An iterative fixed-point method for solving large complementarity problems in multibody systems

Alessandro Tasora
Dipartimento di Ingegneria Industriale, Università degli Studi di Parma
tasora@ied.unipr.it

Parole chiave: Iterative methods, LCP, complementarity, contacts, multibody

SOMMARIO Il presente lavoro, che si colloca nell'alveo delle ricerche su sistemi dinamici con elevato numero di contatti con attrito, intende presentare un originale metodo numerico in grado di risolvere il problema di complementarità lineare attraverso un'iterazione di punto fisso. Tale algoritmo esibisce le proprietà di una mappa contrattiva, fornendo una rapida approssimazione della soluzione al problema LCP anche in presenza di migliaia di vincoli monolateri.

ABSTRACT Aiming at a fast and robust simulation of large multibody systems with contacts and friction, this work presents a novel solution method which can solve large complementarity problems by means of a fixed-point iteration. When thousands of unilateral constraints are added to the system, our method performs like a Banach contractive mapping, providing a fast monotonic approximation to the exact LCP solution.

1. INTRODUCTION

Mechanisms involving contacts and impacts between parts can be modeled in terms of multibody systems with unilateral constraints. The introduction of complementary inequalities in the dynamical model leads to complex LCP (linear complementarity problems), which must be solved at each simulation step [1]. If the simulation entails a large amount of contacts and rigid bodies, such as in the case of part feeders, packaging machines and conveyor belts, the computational burden of classical LCP solvers is unacceptable. In fact most common LCP solution algorithms are based on simplex methods [2], which may exploit a worst-case exponential complexity. Our experience showed that, in spite of deep optimizations [3], simplex methods still couldn't practically handle multibody systems with more than one hundred of colliding bodies.

Fig.1 A complex multibody system with many unilateral and bilateral constraints

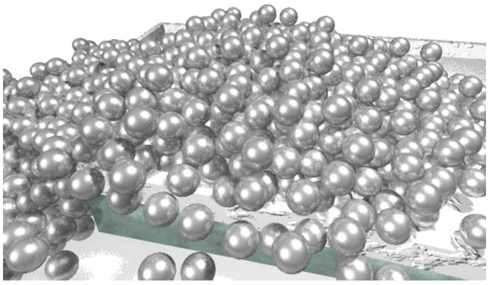

Fig.2 A benchmark: shaker with steel spheres (about 6500 unilateral constraints)

Therefore, we developed a novel solution method, based on a fixed-point iteration, which can solve large complementarity problems with low computational overhead. In case of systems with bilateral constraints only, this method converges to a stationary Gauss-Seidel method with successive over-relaxations. When thousands of unilateral constraints are added to the system, our method performs like a Banach contractive mapping, providing a fast monotonic approximation to the exact LCP solution.

With minimal modifications, our algorithm can handle three-dimensional Coloumb friction; hence, it is able to solve a true NLCP non-linear complementarity problem.

2. THE MULTIBODY MODEL

Dealing with multibody systems with joint constraint only, the dynamical system can be modeled as a DAE differential-algebraic system [4]. However, when unilateral constraints and friction are introduced, the non-smooth nature of the constraints requires the adoption of complex techniques: among these, we endorse the recent approach based on velocity-impulse differential complementarity [5]. That is, inequality constraints are enforced at the velocity level, by means of a linear-complementarity problem (LCP), and a time stepping method is used to solve the differential inclusions for the non-smooth dynamics.

Lets introduce the state vector **q**, the bilateral constraint equations **C**(**q**,t)=**0**, the unilateral constraint equations **D**(**q**)≥**0**, the respective constraint jacobians [$\mathbf{C}_q$] and [$\mathbf{D}_q$], the respective reaction impulses $\boldsymbol{\gamma}_C \in \mathcal{G}_C$ and $\boldsymbol{\gamma}_D \in \mathcal{G}_D$, the stabilization coefficient $0<K<1$, the mass matrix [**M**] and the lagrangian forces $\mathbf{Q}_F$. For sake of simplicity, here we discuss frictionless contacts only.

For the *l-th* integration step *h*, we write the following differential-complementarity problem [3]:

$$\begin{cases} [\mathbf{M}](\dot{\mathbf{q}}^{l+1} - \dot{\mathbf{q}}^{l}) - [\mathbf{C}_q]^T \boldsymbol{\gamma}_C - [\mathbf{D}_q]^T \boldsymbol{\gamma}_D = h\mathbf{Q}_F \\ [\mathbf{C}_q]\dot{\mathbf{q}}^{l+1} + \dfrac{K}{h}\mathbf{C}(\mathbf{q},t) + \dfrac{\partial \mathbf{C}}{\partial t} = \mathbf{0} \\ [\mathbf{D}_q]\dot{\mathbf{q}}^{l+1} + \dfrac{K}{h}\mathbf{D}(\mathbf{q}) \geq \mathbf{0} \quad \perp \quad \boldsymbol{\gamma}_D \geq \mathbf{0} \\ \mathbf{q}^{l+1} = \mathbf{q}^{l} + h\dot{\mathbf{q}}^{l+1} \end{cases} \tag{1}$$

The MLCP mixed-linear complementarity problem embedded in the first three rows of (1) can be expressed in a more useful form, introducing $-\frac{K}{h}\mathbf{C}(\mathbf{q},t) - \frac{\partial \mathbf{C}}{\partial t} = \mathbf{c}$, $-\frac{K}{h}\mathbf{D}(\mathbf{q}) = \mathbf{d}$, $\mathbf{e} = \{\mathbf{c}^T \mid \mathbf{d}^T\}^T$, $[\mathbf{E}_q]^T = [[\mathbf{C}_q]^T \mid [\mathbf{D}_q]^T]$, $\boldsymbol{\gamma}_E = \{\boldsymbol{\gamma}_C^T \mid \boldsymbol{\gamma}_D^T\}^T$, and $[\mathbf{M}]\dot{\mathbf{q}}^l + h\mathbf{Q}_F = \mathbf{f}$:

$$\begin{cases} [\mathbf{N}]\begin{Bmatrix} \boldsymbol{\gamma}_C \\ \boldsymbol{\gamma}_D \end{Bmatrix} + \mathbf{r} = \begin{Bmatrix} \mathbf{0} \\ \boldsymbol{\alpha} \end{Bmatrix} \\ \boldsymbol{\alpha} \geq \mathbf{0} \quad \perp \boldsymbol{\gamma}_D \geq \mathbf{0} \end{cases} \tag{2a}$$

where we used $[\mathbf{N}] = [\mathbf{E}_q][\mathbf{M}]^{-1}[\mathbf{E}_q]^T$ and $\mathbf{r} = [\mathbf{E}_q][\mathbf{M}]^{-1}\mathbf{f} - \mathbf{e}$.

After solving for $\boldsymbol{\gamma}_E$, one can quickly compute

$$\dot{\mathbf{q}}^{l+1} = [\mathbf{M}]^{-1}\mathbf{f} + [\mathbf{M}]^{-1}[\mathbf{E}_q]^T \boldsymbol{\gamma}_E . \tag{2b}$$

3. THE ITERATIVE METHOD

The size of [**N**] in (2a) depends on the number of constraints. For more than one hundred constraints, this MLCP cannot be solved with simplex methods in reasonable time. Therefore we developed a fast *O*(*n*) iterative scheme which does not even need to store [**N**].

We define the following iteration:

$$\boldsymbol{\gamma}_E^{r+1} = f_+\left(\boldsymbol{\gamma}_E^r - [\mathbf{B}^r]\left([\mathbf{N}]\boldsymbol{\gamma}_E^r + \mathbf{r}\right)\right) \tag{3}$$

where we introduced [$\mathbf{B}^r$], a diagonal matrix such that the spectral radius of [$\mathbf{B}^r$] [$\mathbf{N}$] is $0<\lambda<1$, and a projection mapping f_+ defined as follows:

$$f_+(\gamma_i) \begin{cases} = \gamma_i & \text{if } (\gamma_i \in \mathcal{G}_D \wedge \gamma_i > 0) \vee \gamma_i \in \mathcal{G}_C \\ = 0 & \text{if } (\gamma_i \in \mathcal{G}_D \wedge \gamma_i \leq 0) \end{cases} \tag{4}$$

THEOREM 2.1 *The iterative process in (3) is a contraction mapping, and converges to the following unique fixed point:*

$$\boldsymbol{\gamma}_E = f_+\left(\boldsymbol{\gamma}_E - [\mathbf{B}]\left([\mathbf{N}]\boldsymbol{\gamma}_E + \mathbf{r}\right)\right). \tag{5}$$

Proof. Say $T(\boldsymbol{\gamma}_E)$ is the mapping in (3), being $\boldsymbol{\gamma} \in \mathcal{G}$ with $(\mathcal{G}, d)$ complete metric space, then:

$$\left\| T(\boldsymbol{\gamma}_x) - T(\boldsymbol{\gamma}_y) \right\| = \left\| f_+\left(\boldsymbol{\gamma}_x - [\mathbf{B}]\left([\mathbf{N}]\boldsymbol{\gamma}_x + \mathbf{r}\right)\right) - f_+\left(\boldsymbol{\gamma}_y - [\mathbf{B}]\left([\mathbf{N}]\boldsymbol{\gamma}_y + \mathbf{r}\right)\right) \right\|.$$

Also, it is easy to prove, remembering the definition of f_+, that the following inequality holds:

$$\left\| T(\boldsymbol{\gamma}_x) - T(\boldsymbol{\gamma}_y) \right\| \leq \left\| \left(\boldsymbol{\gamma}_x - [\mathbf{B}]\left([\mathbf{N}]\boldsymbol{\gamma}_x + \mathbf{r}\right)\right) - \left(\boldsymbol{\gamma}_y - [\mathbf{B}]\left([\mathbf{N}]\boldsymbol{\gamma}_y + \mathbf{r}\right)\right) \right\|.$$

Rearranging these terms, one gets $\left\| T(\boldsymbol{\gamma}_x) - T(\boldsymbol{\gamma}_y) \right\| \leq \left\| \left([\mathbf{I}] - [\mathbf{B}][\mathbf{N}]\right)\left(\boldsymbol{\gamma}_x - \boldsymbol{\gamma}_y\right) \right\|$.

By properties of norms, $\|ab\| \leq \|a\| \, \|b\|$, then:

$$\left\| T(\boldsymbol{\gamma}_x) - T(\boldsymbol{\gamma}_y) \right\| \leq \left\| \left([\mathbf{I}] - [\mathbf{B}][\mathbf{N}]\right) \right\| \cdot \left\| \boldsymbol{\gamma}_x - \boldsymbol{\gamma}_y \right\|.$$

Recalling that, by hypothesis, the spectral radius of [$\mathbf{B}$][$\mathbf{N}$] is $0<\lambda<1$, we can write:

$$\left\| T(\boldsymbol{\gamma}_x) - T(\boldsymbol{\gamma}_y) \right\| \leq (1-\lambda) \left\| \boldsymbol{\gamma}_x - \boldsymbol{\gamma}_y \right\|.$$

Since $0<(1-\lambda)<1$, by Banach contraction mapping theorem we have that (3) converges monotonically to an unique fixed point. QED.

THEOREM 2.2 *Impulses* $\boldsymbol{\gamma}_E$ *are solutions of the MLCP (2a) if and only if* $\boldsymbol{\gamma}_E$ *satisfies (5).*

Proof. For bilateral constraints, this is straightforward. For unilaterals, this can be easily demonstrated by considering that the outcome of (5) is verified for both the two cases of LCP solutions $\gamma_i =0, \alpha_i>0$ and $\gamma_i >0, \alpha_i=0$, being $\alpha_i = [N_{j\bullet}]\gamma_i + r_i$. The converse is left to the reader.

4. IMPLEMENTATION

Special care has been paid in exploiting the sparsity of jacobians and mass matrix: in fact the method does not require additional matrix elements, so it can run in $O(n)$ space and $O(n)$ time.

Instead of explicitly computing $[\mathbf{N}]=[\mathbf{E}_q][\mathbf{M}]^{-1}[\mathbf{E}_q]^T$ in (3), we compute products between *i-th* rows of jacobians [$\mathbf{E}_{i,}\bullet$] and mass matrix (whose inverse $[\mathbf{M}]^{-1}$ is immediate, if diagonal as usual).

As matrix [$\mathbf{B}$] we use the inverse of [$\mathbf{N}$] diagonal (being [$\mathbf{N}$] definite positive, it can be demonstrated that the method will always converge). This choice of [$\mathbf{B}$], along with improvements such as the immediate update of the n unknowns and the introduction of a speed parameter ω, leads to the final scheme which is similar to a 'projected' Gauss-Seidel or SOR method:

$$\gamma_i^{r+1} = f_+\left(\gamma_i^r - \frac{\omega}{[\mathbf{E}_{i\bullet}][\mathbf{M}]^{-1}[\mathbf{E}_{i\bullet}]^T}[\mathbf{E}_{i\bullet}][\mathbf{M}]^{-1}\left(\sum_{t=1}^{i-1}[\mathbf{E}_{t\bullet}]^T\gamma_t^{r+1} + \sum_{t=i}^{n}[\mathbf{E}_{t\bullet}]^T\gamma_t^r + \mathbf{f}\right) - e_i\right) \tag{6}$$

It is interesting to note that the amount of floating point operation can be minimized in many ways. For example, not all matrix multiplications must be repeated at each *r-th* iteration step. Also, remember that $[\mathbf{M}]^{-1}\left(\sum_t [\mathbf{E}_{t\bullet}]^T \gamma_t + \mathbf{f}\right) = \dot{\mathbf{q}}$ .The final, optimized algorithm is:

Algorithm 4.1

```
for all n constraints, pre-compute i-th values $g_i = [\mathbf{E}_{i\bullet}][\mathbf{M}]^{-1}[\mathbf{E}_{i\bullet}]^T$ , $\mathbf{s}_i = [\mathbf{M}]^{-1}[\mathbf{E}_{i\bullet}]^T$
initialize $\boldsymbol{\gamma}^0 = \mathbf{0}$ (or warm-start with a guess vector $\boldsymbol{\gamma}^0 = \boldsymbol{\gamma}^*$ )
initialize $\dot{\mathbf{q}}^0 = \sum_{i=1}^{n} \mathbf{s}_i^T \boldsymbol{\lambda}_t^0 + [\mathbf{M}]^{-1}\mathbf{f}$   (using sparse matrix algebra)
for all iterations  $r = 0..r_{max}$
    for all constraints  $i = 1..n$
        $\gamma_i^{r+1} = f_+\left(\gamma_i^r - \frac{\omega}{g_i}[\mathbf{E}_{i\bullet}]\dot{\mathbf{q}}^r - e_i\right)$
        $\Delta\gamma_i^{r+1} = \gamma_i^{r+1} - \gamma_i^r$
        $\dot{\mathbf{q}}^{r+1} = \dot{\mathbf{q}}^r + \mathbf{s}_i^T \boldsymbol{\lambda}_t^{r+1}$
```

Iterations, usually stopped when an approximation threshold has been reached, can be also prematurely aborted at r_{max} if the simulation must meet hard-real-time requirements.

Since the algorithm belongs to the *NC* complexity class, it can be implemented also on parallel processors such as in the pipelines of modern GPUs or in synergic processors (like the Cell™).

5. CONCLUSIONS

Aiming at a linear-time solution of dynamical systems with thousands of constraints and contacts, a novel LCP solution method has been presented, implemented and tested.

We implemented this method into the *HyperOctant* library of our *Chrono::Engine* multibody project [6]. Benchmarks proved that our iterative approach is orders of magnitude faster than simplex methods, and we were able to handle large simulations with thousands of colliding rigid bodies (see Fig.1 and 2).

Bibliography

[1] Anitescu, M., Potra, F.A., Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics*, 14, 1997, 231–247

[2] Baraff, D., Fast contact force computation for nonpenetrating rigid bodies. *Computer Graphics (Proc. SIGGRAPH)*, 28, 1994, 23–34

[3] Tasora, A., Manconi, E., Silvestri, M.: Un nuovo metodo del simplesso per il problema di complementarità lineare mista in sistemi multibody con vincoli unilateri, *Proceedings of AIMETA'05*, Firenze, 2005.

[4] Shabana, A., *Multibody systems*, John Wiley & Sons, New York 1989.

[5] Stewart, D. E., Trinkle, J. C., An implicit time-stepping scheme for rigid-body dynamics with inelastic collisions and Coulomb friction , *Int.J. Numerical Methods in Eng.* 39, 673-2691, 1996.

[6] Tasora, A., Chrono::Engine web page, at: *http://www.deltaknowledge.com/chronoengine*