \mathbb{R}^{6m}$:

$$\mathbf{q}^{j+1} = \mathbf{q}^j + h\mathbf{v}^{j+1/2}, \tag{13a}$$

$$M(\mathbf{q}^j)(\mathbf{v}^{j+1/2} - \mathbf{v}^{j-1/2}) = \mathbf{k}^j + G(\mathbf{q}^j, t_j)^T \boldsymbol{\gamma}^j, \tag{13b}$$

$$K \ni \mathbf{g}(\mathbf{q}^j, t_j) \quad \perp \quad \boldsymbol{\gamma}^j \in K^*. \tag{13c}$$

The complementarity in the CCP (13c) means that the scalar product of $\mathbf{g}(\mathbf{q}^j, t_j)$ and $\boldsymbol{\gamma}^j$ must be zero.

Unfortunately, (13) yields an explicit integration method that is not of much use. Whenever $\mathbf{g}(\mathbf{q}^j, t_j)$ is in the interior of K, the CCP (13c) is trivially satisfied by $\boldsymbol{\gamma}^j = \mathbf{0}$. We can immediately evaluate (13b) to calculate $\mathbf{v}^{j+1/2}$. But then there is no guarantee that the constraints are satisfied at the end of the time step, i.e., that

$$\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}) = \mathbf{g}(\mathbf{q}^j + h\mathbf{v}^{j+1/2}, t_{j+1}) \in K.$$

3.1. The SHAKE Integrator

A simple way of transforming (13) into a meaningful method is by introducing some implicitness into the CCP (13c). We replace (13) by

$$\mathbf{q}^{j+1} = \mathbf{q}^j + h\mathbf{v}^{j+1/2}, \tag{14a}$$

$$M(\mathbf{q}^j)(\mathbf{v}^{j+1/2} - \mathbf{v}^{j-1/2}) = \mathbf{k}^j + G(\mathbf{q}^j, t_j)^T \boldsymbol{\gamma}^j, \tag{14b}$$

$$K \ni \mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}) \quad \perp \quad \boldsymbol{\gamma}^j \in K^*. \tag{14c}$$

One can combine (14a) and (14b) to obtain a scheme without velocity terms in the recursion,

$$\begin{aligned} \mathbf{q}^{j+1} - 2\mathbf{q}^j + \mathbf{q}^{j-1} &= (\mathbf{q}^{j+1} - \mathbf{q}^j) - (\mathbf{q}^j - \mathbf{q}^{j-1}) \\ &= h(\mathbf{v}^{j+1/2} - \mathbf{v}^{j-1/2}) \\ &= hM(\mathbf{q}^j)^{-1} \left(\mathbf{k}^j + G(\mathbf{q}^j, t_j)^T \boldsymbol{\gamma}^j \right). \end{aligned}$$

This demonstrates that (14) is exactly the *SHAKE* integrator for nonsmooth mechanical systems with impact, cf. [8, Equation 6.10].

The main task of *SHAKE* in each step is to solve the CCP (14c). The reaction impulse $\boldsymbol{\gamma}^j$ solves the nonlinear, not necessarily convex, constrained optimization problem

$$\min\ \boldsymbol{\gamma}^{jT}\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}) \tag{15}$$

such that

$$\begin{aligned} &\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}) \in K, \quad \boldsymbol{\gamma}^j \in K^*, \\ &\mathbf{q}^{j+1} = \mathbf{q}^j + h\left[\mathbf{v}^{j-1/2} + M(\mathbf{q}^j)^{-1}\left(\mathbf{k}^j + G(\mathbf{q}^j, t_j)^T\boldsymbol{\gamma}^j\right)\right]. \end{aligned} \tag{16}$$

Depending on $\mathbf{g}$, this problem may be very challenging and even not be solvable at all. This is the case especially for frictional constraints. Anitescu proposes in [19] a convexification scheme to simplify the solution. In our work the constraint $\mathbf{g}$ is replaced by a linearized (and scaled) version

$$\begin{aligned} \mathbf{u}^{j+1} :&= \frac{\mathbf{g}(\mathbf{q}^j, t_j)}{h} + G(\mathbf{q}^j, t_j)\mathbf{v}^{j+1/2} + \frac{\partial \mathbf{g}(\mathbf{q}^j, t_j)}{\partial t} \\ &\approx \frac{1}{h}\mathbf{g}(\mathbf{q}^j + h\mathbf{v}^{j+1/2}, t_j + h) = \frac{1}{h}\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}). \end{aligned} \tag{17}$$

Inserting (14b) into (17) yields

$$\begin{aligned} \mathbf{u}^{j+1} &= G(\mathbf{q}^j, t_j)M(\mathbf{q}^j)^{-1}G(\mathbf{q}^j, t_j)^T\boldsymbol{\gamma}^j \\ &\quad + \frac{\mathbf{g}(\mathbf{q}^j, t_j)}{h} + \frac{\partial \mathbf{g}(\mathbf{q}^j, t_j)}{\partial t} + G(\mathbf{q}^j, t_j)\left(\mathbf{v}^{j-1/2} + M(\mathbf{q}^j)^{-1}\mathbf{k}^j\right) \\ &= \bar{N}^j\boldsymbol{\gamma}^j + \bar{\mathbf{r}}^j \end{aligned}$$

where

$$\bar{N}^j := G(\mathbf{q}^j, t_j)M(\mathbf{q}^j)^{-1}G(\mathbf{q}^j, t_j)^T \quad \text{and} \tag{18}$$

$$\bar{\mathbf{r}}^j := \frac{\mathbf{g}(\mathbf{q}^j, t_j)}{h} + \frac{\partial \mathbf{g}(\mathbf{q}^j, t_j)}{\partial t} + G(\mathbf{q}^j, t_j)\left(\mathbf{v}^{j-1/2} + M(\mathbf{q}^j)^{-1}\mathbf{k}^j\right). \tag{19}$$

The CCP (14c) of *SHAKE* thus simplifies to

$$K \ni \mathbf{u}^{j+1} = \bar{N}^j\boldsymbol{\gamma}^j + \bar{\mathbf{r}}^j \ \perp\ \boldsymbol{\gamma}^j \in K^*. \tag{20}$$

Note that $\bar{N}^j$ and $\bar{\mathbf{r}}^j$ depend only on values from the previous step. The only remaining unknown is the reaction impulse $\boldsymbol{\gamma}^j$. We also observe that the Delassus matrix $\bar{N}^j$ is symmetric and positive semi–definite. For the simulation of granular material, it is, however, almost certainly rank–deficient. The numerical method used to solve the CCP should be able to cope with this. The conical constraints $\mathbf{u}^{j+1} \in K$ now appear as regularized, velocity–based constraints, and the resulting complementarity problem is, in fact, very similar to the convexified problem in [19].

3.2. An Extension of RATTLE

One downside of *SHAKE* in case of impacts is that the reaction impulses $\boldsymbol{\gamma}^j$ might not converge as the step size tends to zero. This is an issue when reaction forces are of interest since $\boldsymbol{\gamma}^j$ is the integral of all reaction forces over one time step. The problem of nonconvergence can easily be demonstrated with a one–dimensional example of a point mass q subject to no external forces moving towards the ground. The ground is modeled via the constraint $q \geq 0$. This constraint corresponds to a completely inelastic impact law. Assume that it holds $q(t_j) > 0$ and the point mass would impact the ground in continuous time at $t_c \in [t_j, t_{j+1}]$. Then (14) yields a reaction impulse γ^j and a velocity $v^{j+1/2}$ such that

$$q(t_{j+1}) = q(t_j) + hv^{j+1/2} = 0.$$

But because $q(t_j) > 0$ it holds

$$v(t_{j+1}^-) = v^{j+1/2} < 0,$$

i.e., the point mass has impacted with the ground but still has a negative velocity. Only after an additional step will it hold $q(t_{j+2}) = 0$ and $v(t_{j+2}^-) = 0$. Therefore, an impact is resolved by (14) always within two time steps, and the values of the reaction impulses $\boldsymbol{\gamma}^j$ and $\boldsymbol{\gamma}^{j+1}$ depend on when exactly in the interval $[t_j, t_{j+1}]$ the impact would have occurred in continuous time, see Fig. 3. Hence, the values of the reaction impulses directly depend on the time discretization.

A remedy is to append the system with an additional discrete velocity $\boldsymbol{\nu}^j$

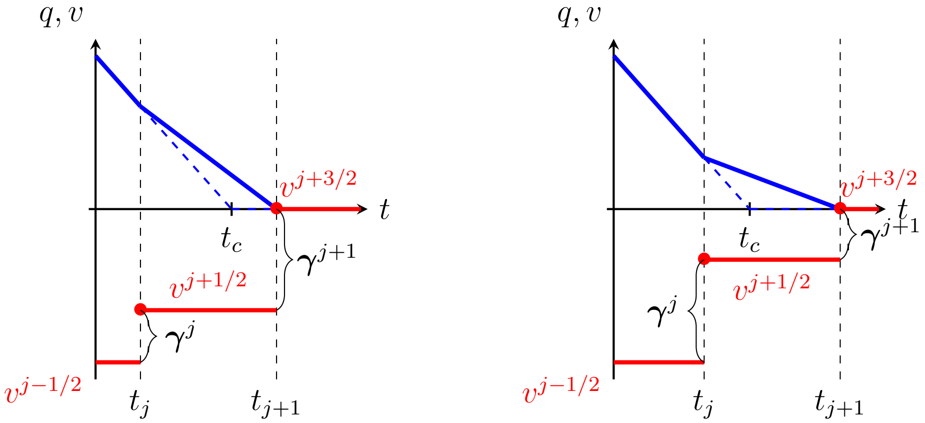


Figure 3: Time step for a point mass impacting ground. Dashed blue line: position in continuous time; solid blue line: result of *SHAKE*. The scenarios on the left and right are identical except that the time window $[t_j, t_{j+1}]$ around the continuous impact time t_c is shifted on the right. As a consequence, the values of $\boldsymbol{\gamma}^j$ and $\boldsymbol{\gamma}^{j+1}$ differ for both simulations.

that differs from $\dot{\mathbf{q}}$ and to solve two CCPs instead of one:

$$\mathbf{q}^{j+1} = \mathbf{q}^j + h\mathbf{v}^{j+1/2}, \tag{21a}$$

$$M(\mathbf{q}^j)(\mathbf{v}^{j+1/2} - \boldsymbol{\nu}^j) = \mathbf{k}^j + G(\mathbf{q}^j, t_j)^T \boldsymbol{\gamma}_l^j, \tag{21b}$$

$$K \ni \mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}) \quad \perp \quad \boldsymbol{\gamma}_l^j \in K^*, \tag{21c}$$

$$M(\mathbf{q}^j)(\boldsymbol{\nu}^{j+1} - \mathbf{v}^{j+1/2}) = G(\mathbf{q}^j, t_j)^T \boldsymbol{\gamma}_r^j, \tag{21d}$$

$$\begin{aligned} &G(\mathbf{q}^{j+1}, t_{j+1})\boldsymbol{\nu}^{j+1} + \frac{\partial \mathbf{g}(\mathbf{q}^{j+1}, t_{j+1})}{\partial t} \in T_K(\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1})) \\ &\quad \perp \quad \boldsymbol{\gamma}_r^j \in T_K^*(\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1})). \end{aligned} \tag{21e}$$

Here, (21a) – (21c) correspond to the *SHAKE* steps (14) while (21d) and (21e) satisfy the constraint on velocity level using the additional velocity $\boldsymbol{\nu}$. Furthermore, $T_K(\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1}))$ is the tangent cone to K evaluated at $\mathbf{g}(\mathbf{q}^{j+1}, t_{j+1})$.

The *total* reaction impulse

$$\boldsymbol{\gamma}^j := \boldsymbol{\gamma}_l^j + \boldsymbol{\gamma}_r^j$$

in the j–th time step now converges with $h \to 0$. The algorithm (21) results in the same solution as algorithm (14) for the positions $\mathbf{q}^j$ and differs only with respect to the reaction impulses. The scheme (21) is very similar to the *RATTLE* algorithm. Yet, there are two major differences that should be emphasized. Firstly, in contrast to the original version, the newly introduced velocity $\boldsymbol{\nu}^j$ has no direct physical interpretation. It is merely an auxiliary variable to ensure the convergence of the reaction impulses with $h \to 0$. Secondly, the force term $\mathbf{k}^j$ only contributes to the *real* velocity $\mathbf{v}^{j+1/2}$ of the dynamical system and plays no role in (21e). Obvioiusly, the computational effort for (21) is about twice the one for (14), and for this reason we concentrate now on the implementation of *SHAKE*.

4. Solving the CCP in Each Time Step

A time step of a simulation of granular material using *SHAKE* consists of three tasks. The first task is collision detection to obtain all pairs of rigid bodies that could potentially come into contact. The second task is the computation of reaction impulses by solving the CCP (20), and finally the integration of the reaction impulses to new velocities and positions via (14a) and (14b).

Of these tasks, the solution of the CCP (20) is the most time consuming. We propose the use of a matrix–free interior point method and provide here a short outline. For the details we refer to [12, 13].

4.1. Outline of the Interior Point Method

For simplicity, assume that all conical inclusions result from a contact model with Coulomb friction and a completely inelastic impact law. Then, the CCP (20) of *SHAKE* can be written as

$$\mathcal{K}_\mu \ni \boldsymbol{\gamma} \quad \perp \quad \mathbf{u} = \bar{N}\boldsymbol{\gamma} + \bar{\mathbf{r}} \in \mathcal{K}_\mu^*, \tag{22}$$

where $\mathcal{K}_\mu = K_{\mu_1} \times \ldots \times K_{\mu_n}$ is the direct product of the friction cones of the n contacts. Superscripts referring to the current time step have been dropped for better readability.

It turns out to be convenient to transform the CCP (22) to a CCP with a symmetric cone $\mathcal{C} = \mathcal{C}^*$. This is achieved using the linear transformations

$$\mathbf{x} = \mathcal{T}_x \cdot \boldsymbol{\gamma} = \begin{bmatrix} T^x_{\mu_1} & & \\ & \ddots & \\ & & T^x_{\mu_n} \end{bmatrix} \cdot \boldsymbol{\gamma} \quad \text{and} \quad \mathbf{y} = \mathcal{T}_y \cdot \mathbf{u} = \begin{bmatrix} T^y_{\mu_1} & & \\ & \ddots & \\ & & T^y_{\mu_n} \end{bmatrix} \cdot \mathbf{u}$$

with

$$T^x_{\mu_i} := \begin{bmatrix} \mu_i & & \\ & 1 & \\ & & 1 \end{bmatrix} \quad \text{and} \quad T^y_{\mu_i} := \begin{bmatrix} 1 & & \\ & \mu_i & \\ & & \mu_i \end{bmatrix}.$$

By writing $N = \mathcal{T}_y \bar{N} \mathcal{T}_x^{-1}$ and $\mathbf{r} = \mathcal{T}_y \bar{\mathbf{r}} \in \mathbb{R}^{3n}$, the CCP (22) is equivalent to

$$\mathcal{C} \ni \mathbf{x} \quad \perp \quad \mathbf{y} = N\mathbf{x} + \mathbf{r} \in \mathcal{C}, \tag{23}$$

where $\mathcal{C} = C \times .... \times C$ with

$$C = K_{\mu=1} = \left\{ \begin{bmatrix} x_{\mathbf{n}} \\ \mathbf{x_t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2 \quad \middle| \quad x_{\mathbf{n}} \geq \|\mathbf{x_t}\| \right\}.$$

It can be shown that

$$\mathbf{x}_i^T \mathbf{y}_i = 0, \;\; \mathbf{x}_i, \mathbf{y}_i \in C \quad \Leftrightarrow \quad \mathbf{x}_i \circ \mathbf{y}_i = \mathbf{0},$$

where

$$\mathbf{x} \circ \mathbf{y} = \begin{bmatrix} \mathbf{x}^T \mathbf{y} \\ x_{\mathbf{n}} \mathbf{y_t} + y_{\mathbf{n}} \mathbf{x_t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2$$

denotes the Jordan product of

$$\mathbf{x} = \begin{bmatrix} x_{\mathbf{n}} \\ \mathbf{x_t} \end{bmatrix}, \mathbf{y} = \begin{bmatrix} y_{\mathbf{n}} \\ \mathbf{y_t} \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^2.$$

The interior point method is iterative. In each iteration one solves

$$\mathbf{y} = N\mathbf{x} + \mathbf{r} \;\; \text{and} \;\; \mathbf{x}_i \circ \mathbf{y}_i = \begin{bmatrix} \alpha \\ \mathbf{0} \end{bmatrix}$$

for one $\alpha > 0$ and every $i = 1, ..., n$ using one Newton iteration. In a second step the so–called *centralizing parameter* α is decreased and the process is repeated. It holds

$$\mathbf{x}_i \circ \mathbf{y}_i \to \mathbf{0} \;\; \text{as} \;\; \alpha \to 0$$

and the method approaches the solution of the CCP via a sequence of regularized problems with beneficial numerical properties. The main task within a Newton step consists of solving a set of linear equations

$$A \Delta \mathbf{x} = \mathbf{b},$$

where $A = W + N$ is composed of the Delassus matrix N that remains constant during all iterations and a block–diagonal matrix W consisting of one 3×3 block per contact. One intrinsic property of the IPM is the fact that $\kappa(A) \in \mathcal{O}\left(\alpha^{-2}\right)$, i.e., the condition of the linear system deteriorates close to the exact solution.

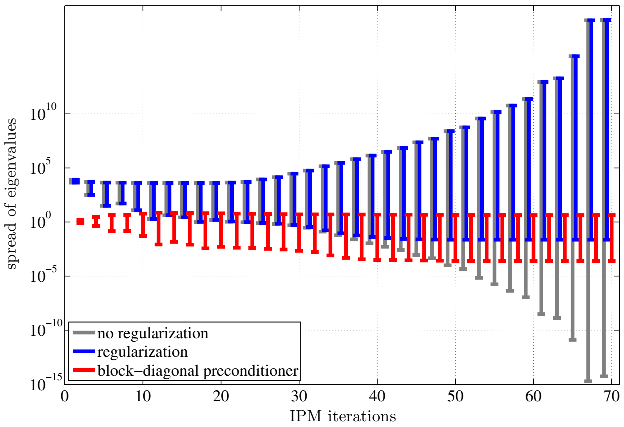


Figure 4: The distribution of eigenvalues in one time step of a simulation.

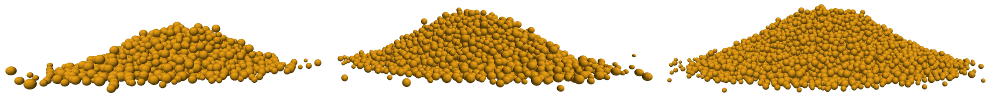
4.2. Preconditioning of the Linear Systems

The IPM can be implemented without the explicit construction of the usually large system Matrix A if the linear systems are solved using a Krylov subspace method, which only requires dot products and evaluations of the matrix vector product with A. The matrix vector product with A is computed by means of data associated to rigid bodies and all contacts without having to explicitly store the matrix in memory [20]. This is crucial as the matrix N changes its size and structure with each new contact.

Krylov subspace methods critically rely on good preconditioning, especially since the condition of A deteriorates close to the exact solution. Preconditioners such as Incomplete Lower Upper (ILU) or Incomplete Cholesky (IC) factorizations make no sense in combination with a matrix–free Krylov method. In our case, it turns out that a simple block–diagonal preconditioner suffices and delivers robust results, see [21] for a worst–case bound that is independent of the centralizing parameter α. An example that underlines this is given in Fig. 4. It shows the distribution of the eigenvalues of A, picked from one time step of a simulation of a static pile of 2,048 spherical non–rotational particles, with regularized linear system and block–diagonal preconditioner.

5. Numerical Results

The SHAKE scheme (14) and the IPM have been implemented in C++. As first benchmark we consider one time step of a simulation of a pile of $2,048$



(a) Test Problem 1 (b) Test Problem 2 (c) Test Problem 3

Figure 5: Three test problems with 2048, 5040 and 10192 particles

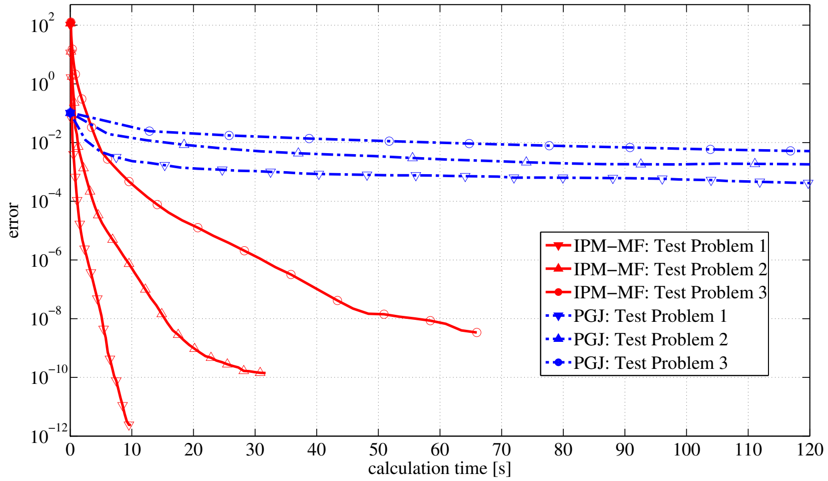
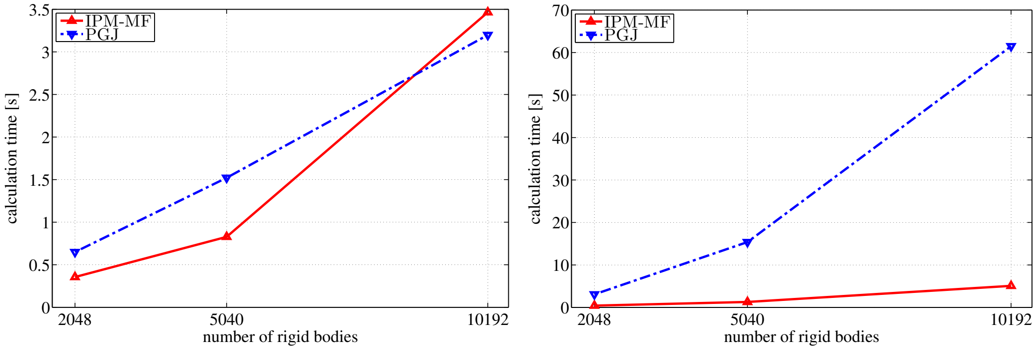


Figure 6: Error of PGJ and IPM depending on computing time

nonrotational spheres, Fig. 5(a). Two bodies are considered as potentially in contact if the minimum distance between them is less than half of the mean radius of the particles. This results in $n = 8,378$ potential contacts between pairs of spheres and $25,135$ unknowns in the CCP (22), which defines Test Problem 1. Test Problem 2 is a similar pile with $5,040$ nonrotational spheres, $n = 21,050$ potential contacts and $63,151$ unknowns in (22). Test Problem 3 consists of a pile with $10,192$ spheres, $n = 43,409$ contacts and $130,228$ unknowns in (22).

We compared different Krylov subspace methods and found that Conjugate Gradient (CG) was slightly faster than the Bi–Conjugate Gradient Method (BiCGstab) and much faster than the Generalized Minimum Residual Method (GMRES) in most cases. We also found the matrix–free implementation (denoted IPM-MF) using a block–diagonal preconditioner to be faster by several orders of magnitude in comparison to our implementation using a sparse matrix library and the ILU or IC preconditioners.



(a) $\texttt{tol} = 5 \cdot 10^{-2}$

(b) $\texttt{tol} = 1 \cdot 10^{-2}$

Figure 7: Computing time to achieve an error less than the tolerance `tol`

We compared the matrix–free IPM with the Projected Gauß Jacobi Method[1]. Figure 6 shows the convergence of PGJ and IPM for all three test cases. PGJ struggles specifically with larger problems. Its convergence rate stalled almost completely before a reasonably small error was achieved. Figure 7 shows how IPM-MF and PGJ scale with the problem size. Figure 7(a) shows the computing time that was needed to achieve an error of less than $5 \cdot 10^{-2}$. This is a very loose error bound where PGJ scales better with the problem size than IPM-MF. While IPM-MF beats PGJ in terms of computing time for the two smaller test cases, PGJ catches up for Test Problem 3, due to the additional overhead of IPM-MF in comparison to PGJ.

Figure 7(b) demonstrates that IPM-MF shines in terms of scalability with increasing accuracy requirements. If the error tolerance is set to 10^{-2}, PGJ is slower than IPM-MF in all three test cases. To achieve an error below $5 \cdot 10^{-4}$ on Test Problem 3, 73,600 PGJ iterations were necessary while IPM needed 2,757 Krylov iterations. PGJ requires here more than 200 times the computing time of IPM-MF. To conclude, IPM-MF scales better with the problem size than PGJ does whenever the accuracy requirements are high.

As a final test, the IPM is applied to an industrial size problem that demonstrates the application of our approach to large-scale problems with sufficient accuracy. A rectangular blade moves through a trench filled with granular material, Fig. 8. The material consists of $105,144$ spherical, non-rotational particles with radii randomly distributed between $0.008\,\mathrm{m}$ and

[1] See [20] for details on the Projected Gauß Jacobi Method for CCPs.

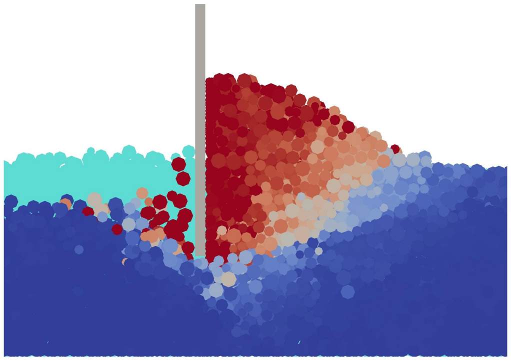


Figure 8: Blade moving through a trench of dry granular material.

0.016 m. The material has a bulk density of $1,382\,\mathrm{kg/m^3}$. All contacts have the same frictional coefficient of $\mu = 0.25$. The blade has a width of 0.2 m. It moves at a constant speed of 0.2 m/s and initial depth of 0.2 m.

A simulation is run with IPM-MF and step size $h = 10^{-2}$ s. In every step a CCP with an average of $1,261,972$ unknowns needs to be solved until the error falls below a predefined tolerance. Reaction forces acting on the blade are computed by dividing the reaction impulses by the step size. The draft force result was compared to the same reference DEM solution as in [6].

The simulation with a tolerance of 10^{-2} failed due to instability. At the beginning of the iteration reaction impulses are unrealistically large, and stopping the iteration prematurely leads to separation of particles. For simulations with tolerances below 10^{-4}, unrealistic separation of material could not be observed, see Figure 9.

Figure 10 shows the draft force result for the simulations with tolerances of 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6}. The draft forces are far off for low accuracy requirements but for tolerances of 10^{-5} and 10^{-6} the results agree with the

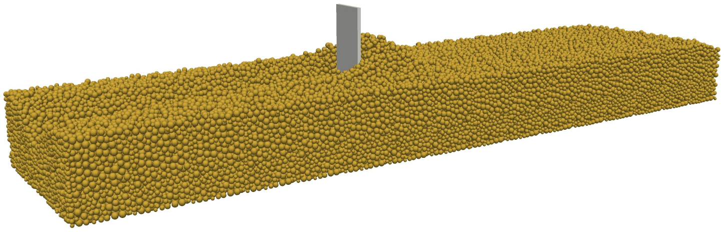


Figure 9: Illustration of simulation for moving blade

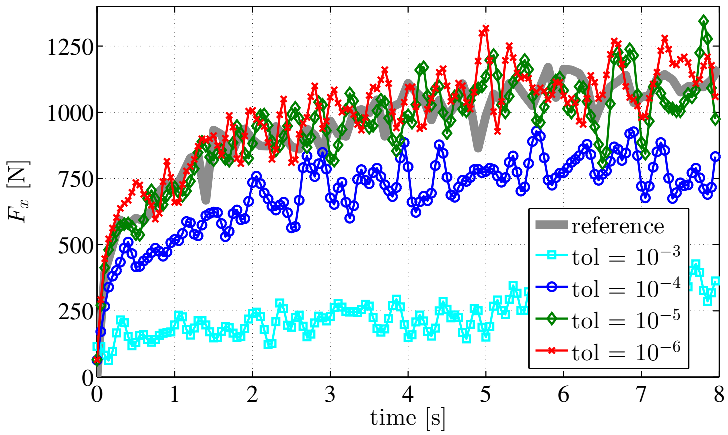


Figure 10: Horizontal draft forces acting on the blade for IPM simulations at different accuracy requirements and a time step size of $h = 10^{-2}$ s

tolerance	RTF	# iterations	$error(\boldsymbol{\gamma}, \mathbf{u})$
10^{-3}	20,884	3,428	$7.06 \cdot 10^{-4}$
10^{-4}	61,291	10,439	$8.69 \cdot 10^{-5}$
10^{-5}	83,945	15,357	$9.35 \cdot 10^{-6}$
10^{-6}	107,650	20,196	$4.20 \cdot 10^{-6}$

Table 1: Real time factor (RTF), average number of Krylov iterations per time step and average errors for simulation at a time step size of $h = 10^{-2}$ s

DEM reference solution. The draft forces converge to the reference solution with increasing accuracy of the CCP.

Table 1 shows the real time factor, the average number of Krylov iterations per time step and the average errors. The simulation with a tolerance of 10^{-5} achieved a result that was in good agreement with the DEM simulation at a real time factor of 83,945. The reference DEM simulation took 73,678 seconds of computing time per second of simulation, so it was in total 12.2% faster than IPM. The best draft force result achieved with PGJ was at a time step size of 10^{-3} s and 2,000 PGJ iterations at a real time factor of 892,780. Thus, IPM was marginally slower than DEM but more than 10 times faster than PGJ.

6. Conclusion and Outlook

A concise mathematical model for nonsmooth mechanical systems has been presented that extends the notion of classical DAEs in multibody dynamics. Discretization in time by a Galerkin projection leads to nonsmooth versions of *SHAKE* and *RATTLE*, where the latter has the property that the Lagrange multipliers converge as the step size tends to zero.

In every time step of a simulation, a cone complementarity problem must be solved, and here our choice of the interior point method turns out to be competitive and superior to projected Gauß-Jacobi for higher accuracy. Using matrix–free Krylov solvers and a simple but effective preconditionier, the IPM achieves the same result in terms of draft forces as the classical DEM in a comparable amount of time.

Many questions, however, remain open in this emerging field. In particular, the numerical analysis lacks appropriate norms and tools to measure and prove convergence. Concerning the computational aspects, the IPM can further be parallelized using domain decomposition methods. If a parallel version of the IPM is used as a solver and there is a reliable error measure that guarantees good force estimates below a certain threshold, NSCD has the potential to outperform classical DEM for all kinds of applications while maintaining its good stability properties.

References

[1] P. A. Cundall, O. D. L. Strack, A discrete numerical model for granular assemblies, Géotechnique 29 (1) (1979) 47 – 65.

[2] J. J. Moreau, Evolution problem associated with a moving convex set in a Hilbert space, Journal of Differential Equations 26 (3) (1977) 347 – 374.

[3] M. Jean, The non-smooth contact dynamics method, Computer Methods in Applied Mechanics and Engineering 177 (3 – 4) (1999) 235 – 257.

[4] M. Obermayr, K. Dressler, C. Vrettos, P. Eberhard, Prediction of draft forces in cohesionless soil with the discrete element method, Journal of Terramechanics 48 (5) (2011) pp. 347–358.

[5] T. Heyn, M. Anitescu, A. Tasora, D. Negrut, Using Krylov subspace and spectral methods for solving complementarity problems in many-body contact dynamics simulation, International Journal for Numerical Methods in Engineering 95 (7) (2013) 541 – 561.

[6] J. Kleinert, M. Obermayr, M. Balzer, Modeling of large scale granular systems using the discrete element method and the non-smooth contact dynamics method: A comparison, in: Proceedings of the ECCOMAS Multibody Dynamics Conference, Zagreb, 2013.

[7] B. Simeon, Computational flexible multibody dynamics: a differential-algebraic approach, Springer Science & Business Media, 2013.

[8] E. Hairer, C. Lubich, G. Wanner, Geometric numerical integration illustrated by the Störmer–Verlet method, Acta Numerica 12 (2003) 399 – 450.

[9] G. Daviet, F. Bertails-Descoubes, L. Boissieux, A hybrid iterative solver for robustly capturing Coulomb friction in hair dynamics, ACM Transactions on Graphics (TOG) - Proceedings of ACM 30 (6).

[10] K. Krabbenhoft, A. V. Lyamin, J. Huang, M. V. da Silva, Granular contact dynamics using mathematical programming methods, Computers and Geotechnics 43 (0) (2012) 165 – 176.

[11] T. Heyn, On the modeling, simulation, and visualization of many–body dynamics problems with friction and contact, Ph.D. thesis, Uiversity of Wisconsin–Madison (2013).

[12] J. Kleinert, B. Simeon, M. Obermayr, An inexact interior point method for the large-scale simulation of granular material, Computer Methods in Applied Mechanics and Engineering 278 (0) (2014) 567 – 598.

[13] J. Kleinert, Simulating granular material using nonsmooth time–stepping and a matrix–free interior point method, Ph.D. thesis, Technische Universität Kaiserslautern (2015).

[14] G. DeSaxcé, Z.-Q. Feng, The bipotential method: A constructive approach to design the complete contact law with friction and improved numerical algorithms, Mathematical and Computer Modelling 28 (4 – 8) (1998) 225 – 245, recent Advances in Contact Mechanics.

[15] J.-S. Pang, D. E. Stewart, Differential variational inequalities, Mathematical Programming 113 (2008) 345 – 424.

[16] I. P. Natanson, Theorie der Funktionen einer reellen Veränderlichen, 4th edition, Harri Deutsch, Zürich, 1977.

[17] M. Gerdts, Optimal Control of ODEs and DAEs, De Gruyter, 2012.

[18] T. Schindler, V. Acary, Timestepping schemes for nonsmooth dynamics based on discontinuous Galerkin methods: Definition and outlook, Mathematics and Computers in Simulation 95 (0) (2014) 180 – 199.

[19] M. Anitescu, Optimization-based simulation of nonsmooth rigid multibody dynamics, Mathematical Programming 105 (1) (2005) 113 – 143.

[20] A. Tasora, M. Anitescu, A matrix-free cone complementarity approach for solving large-scale, nonsmooth, rigid body dynamics, Computer Methods in Applied Mechanics and Engineering 200 (2011) 439 – 453.

[21] R. Kučera, J. Machalová, H. Netuka, P. Ženčák, An interior-point algorithm for the minimization arising from 3d contact problems with friction, Optimization Methods and Software 28 (6) (2013) 1195 – 1217.