An iterative solver for cone complementarity problems of nonsmooth multibody dynamics—and other DVI

Mihai ANITESCU ,

Argonne National Laboratories

Hybrid System Workshop, Koç University, May 2008

Team

- Mihai Anitescu
- Alessandro Tasora
  - University of Parma,
  - Author of ChronoEngine
- Dan Negrut,
  - University of Wisconsin
  - Former ADAMS developer

Nonsmooth contact dynamics—what is it?

- Differential problem with variational inequality constraints – DVI

Newton Equations

Non-Penetration Constraints

Generalized Velocities

$$M\frac{dv}{dt} = \sum_{j=1,2,\ldots,p}\left(c_n^{(j)}n^{(j)} + \beta_1^{(j)}t_1^{(j)} + \beta_2^{(j)}t_2^{(j)}\right) + f_c(q,v) + k(t,q,v)$$

$$\frac{dq}{dt} = \Gamma(q)v$$

$$c_n^{(j)} \geq 0 \quad \perp \quad \Phi^{(j)}(q) \geq 0, \qquad j = 1,2,\ldots,p$$

$$\left(\beta_1^{(j)},\beta_2^{(j)}\right) = \operatorname*{argmin}_{\mu^{(j)}c_n^{(j)} \geq \sqrt{\left(\beta_1^{(j)}+\beta_2^{(j)}\right)^2}} \left[\left(v^T t_1^{(j)}\right)\beta_1 + \left(v^T t_2^{(j)}\right)\beta_2\right]$$

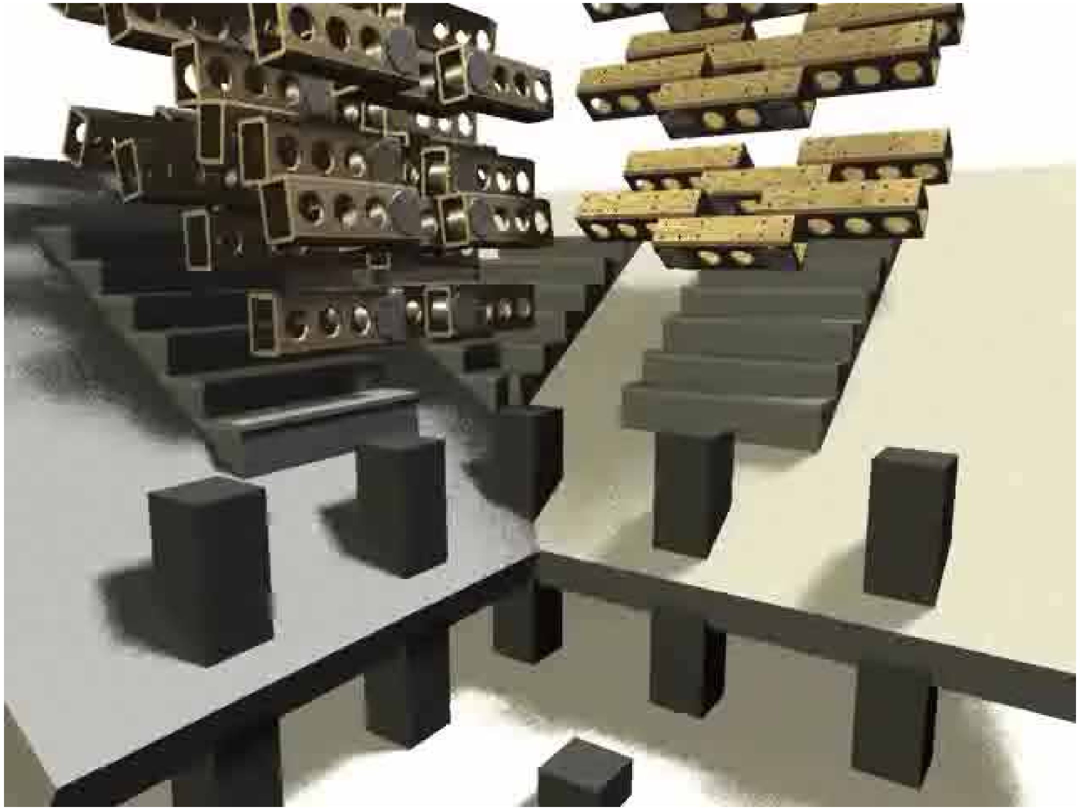
Friction Model

- Truly, a Differential Problem with Equilibrium Constraints

Differential Variational Inequalities– why do it?

- Contact Dynamics.
  - Rigid-Bodies: Differential Operator is ODE.
  - Deformable Bodies: Differential Operator is PDE.
  - Granular Flow, Masonry Stability, Rock Dynamics…
- Finance: Option Pricing-- American Options. PDE-based.
- Dynamics of multicristalline materials: evolution of the boundary between phases.
- Porous Media Flow.
- See Luo, Pang et al, and Kinderlehrer and Stampacchia Monographs..

Or, just for fun .... Physics-based VR



Note: real-time simulation

- Implication: Speed and Stability more weight than of accuracy.

- This “fun” is serious business in the US,
- One of the main drivers of new architectures (GPU, Ageia); huge user community

Question 1: Should we do smoothing?

- Recall, DVI (for C=R+)

$$\dot{x} = f\left(t, x(t), u(t)\right);$$
$$u \geq 0 \perp F\left(t, x(t), u(t)\right) \geq 0$$

- Smoothing

$$\dot{x} = f\left(t, x(t), u(t)\right);$$
$$u_i\ F_i\left(t, x(t), u(t)\right) = \varepsilon, \quad i = 1, 2, \ldots n_u$$

- Followed by forward Euler. Easy to implement!!

$$u_i^n\ F_i\left(t^{n-1}, x^{n-1}, u^{n-1}\right) = \varepsilon, \quad i = 1, 2, \ldots n_u$$
$$x^{n+1} = x^n + hf\left(t^n, x^n, u^n\right);$$

- Compare with the complexity of time-stepping

$$x^{n+1} = x^n + hf\left(t^{n+1}, x^{n+1}, u^{n+1}\right);$$
$$u^{n+1} \geq 0 \perp F\left(t^{n+1}, x^{n+1}, u^{n+1}\right) \geq 0$$

- But does it give good results?

*Applying ADAMS to granular flow**

** From Madsen et al.*

- ADAMS is the workhorse of engineering dynamics.
- ADAMS/View Procedure for simulating.
- Spheres: diameter of 60 mm and a weight of 0.882 kg.
- Forces:smoothing with stiffness of 1E5, force exponent of 2.2, damping coefficient of 10.0, and a penetration depth of 0.1

ChronoIrrlicht collision avicompressed

balls_dropping

ADAMS versus ChronoEngine *

* From Madsen et al.

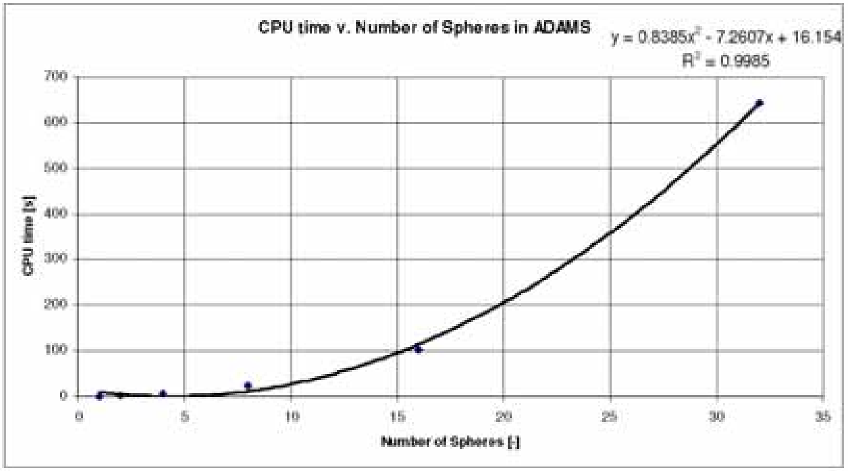
Table 1: Number of rigid bodies v. CPU time in ADAMS

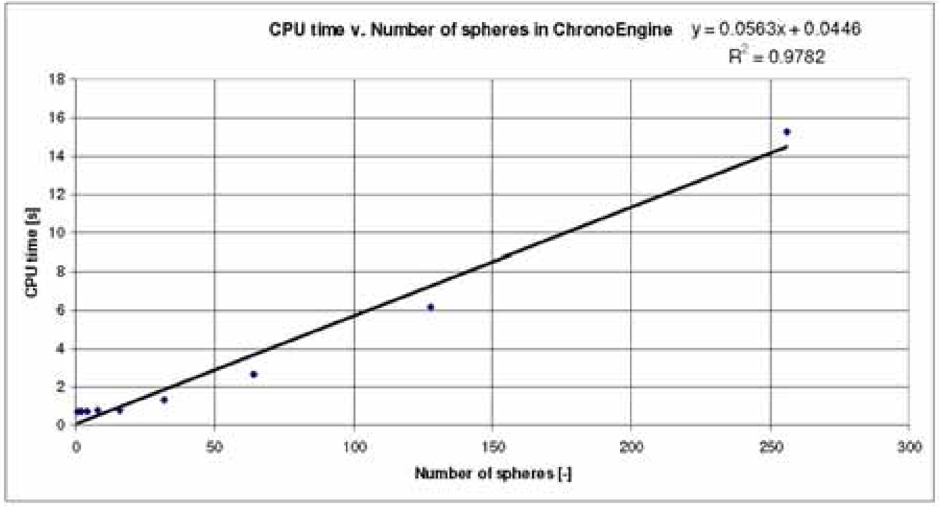
Number of Spheres	Max Number of Mutual Contacts [-]	CPU time (seconds)
1	1	0.41
2	3	3.3
4	14	7.75
8	44	25.36
16	152	102.78
32	560	644.4

The following graph shows the nonlinear increase in the CPU time as the number of colliding bodies increases.

Table 2: Number of rigid bodies v. CPU time in ChronoEngine

Number of Spheres	Max Number of Mutual Contacts [-]	CPU time (seconds)
1	1	0.70
2	3	0.73
4	14	0.73
8	44	0.76
16	152	0.82
32	560	1.32
64	2144	2.65
128	8384	6.17
256	33152	15.30





Conclusion 1: Often, time stepping is more promising,

Nonsmooth contact dynamics

- Differential problem with equilibrium constraints – DPEC.

$$
\begin{aligned}
M\frac{dv}{dt} &= \sum_{j=1,2,..,p}\left(c_n^{(j)}n^{(j)} + \beta_1^{(j)}t_1^{(j)} + \beta_2^{(j)}t_2^{(j)}\right) + f_c(q,v) + k(t,q,v)\\
\frac{dq}{dt} &= v\\
c_n^{(j)} &\geq 0 \quad \perp \quad \Phi^{(j)}(q) \geq 0, \qquad j = 1,2,..,p\\
\left(\beta_1^{(j)}, \beta_2^{(j)}\right) &= \operatorname*{argmin}_{\mu^{(j)}c_n^{(j)} \geq \sqrt{\left(\beta_1^{(j)}+\beta_2^{(j)}\right)^2}} \left[\left(v^T t_1^{(j)}\right)\beta_1 + \left(v^T t_2^{(j)}\right)\beta_2\right]
\end{aligned}
$$

Friction Model

Where is the switching?

- When bodies enter contact (collision, plastic in the previous formulation)
- Stick-Slip transition.

Options and challenges for methods with no smoothing

- Piecewise DAE (Haug, 86)
  - Plus : Uses well understood DAE technology
  - Minus: The density of switches, switching consistency, and Painleve are problems.
- Acceleration-force time-stepping (Glocker & Pfeiffer, 1992, Pang & Trinkle, 1995)
  - Plus: No consistency problem.
  - Minus: Density of switches and Painleve.
- Velocity-impulse time-stepping. (Moreau, 196*, 198*,199*, Stewart and Trinkle, 1996, Anitescu & Potra, 1997)
  - Plus: No consistency, or Painleve. Some have fixed time stepping (Moreau, 198*, Anitescu & Hart 04, Anitescu, 06).
  - Minus: Nonzero restitution coefficient is tough—but its value is disputable in any case

Conic Complementarity IS NATURAL in Coulomb Models.

- Coulomb model.

$$\left(\beta_1^{(j)},\beta_2^{(j)}\right)=\operatorname{argmin}_{\mu^{(j)}c_n^{(j)}\geq\sqrt{\left(\beta_1^{(j)}+\beta_2^{(j)}\right)^2}}\left[\left(v^T t_1^{(j)}\right)\beta_1+\left(v^T t_2^{(j)}\right)\beta_2\right]$$

$$K=\left\{(x,y,z)\middle|\mu^{(j)}z\geq\sqrt{y^2+x^2}\right\}\quad K^*=\left\{(x,y,z)\middle|z\geq\mu^{(j)}\sqrt{y^2+x^2}\right\}$$

$$\begin{pmatrix}c_n^{(j)}\\ \beta_1^{(j)}\\ \beta_2^{(j)}\end{pmatrix}\in K\quad\perp\quad\begin{pmatrix}\mu^{(j)}\sqrt{\left(v^T t_1^{(j)}\right)^2+\left(v^T t_2^{(j)}\right)^2}\\ v^T t_1^{(j)}\\ v^T t_2^{(j)}\end{pmatrix}\in K^*$$

- Most previous approaches discretize friction cone to use LCP…
- Question 2: Can we still get convergence but not do that?

Time stepping scheme -- original

- A measure differential inclusion solution can be obtained by time-stepping (Stewart, 1998, Anitescu 2006)

$$M(\boldsymbol{v}^{(l+1)} - \boldsymbol{v}^{l}) = \sum_{i \in \mathcal{A}(q^{(l)},\epsilon)} (\gamma_n^i \boldsymbol{D}_n^i + \gamma_u^i \boldsymbol{D}_u^i + \gamma_v^i \boldsymbol{D}_v^i) +$$

$$+ \sum_{i \in \mathcal{G}_{\mathcal{B}}} (\gamma_b^i \nabla \Psi^i) + h \boldsymbol{f}_t(t^{(l)}, \boldsymbol{q}^{(l)}, \boldsymbol{v}^{(l)})$$

Speeds

Reaction impulses

Forces

Stabilization terms

$$0 = \frac{1}{h}\Psi^i(\boldsymbol{q}^{(l)}) + \nabla \Psi^{i^T} \boldsymbol{v}^{(l+1)} + \frac{\partial \Psi^i}{\partial t}, \quad i \in \mathcal{G}_{\mathcal{B}}$$

Bilateral constraint equations

$$0 \leq \frac{1}{h}\Phi^i(\boldsymbol{q}^{(l)}) + \nabla \Phi^{i^T} \boldsymbol{v}^{(l+1)}$$

Contact constraint equations

$$\perp \quad \gamma_n^i \geq 0, \ i \in \mathcal{A}(q^{(l)}, \epsilon)$$

COMPLEMENTARITY!

$$(\gamma_u^i, \gamma_v^i) = \operatorname{argmin}_{\mu^i \gamma_n^i \geq \sqrt{(\gamma_u^i)^2 + (\gamma_v^i)^2}} \quad i \in \mathcal{A}(q^{(l)}, \epsilon)$$

$$\left[\boldsymbol{v}^T (\gamma_u \boldsymbol{D}_u^i + \gamma_v \boldsymbol{D}_v^i)\right]$$

Coulomb 3D friction model

$$\boldsymbol{q}^{(l+1)} = \boldsymbol{q}^{(l)} + h \boldsymbol{v}^{(l+1)},$$

Pause: Constraint Stabilization

- Compared to original scheme

$$\nabla\Phi(q^{(l)})^T v^{(l+1)} \geq 0 \Longrightarrow \Phi^{(j)}(q^{(l)}) + \gamma h_l \nabla\Phi(q^{(l)})^T v^{(l+1)} \geq 0.$$

$$\nabla\Theta(q^{(l)})^T v^{(l+1)} = 0 \Longrightarrow \Theta^{(j)}(q^{(l)}) + \gamma h_l \nabla\Theta(q^{(l)})^T v^{(l+1)} = 0.$$

- Allows fixed time steps for plastic collisions.
- How do we know it is achieved? Infeasibility is one order better than accuracy (O(h^2))

Time Stepping -- Convex Relaxation

- A modification (relaxation, to get convex QP with conic constraints):

$$
\begin{aligned}
M(\boldsymbol{v}^{(l+1)} - \boldsymbol{v}^{l}) &= \sum_{i\in\mathcal{A}(q^{(l)},\epsilon)} \left(\gamma_n^i \boldsymbol{D}_n^i + \gamma_u^i \boldsymbol{D}_u^i + \gamma_v^i \boldsymbol{D}_v^i\right) + \\
&\quad + \sum_{i\in\mathcal{G}_{\mathcal{B}}} \left(\gamma_b^i \nabla\Psi^i\right) + h\boldsymbol{f}_t(t^{(l)}, \boldsymbol{q}^{(l)}, \boldsymbol{v}^{(l)}) \\
0 &= \frac{1}{h}\Psi^i(\boldsymbol{q}^{(l)}) + \nabla\Psi^{i^T}\boldsymbol{v}^{(l+1)} + \frac{\partial\Psi^i}{\partial t}, \quad i \in \mathcal{G}_{\mathcal{B}} \\
0 &\le \frac{1}{h}\Phi^i(\boldsymbol{q}^{(l)}) + \nabla\Phi^{i^T}\boldsymbol{v}^{(l+1)} - \mu^i\sqrt{(\boldsymbol{D}_u^{i,T}\boldsymbol{v})^2 + (\boldsymbol{D}_v^{i,T}\boldsymbol{v})^2} \\
&\perp \quad \gamma_n^i \ge 0, \; i \in \mathcal{A}(q^{(l)},\epsilon) \\
(\gamma_u^i, \gamma_v^i) &= \operatorname{argmin}_{\mu^i\gamma_n^i \ge \sqrt{(\gamma_u^i)^2 + (\gamma_v^i)^2}} \quad i \in \mathcal{A}(q^{(l)},\epsilon) \\
&\quad \left[\boldsymbol{v}^T(\gamma_u \boldsymbol{D}_u^i + \gamma_v \boldsymbol{D}_v^i)\right] \\
\boldsymbol{q}^{(l+1)} &= \boldsymbol{q}^{(l)} + h\boldsymbol{v}^{(l+1)},
\end{aligned}
$$

(For small μ and/or small speeds, almost no one-step differences from the Coulomb theory)

But In any case, converges to same MDI as unrelaxed scheme.

[see M.Anitescu, "Optimization Based Simulation of Nonsmooth Rigid Body Dynamics"]

Pause: what does convergence mean here?

We must now assign a meaning to

$$M\frac{dv}{dt} - f_c(q,v) - k(t,q,v) \in FC(q).$$

Definition If ν is a measure and $K(\cdot)$ is a convex-set valued mapping, we say that v satisfies the differential inclusions

$$\frac{dv}{dt} \in K(t)$$

if, for all continuous $\phi \geq 0$ with compact support, not identically 0, we have that

$$\frac{\int \phi(t)\nu(dt)}{\int \phi(t)dt} \in \bigcup_{\tau:\phi(\tau)\neq 0} K(\tau).$$

Pause(2) : What does convergence mean here?

H1 The functions $n^{(j)}(q), t_1^{(j)}(q), t_2^{(j)}(q)$ are smooth and globally Lipschitz, and they are bounded in the 2-norm.

H2 The mass matrix M is positive definite.

H3 The external force increases at most linearly with the velocity and position.

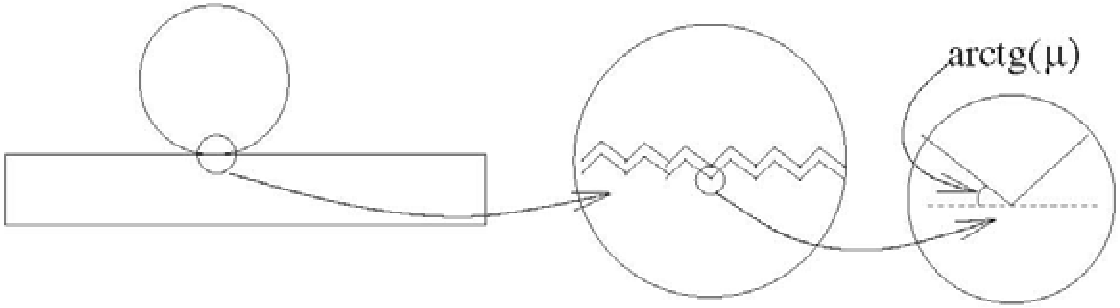
H4 The uniform pointed friction cone assumption holds.

Then there exists a subsequence $h_k \to 0$ where

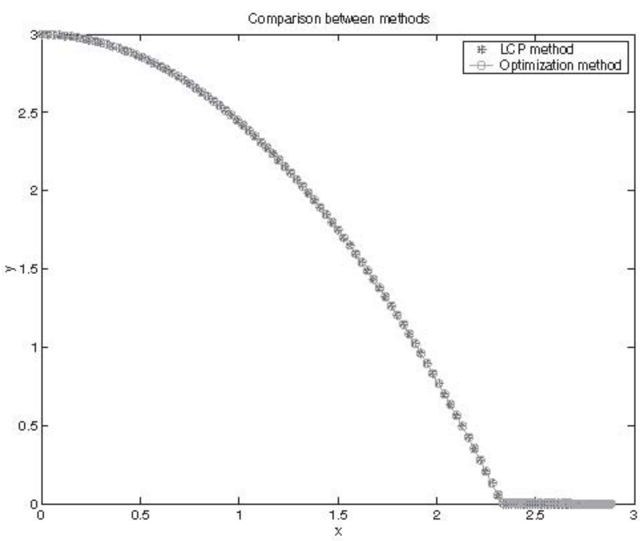
- $q^{h_k}(\cdot) \to q(\cdot)$ uniformly.
- $v^{h_k}(\cdot) \to v(\cdot)$ pointwise a.e.
- $dv^{h_k}(\cdot) \to dv(\cdot)$ weak * as Borel measures. in [0,T], and every such subsequence converges to a solution $(q(\cdot), v(\cdot))$ of MDI.

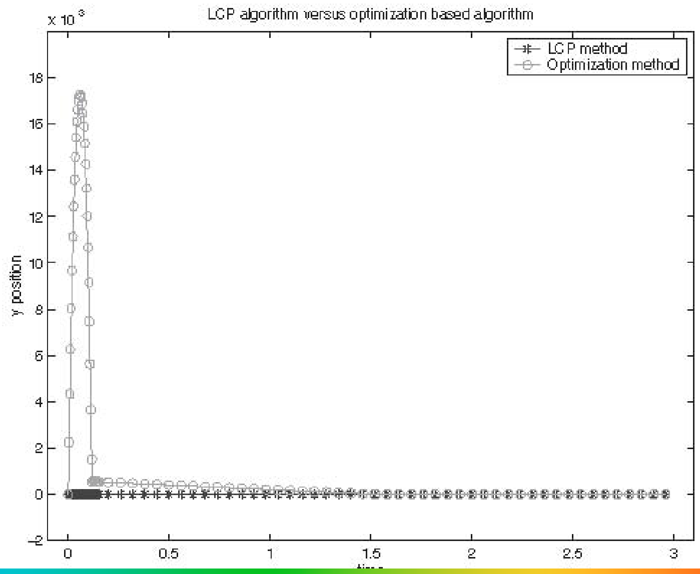
What is physical meaning of the relaxation?

- Origin



- Behavior





Further insight.

- The key is the combination between relaxation and constraint stabilization.

$$0 \leq \frac{1}{h}\Phi^{(j)}\left(q^{(l)}\right)+\nabla_q \Phi^{(j)}\left(q^{(l)}\right)v^{(l+1)}-\mu^{(j)}\sqrt{\left(D_u^{l,t}v\right)^2+\left(D_v^{l,t}v\right)$$

- If the time step is smaller than the variation in velocity then the gap function settles at

$$0 \approx \frac{1}{h}\Phi^{(j)}\left(q^{(l)}\right)-\mu^{(j)}\sqrt{\left(D_u^{l,t}v\right)^2+\left(D_v^{l,t}v\right)^2}$$

- So the solution is the same as the original scheme for a slightly perturbed gap function…..

Cone complementarity*

* Anitescu and Tasora "An iterative approach for cone complementarity problems for nonsmooth dynamics". Preprint ANL/MCS-P1413-0507

- Aiming at a more compact formulation:

$$\boldsymbol{b}_{\mathcal{A}} = \left\{ \frac{1}{h}\Phi^{i_1}, 0, 0, \frac{1}{h}\Phi^{i_2}, 0, 0, \ldots, \frac{1}{h}\Phi^{i_{n_\mathcal{A}}}, 0, 0 \right\}$$

$$\boldsymbol{\gamma}_{\mathcal{A}} = \left\{ \gamma_n^{i_1}, \gamma_u^{i_1}, \gamma_v^{i_1}, \gamma_n^{i_2}, \gamma_u^{i_2}, \gamma_v^{i_2}, \ldots, \gamma_n^{i_{n_\mathcal{A}}}, \gamma_u^{i_{n_\mathcal{A}}}, \gamma_v^{i_{n_\mathcal{A}}} \right\}$$

$$\boldsymbol{b}_{\mathcal{B}} = \left\{ \frac{1}{h}\Psi^1 + \frac{\partial \Psi^1}{\partial t}, \frac{1}{h}\Psi^2 + \frac{\partial \Psi^2}{\partial t}, \ldots, \frac{1}{h}\Psi^{n_\mathcal{B}} + \frac{\partial \Psi^{n_\mathcal{B}}}{\partial t} \right\}$$

$$\boldsymbol{\gamma}_{\mathcal{B}} = \left\{ \gamma_b^1, \gamma_b^2, \ldots, \gamma_b^{n_\mathcal{B}} \right\}$$

$$D_{\mathcal{A}} = \left[D^{i_1} | D^{i_2} | \ldots | D^{i_{n_\mathcal{A}}} \right], \quad i \in \mathcal{A}(\boldsymbol{q}^l, \epsilon) \qquad D^i = [\boldsymbol{D}_n^i | \boldsymbol{D}_u^i | \boldsymbol{D}_v^i]$$

- $D_{\mathcal{B}} = \left[\nabla\Psi^{i_1} | \nabla\Psi^{i_2} | \ldots | \nabla\Psi^{i_{n_\mathcal{B}}} \right], \quad i \in \mathcal{G}_{\mathcal{B}}$

$$\boldsymbol{b}_{\mathcal{E}} \in \mathbb{R}^{n_\mathcal{E}} = \{ \boldsymbol{b}_{\mathcal{A}}, \boldsymbol{b}_{\mathcal{B}} \}$$

$$\boldsymbol{\gamma}_{\mathcal{E}} \in \mathbb{R}^{n_\mathcal{E}} = \{ \boldsymbol{\gamma}_{\mathcal{A}}, \boldsymbol{\gamma}_{\mathcal{B}} \}$$

$$D_{\mathcal{E}} = [D_{\mathcal{A}} | D_{\mathcal{B}}]$$

Cone complementarity

- Also define:

$$\tilde{\boldsymbol{k}}^{(l)} = M\boldsymbol{v}^{(l)} + h\boldsymbol{f}_t(t^{(l)}, \boldsymbol{q}^{(l)}, \boldsymbol{v}^{(l)})$$

$$N = D_{\mathcal{E}}^T M^{-1} D_{\mathcal{E}}$$

$$\boldsymbol{r} = D_{\mathcal{E}}^T M^{-1} \tilde{\boldsymbol{k}} + \boldsymbol{b}_{\mathcal{E}}$$

- Then:

$$\begin{aligned}
M(\boldsymbol{v}^{(l+1)} - \boldsymbol{v}^{l}) &= \sum_{i \in \mathcal{A}(q^{(l)},\epsilon)} (\gamma_n^i \boldsymbol{D}_n^i + \gamma_u^i \boldsymbol{D}_u^i + \gamma_v^i \boldsymbol{D}_v^i) + \\
&\quad + \sum_{i \in \mathcal{G}_{\mathcal{B}}} (\gamma_b^i \nabla \Psi^i) + h\boldsymbol{f}_t(t^{(l)}, \boldsymbol{q}^{(l)}, \boldsymbol{v}^{(l)}) \\
0 &= \frac{1}{h}\Psi^i(\boldsymbol{q}^{(l)}) + \nabla {\Psi^i}^T \boldsymbol{v}^{(l+1)} + \frac{\partial \Psi^i}{\partial t}, \quad i \in \mathcal{G}_{\mathcal{B}} \\
0 &\leq \frac{1}{h}\Phi^i(\boldsymbol{q}^{(l)}) + \nabla {\Phi^i}^T \boldsymbol{v}^{(l+1)} \\
&\quad \perp \quad \gamma_n^i \geq 0, \ i \in \mathcal{A}(q^{(l)}, \epsilon) \\
(\gamma_u^i, \gamma_v^i) &= \operatorname{argmin}_{\mu^i \gamma_n^i \geq \sqrt{(\gamma_u^i)^2 + (\gamma_v^i)^2}} \quad i \in \mathcal{A}(q^{(l)}, \epsilon) \\
&\quad [\boldsymbol{v}^T(\gamma_u \boldsymbol{D}_u^i + \gamma_v \boldsymbol{D}_v^i)]
\end{aligned}$$

This is a CCP,
CONE COMPLEMENTARITY PROBLEM

becomes..

$$(N\gamma_{\mathcal{E}} + \boldsymbol{r}) \in -\Upsilon^{\circ} \quad \perp \quad \gamma_{\mathcal{E}} \in \Upsilon$$

Cone complementarity—Decomposable cones.

- Here we introduced the convex cone

$$\Upsilon = \left(\bigoplus_{i \in \mathcal{A}(\boldsymbol{q}^l, \epsilon)} \mathcal{FC}^i \right) \bigoplus \left(\bigoplus_{i \in \mathcal{G}_\mathcal{B}} \mathcal{BC}^i \right)$$

$\mathcal{FC}^i$ In R^3 is *i*-th friction cone

$\mathcal{BC}^i$ is R

- ..and its polar cone:

$$\Upsilon^\circ = \left(\bigoplus_{i \in \mathcal{A}(\boldsymbol{q}^l, \epsilon)} \mathcal{FC}^{i\circ} \right) \bigoplus \left(\bigoplus_{i \in \mathcal{G}_\mathcal{B}} \mathcal{BC}^{i\circ} \right)$$

CCP: $(N\boldsymbol{\gamma}_\mathcal{E} + \boldsymbol{r}) \in -\Upsilon^\circ \quad \perp \quad \boldsymbol{\gamma}_\mathcal{E} \in \Upsilon$

General: The iterative method

- Question 3: How to efficiently solve the Cone Complementarity Problem for large-scale systems?

$$(N\boldsymbol{\gamma}_{\mathcal{E}} + \boldsymbol{r}) \in -\Upsilon^{\circ} \quad \perp \quad \boldsymbol{\gamma}_{\mathcal{E}} \in \Upsilon$$

- Our method: use a fixed-point iteration

$$\gamma^{r+1} = \lambda \Pi_{\Upsilon} \left(\gamma^{r} - \omega B^{r} \left(N\gamma^{r} + \boldsymbol{r} + K^{r} \left(\gamma^{r+1} - \gamma^{r} \right) \right) \right) + (1-\lambda)\, \gamma^{r}$$

- with matrices:
- ..and a non-extensive orthogonal projection operator onto feasible set

$$B^{r} = \begin{bmatrix} \eta_1 I_{n_1} & 0 & \cdots & 0 \\ 0 & \eta_2 I_{n_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \eta_{n_k} I_{n_{n_k}} \end{bmatrix}$$

$$N^{T} = \begin{bmatrix} 0 & K_{12} & K_{13} & \cdots & K_{1n_k} \\ 0 & 0 & K_{23} & \cdots & K_{2n_k} \\ 0 & 0 & 0 & \cdots & K_{3n_k} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ 0 & 0 & 0 & & 0 \end{bmatrix}$$

$$\Pi_{\Upsilon} : \mathbb{R}^{n_{\mathcal{E}}} \rightarrow \mathbb{R}^{n_{\mathcal{E}}}$$

General: The iterative method

- ASSUMPTIONS

A1 The matrix N of the problem (CCP) is symmetric and positive semi-definite.

A2 There exists a positive number, $\alpha > 0$ such that, at any iteration r, $r = 0, 1, 2, \ldots$, we have that $B^r \succ \alpha I$

A3 There exists a positive number, $\beta > 0$ such that, at any iteration r, $r = 0, 1, 2, \ldots$, we have that $(x^{r+1} - x^r)^T \left((\lambda \omega B^r)^{-1} + K^r - \frac{N}{2} \right) (x^{r+1} - x^r) \geq \beta \left\| x^{r+1} - x^r \right\|^2$.

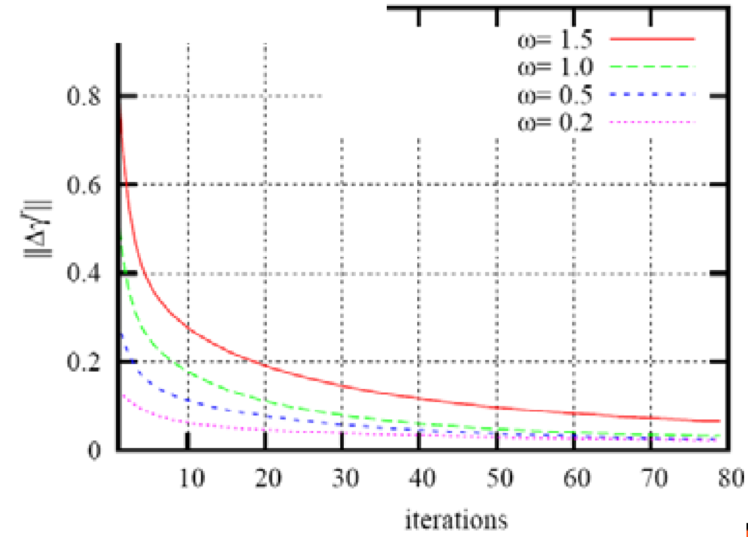
Always satisfied in multibody systems

Essentially free choice, we use identity blocks

Use ω overrelaxation factor to adjust this

- Under the above assumptions, we can prove THEOREMS about convergence.

- The method produces a bounded sequence with an unique accumulation point.



General: Theory

$$(OC) \quad \begin{array}{lll} \min & f(x) = \frac{1}{2} x^T N x + r^T x & \\ s.t. & x_i \in \Upsilon^i, & i = 1, 2, \ldots, n_k. \end{array}$$

Theorem Assume that $x^0 \in \Upsilon$ and that the sequences of matrices B^r and K^r are bounded. Then we have that

$$f(x^{r+1}) - f(x^r) \leq -\beta \left\| x^{r+1} - x^r \right\|^2$$

for any iteration index r, and any accumulation point of the sequence x^r is a solution of (CCP).

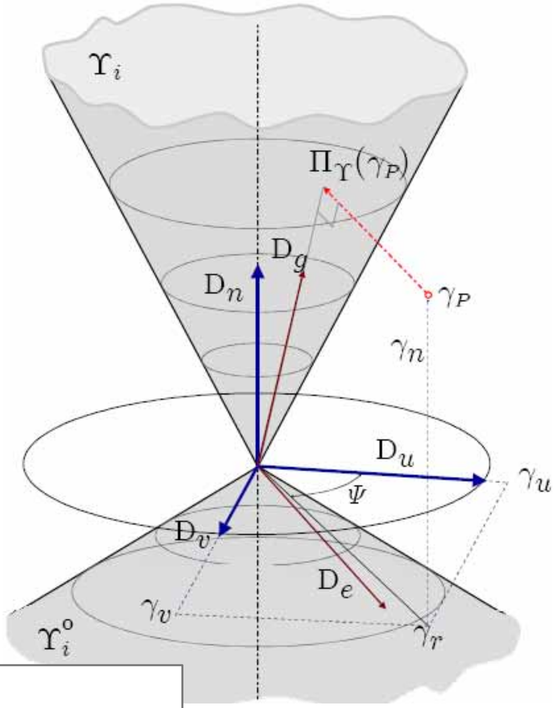
Corollary Assume that the friction cone of the configuration is pointed The algorithm produces a bounded sequence, and any **accumulation point results in the same velocity solution**

- Answer 2: Simple, but first result of this nature for conic constraints—and HIGHLY EFFICIENT

The projection operator is easy and separable

- *For each frictional contact constraint:*

$$\Pi_{\Upsilon} = \left\{\Pi_{\Upsilon_1}(\gamma_1)^T, \ldots \Pi_{\Upsilon_{n_{\mathcal{A}}}}(\boldsymbol{\gamma}^{n_{\mathcal{A}}})^T, \Pi_b^1(\gamma_b^1), \ldots, \Pi_b^{n_{\mathcal{B}}}(\gamma_b^{n_{\mathcal{B}}})\right\}^T$$



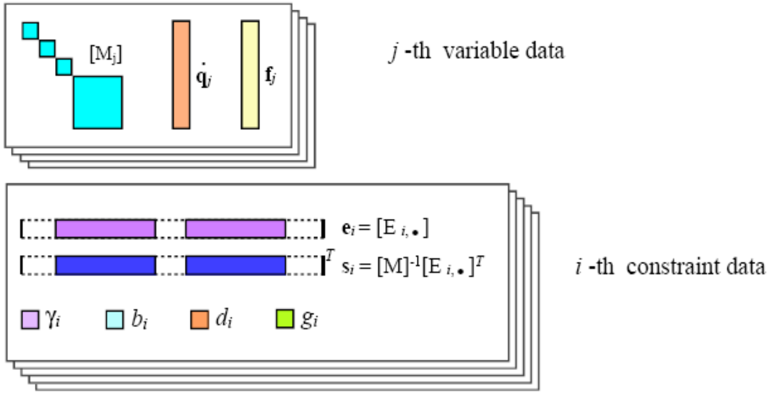
- *For each bilateral constraint, simply do nothing.*
- The complete operator:

$$\forall i \in \mathcal{A}(\boldsymbol{q}^{(l)}, \epsilon)$$

$$\begin{array}{ll}
\gamma_r < \mu_i \gamma_n & \Pi_i = \boldsymbol{\gamma}_i \\
\gamma_r < -\dfrac{1}{\mu_i}\gamma_n & \Pi_i = \{0, 0, 0\} \\
\gamma_r > \mu_i \gamma_n \wedge \gamma_r > -\dfrac{1}{\mu_i}\gamma_n & \Pi_{i,n} = \dfrac{\gamma_r \mu_i + \gamma_n}{\mu_i^2 + 1} \\
 & \Pi_{i,u} = \gamma_u \dfrac{\mu_i \Pi_{i,n}}{\gamma_r} \\
 & \Pi_{i,v} = \gamma_v \dfrac{\mu_i \Pi_{i,n}}{\gamma_r}
\end{array}$$

The algorithm

- Development of an efficient algorithm for fixed point iteration:



- *avoid temporary data, exploit sparsity. Never compute explicitly the N matrix!*
- *implemented in incremental form. Compute only deltas of multipliers.*
- *O(n) space requirements and supports premature termination*
- *for real-time purposes: O(n) time*

The algorithm is specialized, for minimum memory use!

-

// Pre-compute some data for friction constraints

for $i := 1$ **to** $n_{\mathcal{A}}$

$\quad \boldsymbol{s}_a^i = M^{-1} D^i$

$\quad g_a^i = D^{i,T} \boldsymbol{s}_a^i$

$\quad \eta_a^i = \frac{3}{\text{Trace}(g_a^i)}$

// Pre-compute some data for bilateral constraints

for $i := 1$ **to** $n_{\mathcal{B}}$

$\quad s_b^i = M^{-1} \boldsymbol{\nabla \Psi}^i$

$\quad g_b^i = \boldsymbol{\nabla \Psi}^{i,T} s_b^i$

$\quad \eta_b^i = \frac{1}{g_b^i}$

// Initialize impulses

if warm start with initial guess $\boldsymbol{\gamma}_{\mathcal{E}}^*$

$\quad \boldsymbol{\gamma}_{\mathcal{E}}^0 = \boldsymbol{\gamma}_{\mathcal{E}}^*$

else

$\quad \boldsymbol{\gamma}_{\mathcal{E}}^0 = 0$

// Initialize speeds

$\boldsymbol{v} = \sum_{i=1}^{n_{\mathcal{A}}} \boldsymbol{s}_a^i \boldsymbol{\gamma}_a^{i,0} + \sum_{i=1}^{n_{\mathcal{B}}} s_b^i \gamma_b^{i,0} + M^{-1} \tilde{\boldsymbol{k}}$

// Main iteration loop

for $r := 0$ **to** r_{max}

$\quad$ *// Loop on frictional constraints*

$\quad$ **for** $i := 1$ **to** $n_{\mathcal{A}}$

$\quad\quad \boldsymbol{\delta}_a^{i,r} = \left(\boldsymbol{\gamma}_a^{i,r} - \omega \eta_a^i \left(D^{i,T} \boldsymbol{v}^r + \boldsymbol{b}_a^i \right) \right);$

$\quad\quad \boldsymbol{\gamma}_a^{i,r+1} = \lambda \Pi_{\Upsilon} \left(\boldsymbol{\delta}_a^{i,r} \right) + (1-\lambda) \boldsymbol{\gamma}_a^{i,r} \ ;$

$\quad\quad \Delta \boldsymbol{\gamma}_a^{i,r+1} = \boldsymbol{\gamma}_a^{i,r+1} - \boldsymbol{\gamma}_a^{i,r} \ ;$

$\quad\quad \boldsymbol{v} := \boldsymbol{v} + \boldsymbol{s}_a^{i^T} \Delta \boldsymbol{\gamma}_a^{i,r+1}.$

$\quad$ *// Loop on bilateral constraints*

$\quad$ **for** $i := 1$ **to** $n_{\mathcal{B}}$

$\quad\quad \delta_b^{i,r} = \left(\gamma_b^{i,r} - \omega \eta_b^i \left(\nabla \boldsymbol{\Psi}^{i,T} \boldsymbol{v}^r + b_b^i \right) \right);$

$\quad\quad \gamma_b^{i,r+1} = \lambda \Pi_{\Upsilon} \left(\delta_b^{i,r} \right) + (1-\lambda) \gamma_b^{i,r} \ ;$

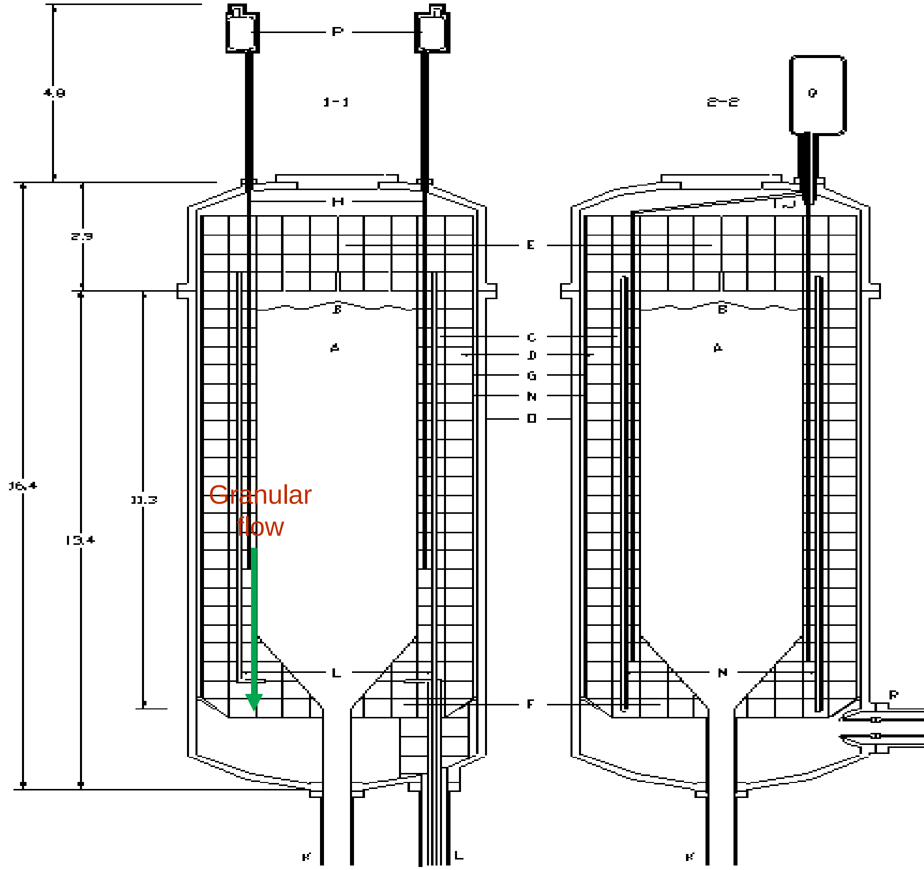
$\quad\quad \Delta \gamma_b^{i,r+1} = \gamma_b^{i,r+1} - \gamma_b^{i,r} \ ;$

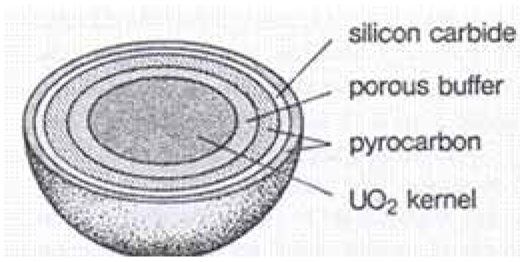
$\quad\quad \boldsymbol{v} := \boldsymbol{v} + s_b^{i^T} \Delta \gamma_b^{i,r+1}.$

return $\boldsymbol{\gamma}_{\mathcal{E}}, \boldsymbol{v}$

Simulating the PBR nuclear reactor

- The PBR nuclear reactor:
  - Fourth generation design
  - Inherently safe, by Doppler broadening of fission cross section
  - Helium cooled > 1000 °C
  - Can crack water (mass production of hydrogen)
  - Continuous cycling of 360’000 graphite spheres in a pebble bed

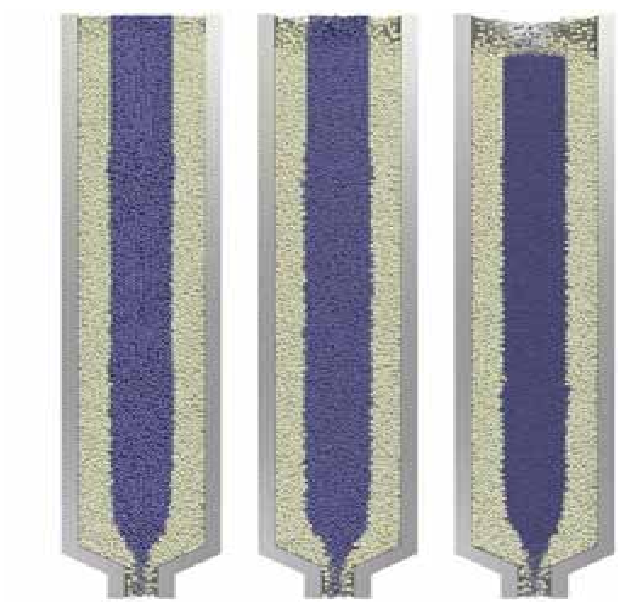




Simulating the PBR nuclear reactor

- Problem of bidisperse granular flow with dense packing.
- Previous attempts: DEM methods on supercomputers at Sandia Labs regularization)
- 40 seconds of simulation for 440,000 pebbles needs 1 week on 64 processors dedicated cluster (Rycroft et al.)

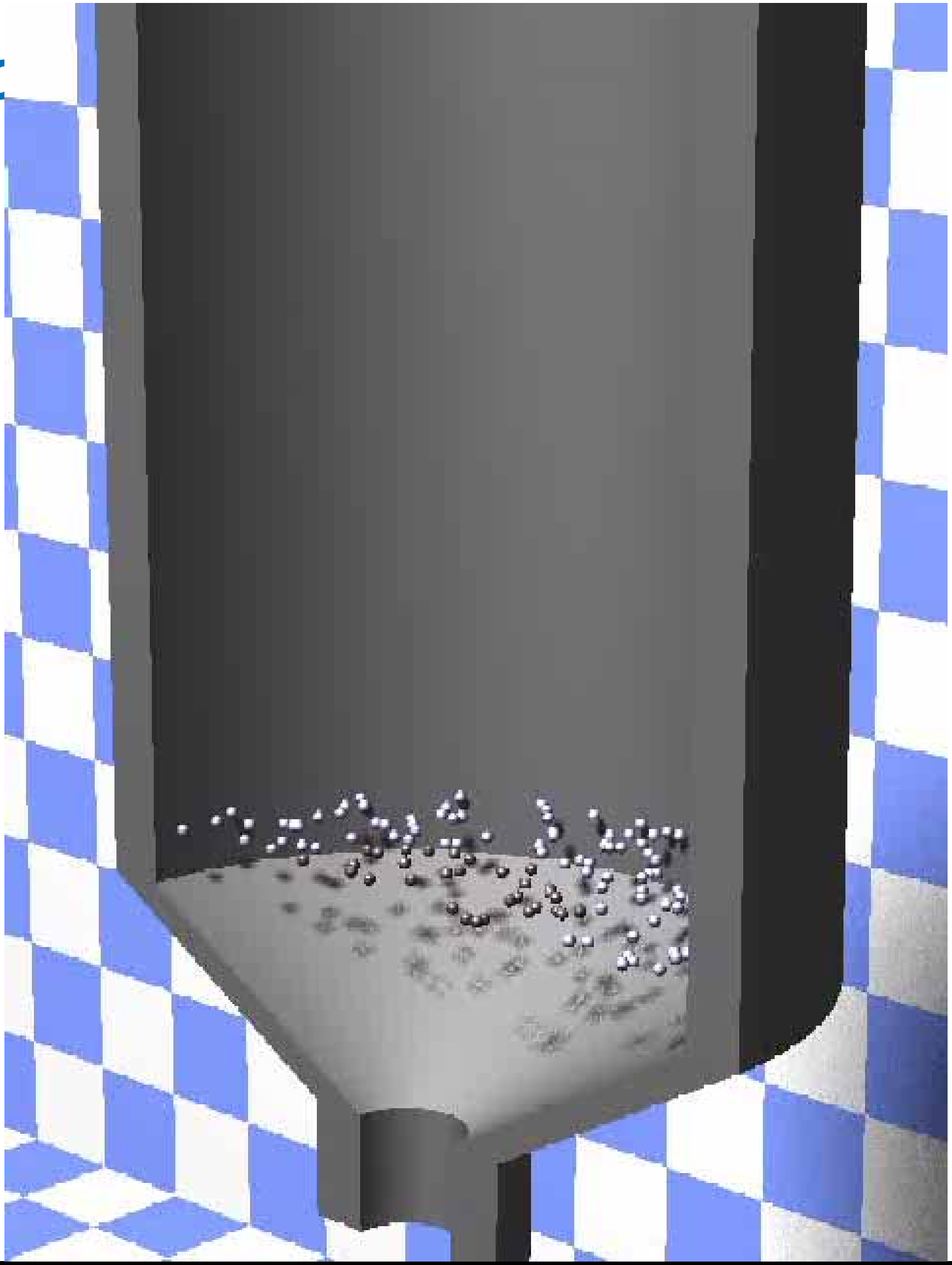
model a frictionless wall, $\mu_w=0.0$. For the current simulations we set $k_t=\frac{2}{7}k_n$ and choose $k_n=2\times10^5\ gm/d$. While this is significantly less than would be realistic for graphite pebbles, where we expect $k_n>10^{10}\ gm/d$, such a spring constant would be prohibitively computationally expensive, as the time step scales as $\delta t\propto k_n^{-1/2}$ for collisions to be modeled effectively. Previous simulations have shown that



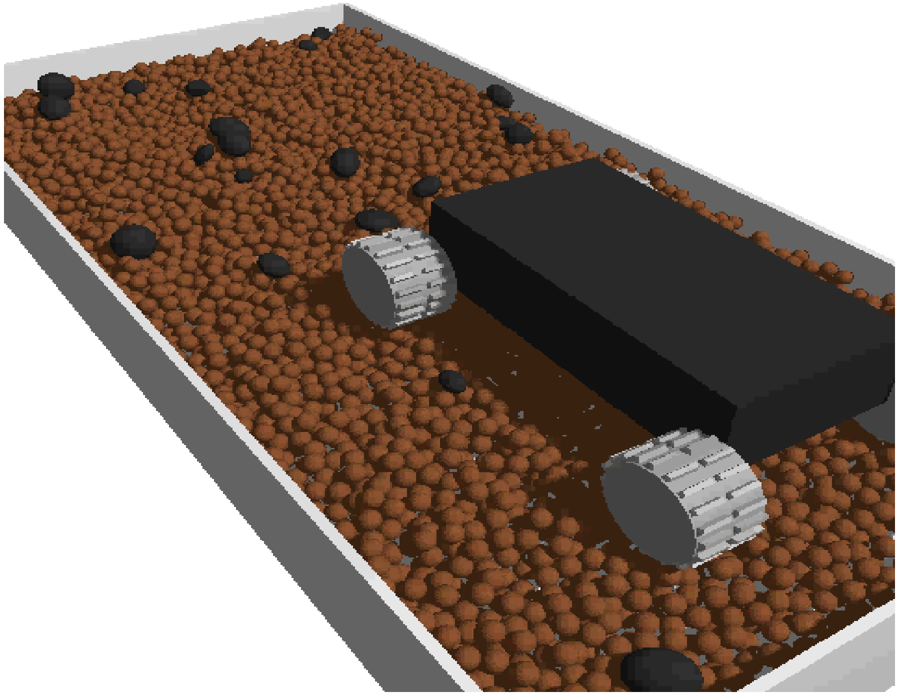
Simulations with DEM. Bazant et al. (MIT and Sandia laboratories).

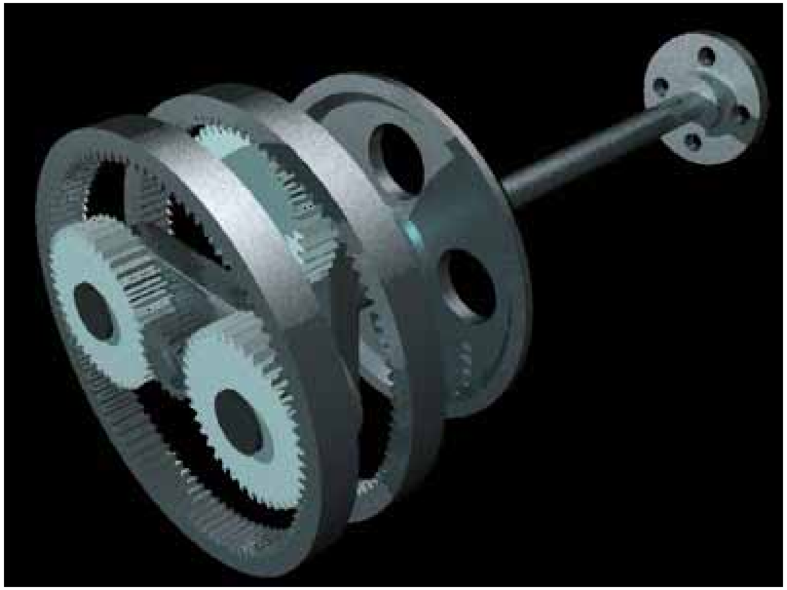
Simulating the PBR nuc

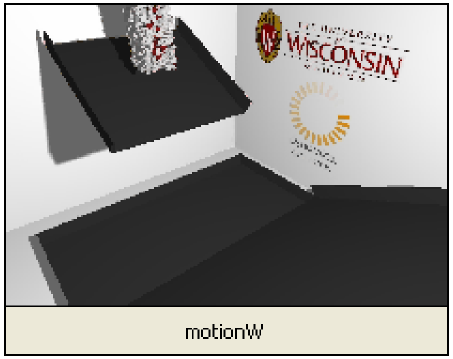
- 160’000 Uranium-Graphite spheres, 600’000 contacts on average
- Two millions of primal variables, six millions of dual variables
- *1 day on a Windows station…*
- But we are limited by the 2GB user mode limit, 64 bit port in progress—but linear scaling..
- We estimate 3CPU days, compare with 450 CPU days for an incomplete solution in 2006 !!!
- Answer 3: Our approach is efficient for large scale!!



In addition, we can approach efficiently approach many engineering problems (see website for papers)

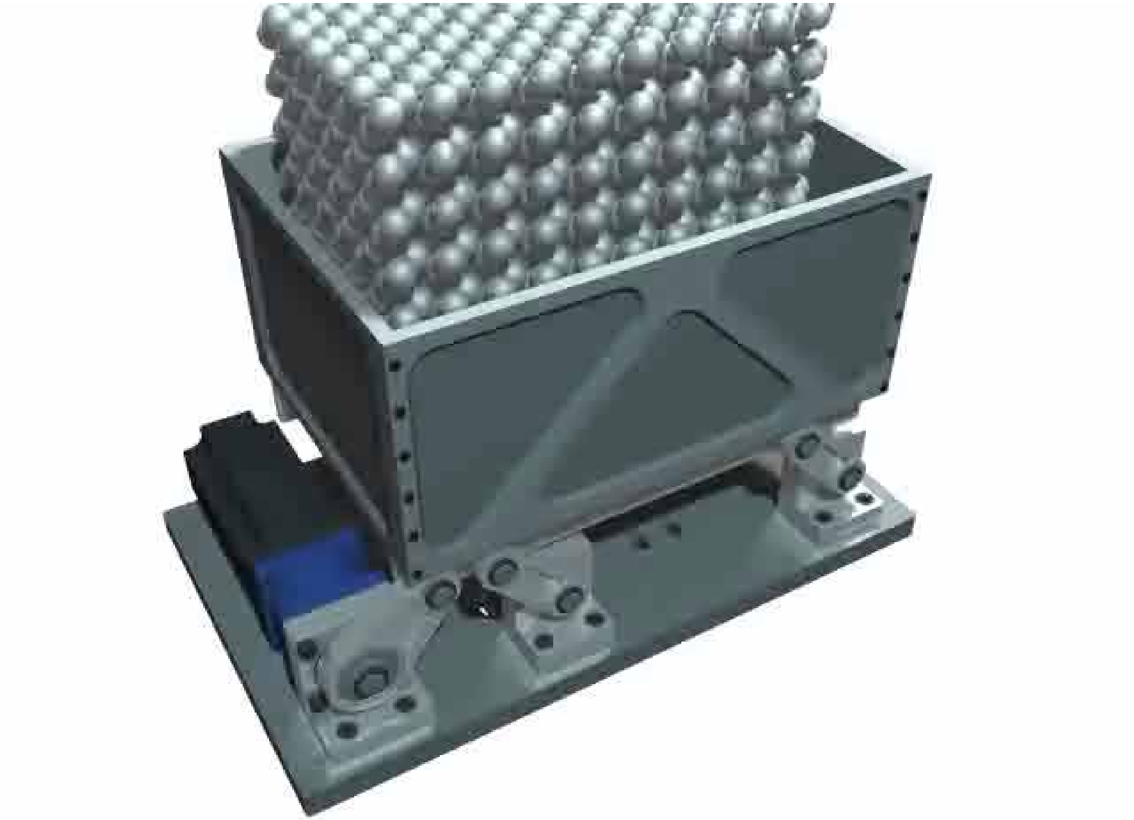






Examples

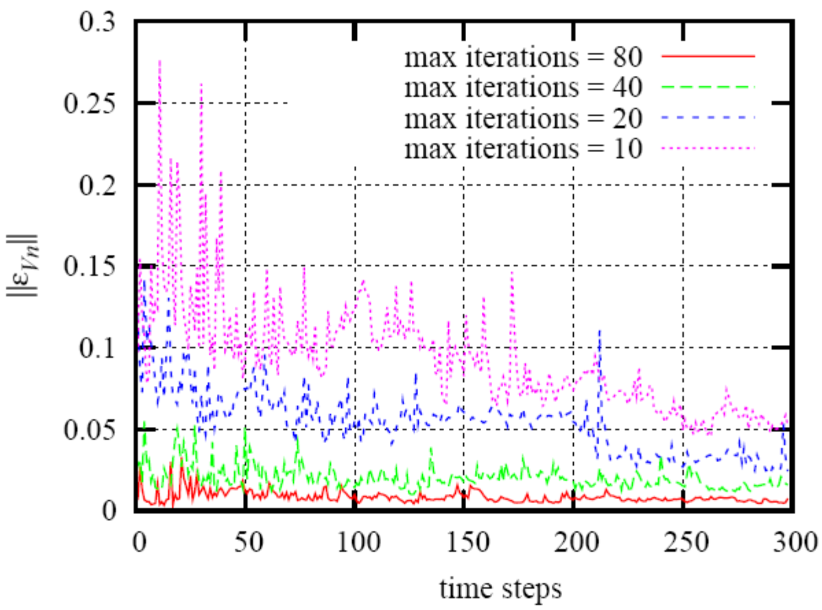
- Example: size-segregation in shaker, with thousands of steel spheres

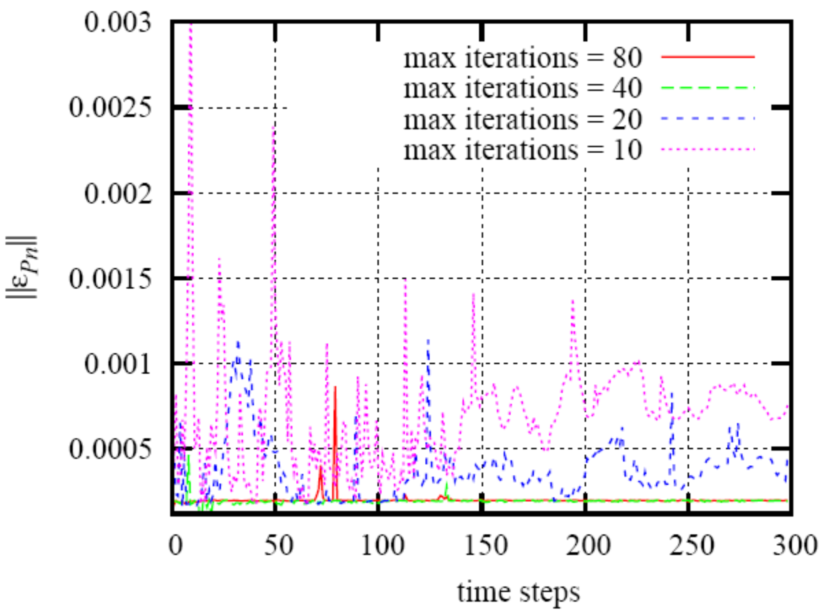


Note: solution beyond reach of Lemke-type LCP solvers!

Tests

- Feasibility accuracy increases with number of iterations:





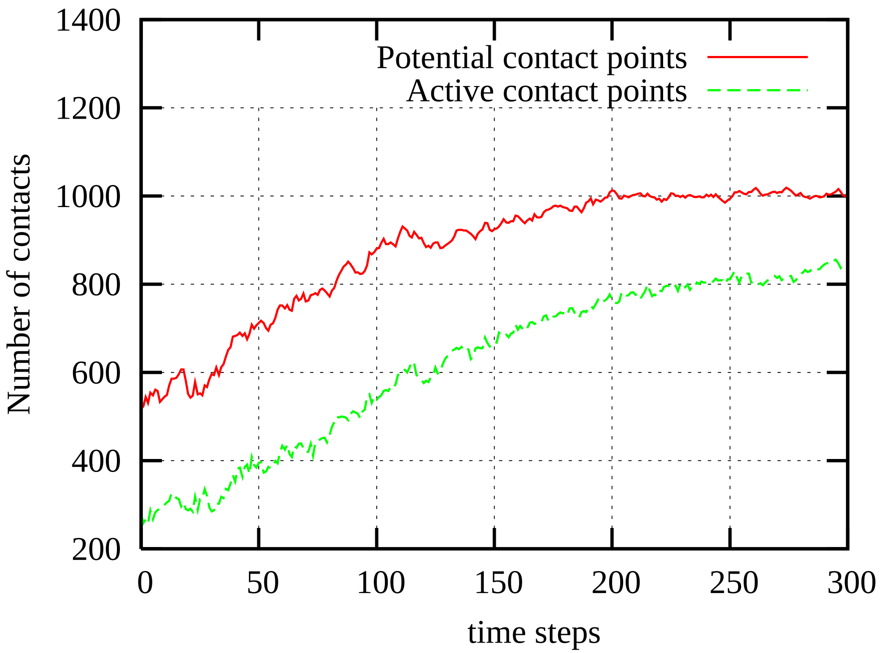
Speed violation in constraints

Position error in constraints (penetration)

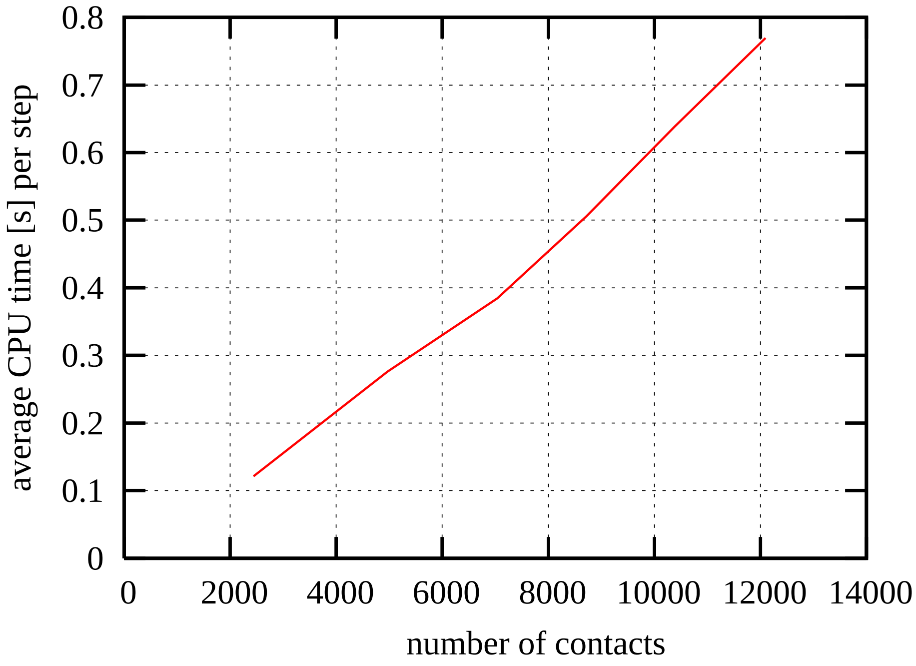
(with example of 300 spheres in shaker)

Tests: Scalability

- CPU effort per contact, since our contacts are the problem variables.
- Penetration error was uniformly no larger than 0.2% of diameter.

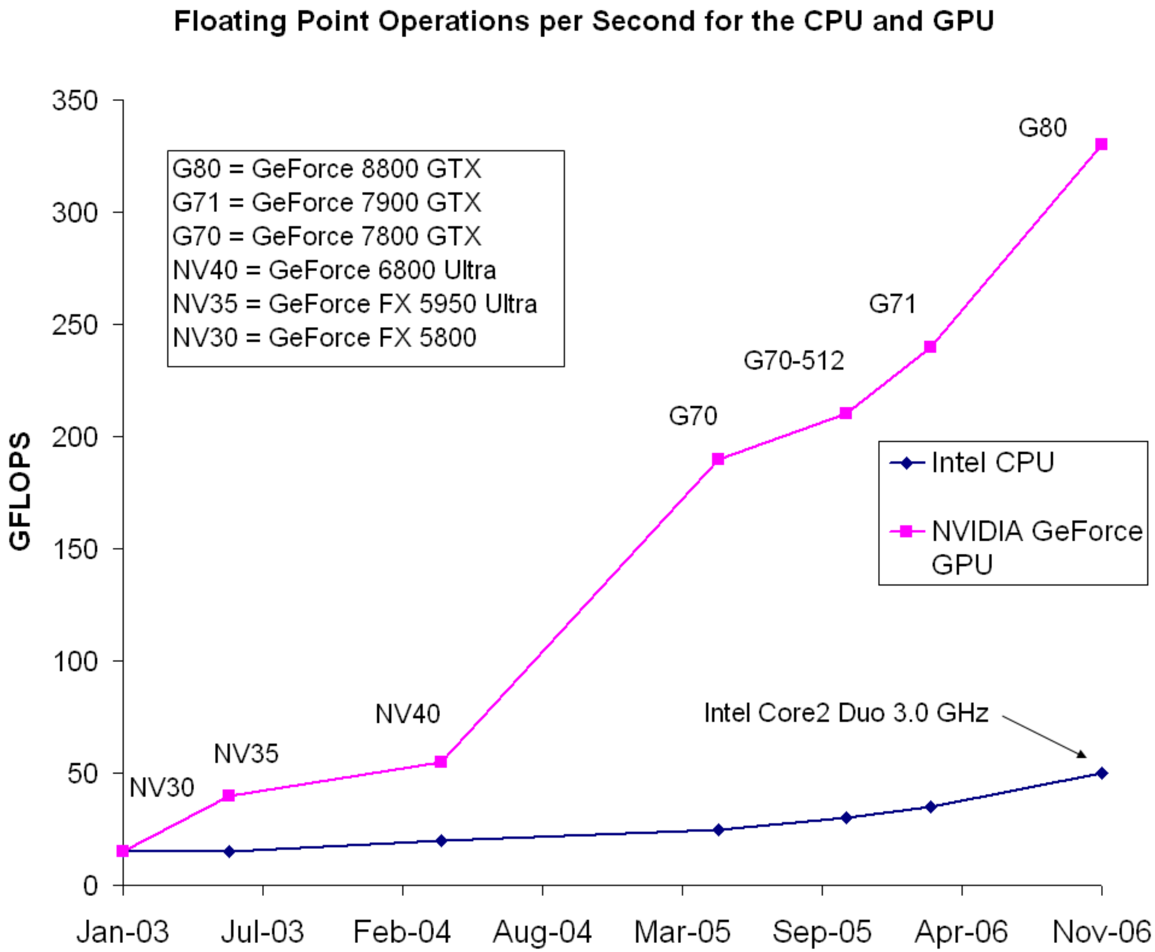


Number of contacts in time, 300 spheres



CPU time per step for 300-1500 spheres

New large scale computational opportunity Graphical Processing Unit *



*NVIDIA CUDA Compute Unified Device Architecture

Programming Guide

IBM BlueGene/L—GPU comparison

- Entry model: 1024 dual core nodes
- 5.7 Tflop (compare to 0.5 Tflop for NVIDIA Tesla GPU)
- Dedicated OS
- Dedicated power management solution
- Require dedicated IT support
- Price (2007): $1.4 million
- Same GPU power (2008): 7K!!!
- Of course, GPU much harder to work with at the moment, and unsuitable for general purpose computing.

*Brick Wall Example **

*[*Alessandro Tasora, Dan Negrut and Mihai Anitescu. "Large-Scale Parallel Multibody Dynamics with Frictional Contact on the Graphical Processing Unit ". Preprint ANL/MCS-P1494-0508]*

- Times reported are in seconds for one second long simulation
- GPU: NVIDIA GeForce 8800 GTX

Bricks	Sequential Version	GPU Co-processing Version
1000	43	6
2000	87	10
8000	319	42

Future work

- N non symmetric, but positive semidefinite.
- Parallelizing the algorithms: block Jacobi with Gauss Seidel blocks.
- Asynchronous version of the algorithm, particularly for use with GPU.
- Including a good collision model– here we are at a loss with rigid body theory – may need some measure of deformability.
- Compare with experimental data.

Conclusions

- We have defined a new algorithm for complementarity problems with conic constraints.
- We have shown that it can solve very large problems in granular flow far faster than DEM.
- It is the first iterative algorithm that provably converges for nonsmooth rigid body dynamics.
- Its scalability is decent.
- We have created a multithreaded implementation and GPU port increases computational speed by a factor of 7-8.

References (preprints are at authors' web site)

- M Anitescu, A. Tasora. ”An iterative approach for cone complementarity problems for nonsmooth dynamics”. Preprint ANL/MCS-P1413-0507, May 2007.
- M. Anitescu. Optimization-based simulation of nonsmooth dynamics. Mathematical Programming, series A, 105, pp 113–143, 2006.
- Madsen, J., Pechdimaljian, N., and Negrut, D., 2007. Penalty versus complementarity-based frictional contact of rigid bodies: A CPU time comparison. Preprint. TR-2007-05, Simulation-Based Engineering Lab, University of Wisconsin, Madison.