An open question : How to solve efficiently 3D frictional contact problem ?

Vincent Acary, Maurice Brémond
INRIA Rhône–Alpes, Grenoble.

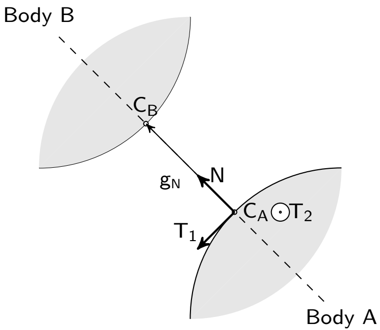
Computational Contact Mechanics: Advances and Frontiers in Modeling Contact,
Banff International Research station,
Canada, February 15–21th, 2013,

Joint work with Florent Cadoux, Claude Lemaréchal, Jérôme Malick,
Florence Bertails–Descoubes, Gilles Daviet

One of the father of Nonsmooth Mechanics and Convex Analysis

Jean Jacques Moreau (1923 – 2014)

Signorini's condition and Coulomb's friction



- gap function $g_N = (C_B - C_A)N$.
- reaction forces

$$r = r_N N + r_T, \quad \text{with } r_N \in \boldsymbol{R} \text{ and } r_T \in \boldsymbol{R}^2.$$

- Signorini condition at position level

$$0 \leqslant g_N \perp r_N \geqslant 0.$$

- relative velocity

$$u = u_N N + u_T, \quad \text{with } u_N \in \boldsymbol{R} \text{ and } u_T \in \boldsymbol{R}^2.$$

- Signorini condition at velocity level

$$\begin{cases} 0 \leqslant u_N \perp r_N \geqslant 0 & \text{if } g_N \leqslant 0 \\ r_N = 0 & \text{otherwise.} \end{cases}$$

Signorini's condition and Coulomb's friction

Modeling assumption

Let μ be the coefficient of friction. Let us define the Coulomb friction cone K which is chosen as the isotropic second order cone

$$K = \{r \in \mathbf{R}^3 \mid \|r_\mathrm{T}\| \leqslant \mu r_n\}. \tag{1}$$

The Coulomb friction states

- for the sticking case that

$$u_\mathrm{T} = 0, \quad r \in K \tag{2}$$

- and for the sliding case that

$$u_\mathrm{T} \neq 0, \quad r \in \partial K, \exists\, \alpha > 0, r_\mathrm{T} = -\alpha u_\mathrm{T}. \tag{3}$$

Disjunctive formulation of the frictional contact behavior

$$\begin{cases} r = 0 & \text{if } g_\mathrm{N} > 0 \quad \text{(no contact)} \\ r = 0, u_\mathrm{N} \geqslant 0 & \text{if } g_\mathrm{N} \leqslant 0 \quad \text{(take–off)} \\ r \in K, u = 0 & \text{if } g_\mathrm{N} \leqslant 0 \quad \text{(sticking)} \\ r \in \partial K, u_\mathrm{N} = 0, \exists\, \alpha > 0, u_\mathrm{T} = -\alpha r_\mathrm{T} & \text{if } g_\mathrm{N} \leqslant 0 \quad \text{(sliding)} \end{cases} \tag{4}$$

Signorini's condition and Coulomb's friction

Second Order Cone Complementarity (SOCCP) formulation [?]

- Modified relative velocity $\hat{u} \in \mathbf{R}^3$ defined by

$$\hat{u} = u + \mu \|u_{\mathrm{T}}\| \mathrm{N}. \tag{5}$$

- Second-Order Cone Complementarity Problem (SOCCP)

$$K^\star \ni \hat{u} \perp r \in K \tag{6}$$

if $g_{\mathrm{N}} \leqslant 0$ and $r = 0$ otherwise. The set $K^\star$ is the dual convex cone to K defined by

$$K^\star = \{u \in \mathbf{R}^3 \mid r^\top u \geqslant 0, \quad \text{for all } r \in K\}. \tag{7}$$

Signorini's condition and Coulomb's friction

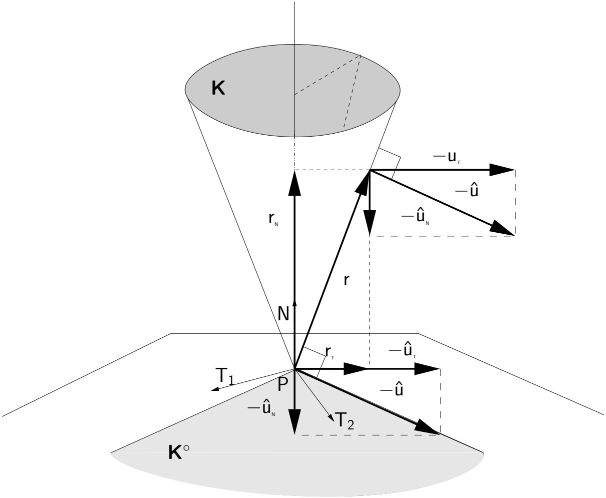


Figure: Coulomb's friction and the modified velocity $\hat{u}$. The sliding case.

3D frictional contact problem

Multiple contact notation

For each contact $\alpha \in \{1, \dots n_c\}$, we have

- the local velocity : $u^\alpha \in \boldsymbol{R}^3$, and

$$u = [[u^\alpha]^\top, \alpha = 1 \dots n_c]^\top$$

- the local reaction vector $r^\alpha \in \boldsymbol{R}^3$

$$r = [[r^\alpha]^\top, \alpha = 1 \dots n_c]^\top$$

- the local Coulomb cone

$$K^\alpha = \{r^\alpha, \|r^\alpha_{\mathrm{T}}\| \leqslant \mu^\alpha |r^\alpha_{\mathrm{N}}|\} \subset \boldsymbol{R}^3$$

  and the set K is the cartesian product of Coulomb's friction cone at each contact, that

$$K = \prod_{\alpha=1\dots n_c} K^\alpha \tag{8}$$

  and $K^\star$ is dual.

3D frictional contact problems

Problem 1 (General discrete frictional contact problem)

Given

- *a symmetric positive definite matrix* $M \in \mathbb{R}^{n\times n}$,
- *a vector* $f \in \mathbb{R}^n$,
- *a matrix* $H \in \mathbb{R}^{n\times m}$,
- *a vector* $w \in \boldsymbol{R}^m$,
- *a vector of coefficients of friction* $\mu \in \boldsymbol{R}^{n_c}$,

find three vectors $v \in \mathbb{R}^n$, $u \in \boldsymbol{R}^m$ *and* $r \in \boldsymbol{R}^m$, *denoted by* $\mathrm{FC/I}(M, H, f, w, \mu)$ *such that*

$$\begin{cases} Mv = Hr + f \\ u = H^\top v + w \\ \hat{u} = u + g(u) \\ K^\star \ni \hat{u} \perp r \in K \end{cases} \tag{9}$$

with $g(u) = [[\mu^\alpha \|u_T^\alpha\| \mathrm{N}^\alpha]^\top, \alpha = 1 \ldots n_c]^\top$. □

3D frictional contact problems

Problem 2 (Reduced discrete frictional contact problem)

Given

- *a symmetric positive semi–definite matrix* $W \in \mathbb{R}^{m \times m}$,
- *a vector* $q \in \mathbb{R}^m$,
- *a vector* $\mu \in \boldsymbol{R}^{n_c}$ *of coefficients of friction,*

find two vectors $u \in \boldsymbol{R}^m$ *and* $r \in \boldsymbol{R}^m$, *denoted by* $\mathrm{FC/II}(W, q, \mu)$ *such that*

$$\begin{cases} u = Wr + q \\ \hat{u} = u + g(u) \\ K^\star \ni \hat{u} \perp r \in K \end{cases} \tag{10}$$

with $g(u) = [[\mu^\alpha \| u_T^\alpha \| \mathrm{N}^\alpha]^\top, \alpha = 1 \ldots n_c]^\top$. □

Relation with the general problem

$W = H^\top M^{-1} H$ and $q = H^\top M^{-1} f + w$.

3D frictional contact problems

Wide range of applications

Origin of the linear relations .

$$Mv = Hr + f, \quad u = H^\top v + w$$

- Time–discretization of the discrete dynamical mechanical system
  - Event–capturing time–stepping schemes
  - Event–detecting time–stepping schemes (event-driven)
- Time–discretization and space discretization of the elasto dynamic problem of solids
- Space discretization of the quasi–static problem of solids.

with a possible linearization (Newton procedure.)

➜ These problems are really representative of a lot of applications.

From the mathematical programming point of view

Nonmonotone and nonsmooth problem

$$K^\star \ni Wr + q + g(Wr + q) \perp r \in K \tag{11}$$

- if we neglect $g(\cdot)$, (11) is a gentle monotone SOCLCP that is the KKT conditions of a convex SOCQP.
- otherwise, the problem is nonmonotone and nonsmooth since $g()$ is nonsmooth

➜ The problem is very hard to solve efficiently.

Possible reformulation

- Variational inequality or normal cone inclusion

$$-(Wr + q + g(Wr + q)) \stackrel{\Delta}{=} -F(r) \in N_K(r). \tag{12}$$

- Nonsmooth equations $G(r) = 0$
  - The natural map F^{nat} associated with the VI (12) $F^{\mathrm{nat}}(z) = z - P_X(z - F(z))$.
  - Variants of this map (Alart-Curnier formulation, ...)
  - one of the SOCCP-functions. (Fisher-Bursmeister function)
- and many other ...

An existence result. (F. Cadoux PhD)

Let us introduce a slack variable

$$s^\alpha := \|u_T^\alpha\|$$

New formulation of the modified velocity with $A \in \boldsymbol{R}^{m \times n_c}$

$$\hat{u} := u + As \qquad (g(u) = As)$$

The problem $\mathrm{FC/I}(M, H, f, w, \mu)$ can be reformulated as

$$\begin{cases} Mv = Hr + f \\ \widetilde{u} = H^\top v + w + As \\ K^\star \ni \hat{u} \perp r \in K \end{cases}$$

An existence result.

The problem (14) appears to be the KKT condition of

primal problem

$$\begin{cases} \min \quad J(v) := \frac{1}{2} v^\top M v + f^\top v \\ H^\top v + w + As \in K^\star \end{cases} \tag{D_s}$$

dual problem

$$\begin{cases} \min \quad J_s(r) := \frac{1}{2} r^\top W r - q_s^\top r \\ r \in K \end{cases} \tag{P_s}$$

with $q_s = q + As$

Interest

Two convex program ➜ existence of solutions under feasibility conditions.

An existence result.

Fixed point problem

Introducing

$$u(s) := \mathrm{argmin}_u(P_s) = \mathrm{argmin}_u(D_s)$$

practically computable by optimization software, and

$$F^\alpha(s) := \|u_T^\alpha(s)\|,$$

the incremental problem becomes a fixed point problem

$$F(s) = s$$

An existence result.

Key assumption

$$\exists v \in \mathbb{R}^m \ : \ Hv + w \in K^\star \tag{13}$$

Using Assumption (13),

- the application $F : \mathbb{R}^n_+ \rightarrow \mathbb{R}^n_+$ is well-defined, continuous and bounded
- apply Brouwer's theorem

Theorem 3

A fixed point exists

This result is a variant of a previous result obtained by [?].

An existence result.

Numerical validation of the assumption

The assumption by solving a linear program over a product of SOC.
Find $x \geqslant 0$

$$\begin{cases} \max x \\ Hv + w - ax \in K^\star \end{cases}$$

where $a = [N^{\alpha,\top}]^\top \in \boldsymbol{R}^m$.

Numerical interest

The fixed point equation $F(s) = s$ can be tackled by

- fixed-point iterations
$$s \leftarrow F(s)$$
- Newton iterations
$$s \leftarrow \mathrm{Jac}[F](s) \backslash F(s)$$
- Variants possible (truncated resolution of inner problem...)

VI based methods

Standard methods

- Basic fixed point iterations with projection

$$z_{k+1} \leftarrow P_X(z_k - \rho_k F(z_k))$$

- Extragradient method

$$z_{k+1} \leftarrow P_X(z_k - \rho_k F(P_X(z_k - \rho_k F(z_k))))$$

- Hyperplane projection method

Self-adaptive procedure for ρ_k

For instance,

$$m_k \in \boldsymbol{N} \quad \text{such that} \quad \begin{array}{l} \rho_k = \rho 2^{m_k}, \\ \rho_k \|F(z_k) - F(\bar{z}_k)\| \leqslant \|z_k - \bar{z}_k\| \end{array} \tag{14}$$

Nonsmooth Equations based methods

Nonsmooth Newton on $G(z) = 0$

$$z_{k+1} = z_k - \Phi^{-1}(z_k)(G(z_k)), \qquad \Phi(z_k) \in \partial G(z_k)$$

- Alart–Curnier Formulation [?]

$$\begin{cases} r_{\mathrm{N}} - P_{\boldsymbol{R}_+^{n_c}}(r_{\mathrm{N}} - \rho_{\mathrm{N}} u_{\mathrm{N}}) = 0, \\ r_{\mathrm{T}} - P_{D(\mu, r_{\mathrm{N},+})}(r_{\mathrm{T}} - \rho_{\mathrm{T}} u_{\mathrm{T}}) = 0, \end{cases} \tag{15}$$

- Direct normal map reformulation

$$r - P_K\left(r - \rho(u + g(u))\right) = 0$$

- Extension of Fischer-Burmeister function to SOCCP

$$\phi_{\mathrm{FB}}(x, y) = x + y - (x^2 + y^2)^{1/2}$$

with Jordan product and square root

Matrix block-splitting and projection based algorithms [?, ?]

Block splitting algorithm with $W^{\alpha\alpha} \in \boldsymbol{R}^3$

$$\begin{cases} u^\alpha_{i+1} - W^{\alpha\alpha} P^\alpha_{i+1} = q^\alpha + \sum\limits_{\beta<\alpha} W^{\alpha\beta} r^\beta_{i+1} + \sum\limits_{\beta>\alpha} W^{\alpha\beta} r^\beta_i \\ \\ \widehat{u}^\alpha_{i+1} = \left[u^\alpha_{\mathrm{N},i+1} + \mu^\alpha \, ||u^\alpha_{\mathrm{T},i+1}||, u^\alpha_{\mathrm{T},i+1} \right]^T \\ \\ \mathbf{K}^{\alpha,*} \ni \widehat{u}^\alpha_{i+1} \perp r^\alpha_{i+1} \in \mathbf{K}^\alpha \end{cases} \tag{16}$$

for all $\alpha \in \{1 \dots m\}$.

One contact point problem

- closed form solutions
- Any solver listed before.

Proximal point technique [?, ?, ?]

Principle

We want to solve

$$\min_x f(x) \tag{17}$$

We define the approximation problem for a given x_k

$$\min_x f(x) + \rho \|x - x_k\|^2 \tag{18}$$

with the optimal point $x^\star$.

$$x^\star \triangleq \mathrm{prox}_{f,\rho}(x_k) \tag{19}$$

Proximal point algorithm

$$x_{k+1} = \mathrm{prox}_{f,\rho_k}(x_k)$$

Special case for solving $G(x) = 0$

$$f(x) = \frac{1}{2} G^\top(x) G(x)$$

Optimization based methods

- Successive approximation with Tresca friction (Haslinger et al.)

$$\begin{cases} \theta = h(r_{\mathrm{N}}) \\ \min \ \frac{1}{2} r^\top W r + r^\top q \\ \quad \text{s.t.} \quad r \in C(\mu,\theta) \end{cases} \tag{20}$$

  where $C(\mu,\theta)$ is the cylinder of radius $\mu\theta$.

- Fixed point on the norm of the tangential velocity [A., Cadoux, Lemaréchal, Malick(2011)] .

$$\begin{cases} s = \|u_{\mathrm{T}}\| \\ \min \ \frac{1}{2} r^\top W r + r^\top (q+\alpha s) \\ \quad \text{s.t.} \quad r \in K \end{cases} \tag{21}$$

  Fixed point or Newton Method on $F(s) = s$

- Alternating optimization problems (Panagiotopoulos et al.)

Siconos/Numerics

Siconos
Open source software for modelling and simulation of nonsmooth systems

Siconos/Numerics
Collection of C routines to solve FC3D problem

- NonSmoothGaussSeidel : VI based projection/splitting algorithm
- TrescaFixedPoint : fixed point algorithm on Tresca fixed point
- LocalAlartCurnier : semi–smooth newton method of Alart–Curnier formulation
- ProximalFixedPoint : proximal point algorithm
- VIFixedPointProjection : VI based fixed-point projection
- VIExtragradient : VI based extra-gradient method
- ...

http://siconos.gforge.inria.fr
use and contribute ...

Performance profiles [?]

- Given a set of problems $\mathcal{P}$
- Given a set of solvers $\mathcal{S}$
- A performance measure for each problem with a solver $t_{p,s}$ (cpu time, flops, ...)
- Compute the performance ratio

$$\tau_{p,s} = \frac{t_{p,s}}{\min\limits_{s \in \mathcal{S}} t_{p,s}} \geqslant 1 \tag{22}$$

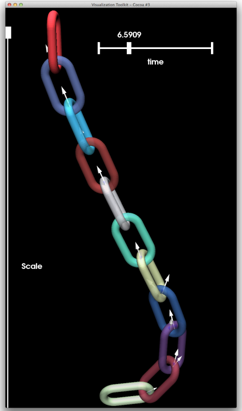
- Compute the performance profile $\rho_s(\tau) : [1, +\infty] \rightarrow [0, 1]$ for each solver $s \in \mathcal{S}$

$$\rho_s(\tau) = \frac{1}{|\mathcal{P}|} \big|\{p \in \mathcal{P} \mid \tau_{p,s} \leqslant \tau\}\big| \tag{23}$$

The value of $\rho_s(1)$ is the probability that the solver s will win over the rest of the solvers.

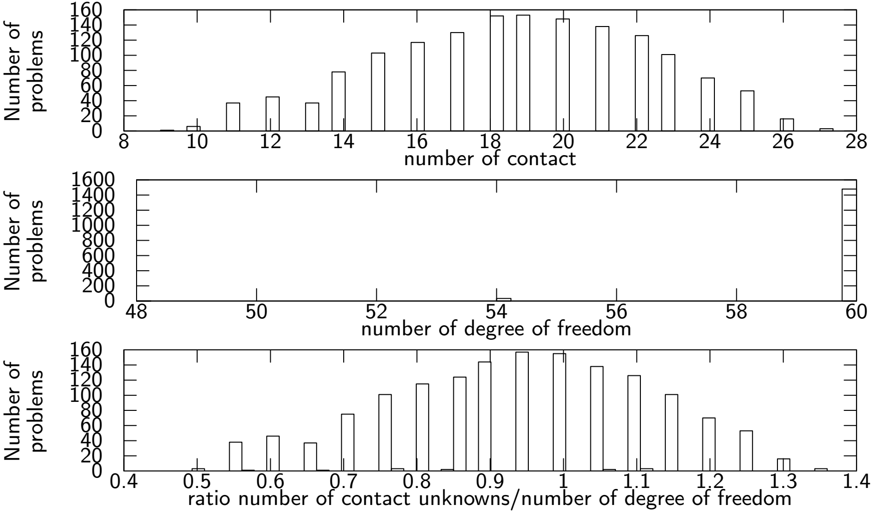
First comparisons. Chain

Hanging chain with initial velocity at the tip



coefficient of friction	0.3
number of problems	1514
number of degrees of freedom	[48 : 60]
number of contacts	[8 :28]
required accuracy	10^{-8}

First comparisons. Chain

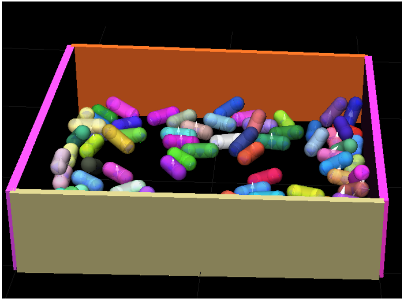


First comparisons. Chain

profile-Chain.pdf

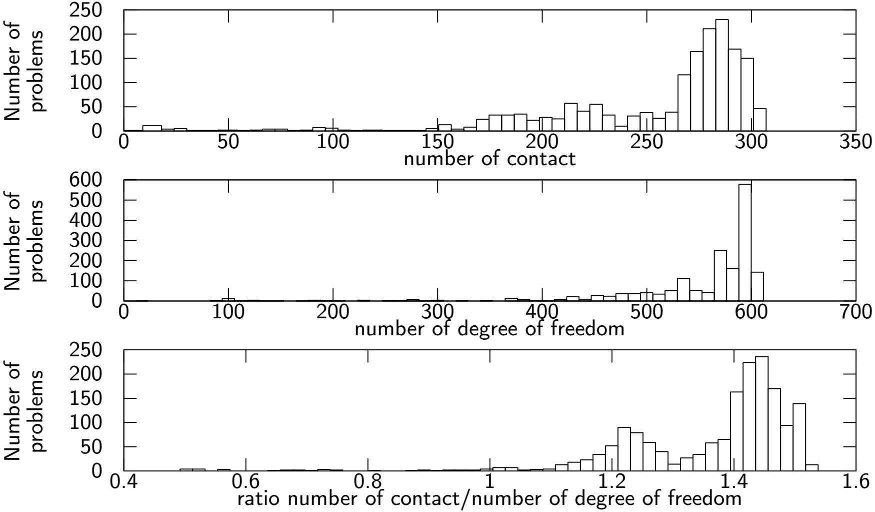
First comparisons. Capsules

100 capsules dropped into a box.



coefficient of friction	0.7
number of problems	1705
number of degrees of freedom	[6 : 600]
number of contacts	[0:300]
required accuracy	10^{-8}

First comparisons. Capsules

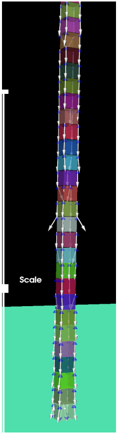


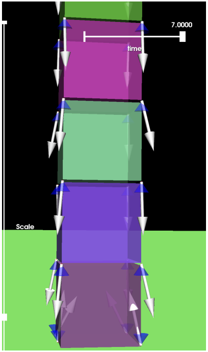
First comparisons. Capsules

profile-Capsules.pdf

First comparisons. BoxesStack

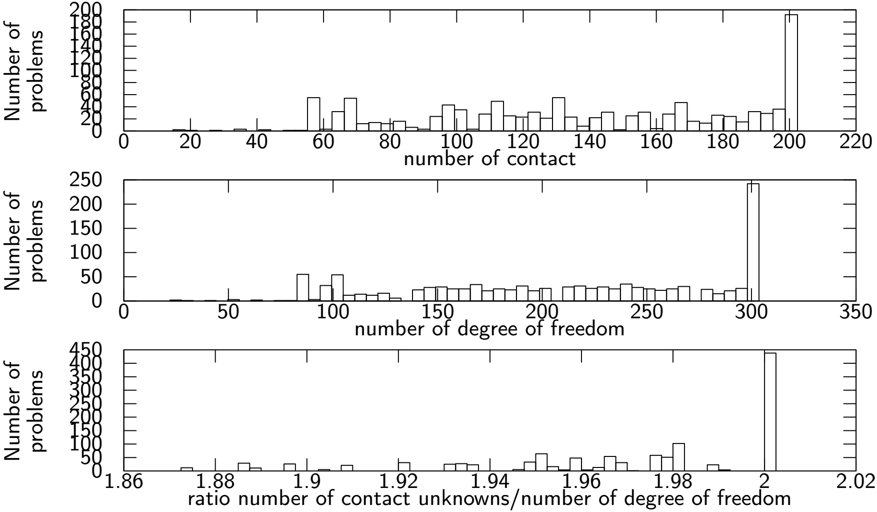
50 boxes stacked under gravity.





coefficient of friction	0.7
number of problems	1159
number of degrees of freedom	[6 : 300]
number of contacts	[0: 200]
required accuracy	10^{-8}

First comparisons. BoxesStack

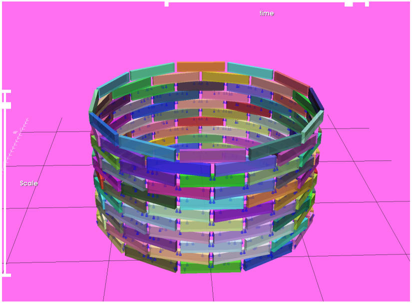


First comparisons. BoxesStack

profile-BoxesStack1.pdf

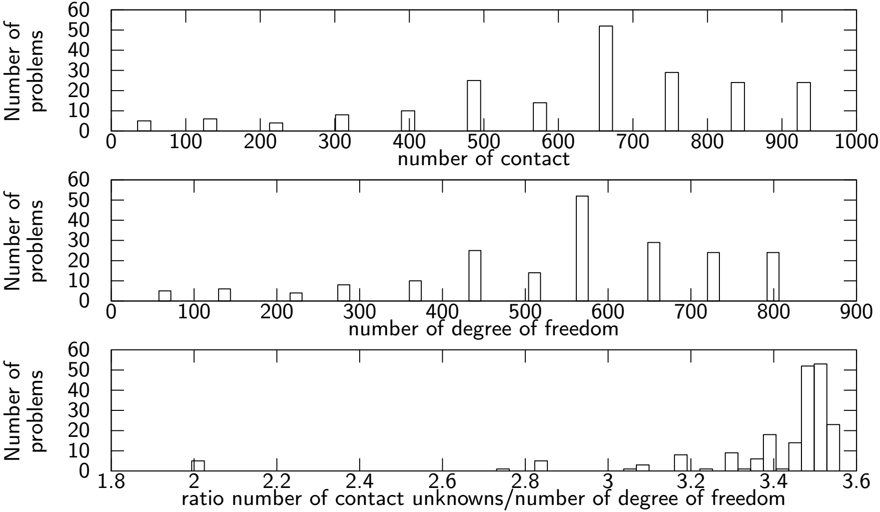
A tower of Kaplas

A Tower of Kaplas



coefficient of friction	0.3
number of problems	201
number of degrees of freedom	[72 : 864]
number of contacts	[0: 950]
required accuracy	10^{-8}

A tower of Kaplas



A tower of Kaplas

profile-KaplasTower.pdf

Conclusions & Perspectives

Conclusions

1. A bunch of articles in the literature
2. No “Swiss–knife” solution : choose efficiency OR robustness
3. Newton–based solver solves efficiently the problems but robustness issues
4. First order iterative methods solves all the problems but very slowly
5. The rank of the H matrix (ratio number of contacts unknows/number of d.o.f) plays an important role.

Perspectives

1. Develop new algorithm and compare other algorithm in the literature. (issues with standard optimization software.)
2. Study the influence of the friction coefficient, the size of problem, the conditionning of the problem , . . .
3. Set up a collection of benchmarks ➜ FCLIB

FCLIB : a collection of discrete 3D Frictional Contact (FC) problems

Our inspiration: MCPLIB or CUTEst

What is FCLIB ?

- A open source collection of Frictional Contact (FC) problems stored in a specific HDF5 format
- A open source light implementation of Input/Output functions in C Language to read and write problems (Python and Matlab coming soon)

Goals of the project

Provide a standard framework for testing available and new algorithms for solving discrete frictional contact problems share common formulations of problems in order to exchange data

Call for contribution

http://fclib.gforge.inria.fr

Thank you for your attention.