Comparison and Coupling of Algorithms for Collisions, contact and friction in rigid multi-body simulations.

Mathieu Renouf, Vincent Acary

III European Conference on Computational Mechanics
Solids, Structures and Coupled Problems in Engineering
C.A. Mota Soares et.al. (eds.)
Lisbon, Portugal, 5–8 June 2006

COMPARISON AND COUPLING OF ALGORITHMS FOR COLLISIONS, CONTACT AND FRICTION IN RIGID MULTI-BODY SIMULATIONS

Mathieu Renouf[1] **and Vincent Acary**[2]

[1]LaMCoS - TMI team
CNRS/INSA de Lyon - UMR5514 18-20, rue des sciences F69621 VILLEURBANNE cedex - France
e-mail: Mathieu.Renouf@insa-lyon.fr
[2]Bipop Project, INRIA Rhone–Alpes
ZIRST Montbonnot, 655, avenue de l'Europe, 38664 Saint ISMIER, France
e-mail: Vincent.Acary@inrialpes.fr

Keywords: Frictional Contact, Block structure, direct method, iterative method

Abstract. *Numerous works in computational mechanics are dedicated to multi-body systems. This leads to the use of various methods to simulate the static or dynamic evolution of complex systems. The case of dense multi-contact assemblies is one of the more complex one: the problem have often a large number of unknown and have a infinity of solution due to the definition of the matrix of the system. Moreover this problem become harder when friction or more complex laws are introduced in the system. Thus we need fast and robust solvers to perform mechanical studies. These performances can be increased when the special problem structure is considered (sparse matrices, block structured problem).*

Our work is based on the Non Smooth Contact Dynamic framework introduced by Moreau. The method uses a time-stepping integrator without explicit event-handling procedure and an unilateral contact impact formulation associated to Coulomb's friction. In this paper we use and compare different iterative algorithms such as Gauss-Seidel, projected conjugate gradient and direct ones as Lemke and Quadratic programming solvers. The efficiency of the methods is compared in terms of complexity, convergence criterion and of CPU time.

To illustrate the results, we focus on granular assemblies. 3D frictional contact simulations are performed with ConF&TiS and the Numerics library of the siconos project.

1 INTRODUCTION

Numerous works in computational mechanics are dedicated to the simulation of multi-body systems [1] with application fields in soil mechanics [2], civil engineering [3] and also in computer graphics [4]. This multiplicity of applications and specific constraints lead to use a large panel of methods to simulate the static or dynamic evolution of complex systems. The case of dense multi-contact assemblies is one of the more complex one. The problem have often a large number of unknown and have a infinity of solutions due to the definition of the matrix of the problem. If for frictionless system, theoritical result can be exhibit when friction is take into account the problem become harder and leads some authors to use friction approximation [5].

Granular material are a specific case of large dense multi-body system. To the mathematical complexity we must add the physical one due to a behaviour which can be liquid [6] or solid [2]. When local contact impulses are used as primal unknown, the matrix issue from the formulation of the frictional contact problem presents a block structure (with or without approximation of the friction cone). In the general case a standard block splitting method is used [7], call also Non Smooth Block Gauss-Seidel algorithm. The numerical efficiency is critical (robust but slowly converence) justifying studies and improvements [8]. More complex algorithm as Projected Conjugate Gradient one [9] are faster for two-dimensional problem but does not allow in the case of three-dimensional frictional contact problem to obtain a large CPU time gain. Direct method such as Lemke method can also be used but need special pivot research [10]. Quadratic programming solver are also an alternative but present result similar than Lemke algorithm. In the multi-contact case, pivot method appear to be slower than iterative ones, especially for a large number of unknown [4]. But for small system, direct methods are often more efficient and faster than iterative ones for some kind of problem. Nevertheless for three-dimensional frictional contact problem approximation of the friction cone must be used.

The aim of this paper is to present a combination of algorithms according to the block structure of three-dimensional frictional contact problem rigid multi-body systeme. We focus on the case of granular material simulations. This combination is performed using different algorithms (direct or iterative) with a global standard block splitting method. After a presentation of the framework used for our three-dimensional frictional contact problem in section 2, we present in details the block structure formulation in section 3 for formulation based or not based on an approximation of the friction cone. The section 4 gives an overview of the different algorithms. Results are presented in section 5 and section 6 concludes the paper.

2 FORMULATION OF THE FRICTIONAL CONTACT PROBLEMS

In computational mechanics, when we look for approaches well suited for the rigid body dynamics with contact, friction and impact, we found the duality between compliant/unilateral model and event-driven/time-stepping scheme. In the context of granular materials, where large collections of bodies are encountered, Cundall [11] was the first to propose a numerical tools based on a Euler scheme and where contacts are governed by a compliant model. With a definitely different approach, Moreau [12] and Jean [7] present a treatment of rigid bodies with unilateral contact, Coulomb's dry friction and impact in the framework of the non-smooth mechanics and convex analysis. This framework yields a time-stepping scheme (without explicit event-handling) where velocity and impulses are the primary variables. Still in a non-smooth framework, Pfeiffer and Glocker [1] or Stewart and Trinkle [13] designed an event-driven algorithm as time integration scheme and proposed a general formulation of the dynamics at the acceleration-force level. The resolution is performed using direct method [5], quadratic pro-

graming solver [14] or MCP solver [15]. But this kind of integration scheme are out of the scope of this paper a well as Mixed Complementary Problem. The approach used to simulate our multi-body system is based on the *Non Smooth Contact Dynamic* (NSCD) framework evoked previously. The scheme is a very efficient numerical tool on a great number of applications that are well-known for theirs difficulties [3, 2]. We present here the headlines of the method and refer to [12, 7] for detailed explanations.

2.1 Discretization of equation of motion

Formalism of the method lays on a special formulation of the equation of motion. When we deal with multi-contact systems, shocks must be expected during the evolution of the system. This shocks lead to velocity discontinuities and cannot allow to define the acceleration as the usual second time derivative of the configuration parameter (denote $\mathbf{q}$). Consequently equation of motion must be formulated in terms of a measure differential equation,

$$\mathbb{M}d\dot{\mathbf{q}} = \mathbf{F}(t, \mathbf{q}, \dot{\mathbf{q}})dt + d\mathbf{R}, \tag{1}$$

where dt is the Lebesgue measure on $\mathbb{R}$, $d\dot{\mathbf{q}}$ is a differential measure representing the acceleration measure and $d\mathbb{R}$ is a non-negative real measure. In the equation (1), $\mathbb{M}$ represents the mass matrix and $\mathbf{F}(t, \mathbf{q}, \dot{\mathbf{q}})$ the external forces (internal forces vanish in rigid body systems).

The contact problem is solved over the time discretization interval $]t_i, t_{i+1}]$ in the sense defined previously. In this way, the equation (1) is integrated on each time interval and approximated using a θ method, an implicit first-order scheme, using the configuration parameter and its first derivative [7]. Its stability condition implies that θ must remain between $1/2$ and 1. Successive approximations of equation (1) lead to the following system

$$\left\{ \begin{array}{l} \dot{\mathbf{q}}_{i+1} = \dot{\mathbf{q}}_i^{free} + (\mathbb{M}^{-1})\mathbf{R}_{i+1} \\ \mathbf{q}_{i+1} = \mathbf{q}_i + h\theta\dot{\mathbf{q}}_{i+1} + h(1-\theta)\dot{\mathbf{q}}_i \end{array} \right. \tag{2}$$

with

$$\dot{\mathbf{q}}_i^{free} = \dot{\mathbf{q}}_i + \mathbb{M}^{-1}h(\theta\mathbf{F}_{i+1} + (1-\theta)\mathbf{F}_i).$$

where $\dot{\mathbf{q}}^{free}$ denotes the free velocity (velocity computed without contact forces). Index i (resp. $i+1$) refers to time t_i (resp. t_{i+1}). Global quantities such as the sum of contact forces $\mathbf{R}$ and the body velocity $\dot{\mathbf{q}}$ are related to local variables via two linear mappings denote $\mathbb{H}$ and its transpose $\mathbb{H}^*$. Thus the local forces vector $\mathbf{r}$, expressed in the local frame, are related to $\mathbf{R}$ by the relation

$$\mathbf{R} = \mathbb{H}(\mathbf{q}), \mathbf{r}, \tag{3}$$

where α is the index over the contact set. In the same way, the velocity of the bodies $\dot{\mathbf{q}}$ is related to the contact relative velocities vector $\mathbf{v}$ by the relation,

$$\mathbf{v} = \mathbb{H}^*(\mathbf{q}), \dot{\mathbf{q}}, \tag{4}$$

Using the equations (3) and (4), the discretization of the equation of motion and the contact law can be summarised in the following system:

$$\left\{ \begin{array}{l} \mathbb{W}\mathbf{r}_{i+1} - \mathbf{v}_{i+1} = -\mathbf{v}_{free} \\ law_\alpha[\mathbf{v}_{\alpha,i+1}, \mathbf{r}_{\alpha,i+1}] = .true., \quad \alpha = 1, \ldots, n_c \end{array} \right. \tag{5}$$

where $\mathbb{W}$ $(= \mathbb{H}^*\mathbb{M}^{-1}\mathbb{H})$ is the Delassus operator. The right-hand-side of the first equation in (5) represents the free relative velocity similar to $\dot{\mathbf{q}}^{free}$. The second equation in (5) denotes

the contact law which must be satisfied by each component of the couple $(\mathbf{v}, \mathbf{r})_\alpha$. The contact law used in the context of this paper is a classical Signorini condition coupled with Coulomb's friction. An extension of the Newton impact law to multi-contact assemblies [12] is coupled to the friction law.

3 DELASSUS OPERATOR FORMULATION

With the NSCD framework, the Delassus operator presents a block structure. When we related this theoritical structure to the physical structure of granular material, each block on the diagonal of the operator contains the local information of each contact, and the blocks out the diagonal represent the connectivity between contact. We present in this section the explicit formulation of the Delassus operator used in the problem (5). This explicit formulation is dressed for a one contact case in a non approximated problem and an approximated one.

3.1 Without approximation

The linear mapping $\mathbb{H}$ introduced in section 2 can be decomposed in a normal part $\mathbb{H}_N$ and a tangential one $\mathbb{H}_T$ and expressed as $\mathbb{H} = \left[\begin{array}{cc} \mathbb{H}_T & \mathbb{H}_N \end{array} \right]$. Using the definition of $\mathbb{W}$ we obtain

$$\mathbb{W} = \left[\begin{array}{cc} \mathbb{W}_{TT} & \mathbb{W}_{TN} \\ \mathbb{W}_{NT} & W_{NN} \end{array} \right] = \left[\begin{array}{cc} \mathbb{H}_T^* \mathbb{M}^{-1} \mathbb{H}_T & \mathbb{H}_T^* \mathbb{M}^{-1} \mathbb{H}_N \\ \mathbb{H}_n^* \mathbb{M}^{-1} \mathbb{H}_T & \mathbb{H}_N^* \mathbb{M}^{-1} \mathbb{H}_N \end{array} \right]$$

As $\mathbb{M}$ is diagonal (in the rigid body context), the matrix is easily invertible. Here the contact index is omitted to keep a pleasant reading.

3.2 Delassus operator for an inner approximation

3.2.1 Definition

To use direct resolution methods such as Lemke or Quadratic Programming solvers [16, 17] to solve the frictional contact problem, an approximation of the friction cone must be used. For this kind of approximation, contrary to the first one, some existence of solutions can be exhibit [13].

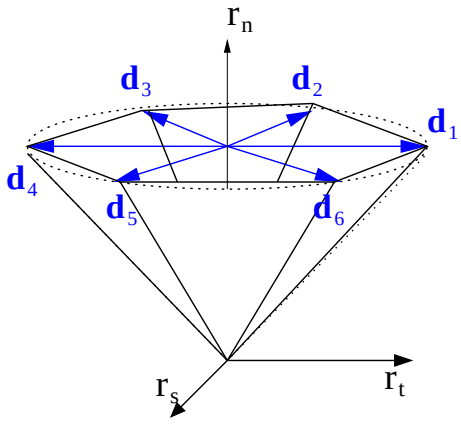

Figure 1: Inner approximation

Different kind of approximations can be performed : regular or not, inner or outer. We will describe here the global form of the matrix using an regular inner approximation as represented on figure 1. First we define the approximated friction cone as

$$\widehat{FC} = \{r_N \mathbf{n} + \mathbb{D}\boldsymbol{\beta} \,|\, r_N \geq 0,\, \beta \geq \mathbf{0},\, \mathbf{e}^* \boldsymbol{\beta} \leq \mu r_N\}$$

with

$$\mathbb{D} = \left[\begin{array}{c|c|c} \mathbf{d}_1 & ... & \mathbf{d}_\nu \end{array} \right].$$

Each vector $\mathbf{d}_i, \forall i \in \{1, ..., \nu\}$ (see Fig.1) and the dimension of vector β are related to the approximation by the ν-gon.

3.2.2 One contact case

Using the previous definition of the approximation of the friction cone and the definition of $\mathbb{W}$, we can build the matrix $\tilde{\mathbb{W}}$ and the right hand side $\tilde{\mathbf{v}}_{free}$ of the approximated frictional contact as follow

$$\tilde{\mathbb{W}} = \left[\begin{array}{ccc} \mathbb{D}^* \mathbb{W}_{TT} \mathbb{D} & \mathbb{D}^* \mathbb{W}_{TN} & \mathbf{e} \\ \mathbb{W}_{NT} \mathbb{D} & W_{NN} & 0 \\ -\mathbf{e}^* & \mu & 0 \end{array} \right] \quad \text{and} \quad \tilde{\mathbf{v}}_{free} = \left[\begin{array}{c} \mathbb{D}^* \mathbf{v}_{free,T} \\ v_{free,N} \\ 0 \end{array} \right].$$

Note that this kind of approximation leads to increase the dimension of the problem and generate some problem with strong impact in special applications such as haptic control [4].

3.3 Multi-contact strategy

When multi-contact assemblies are considered, the size of $\mathbb{W}$ can increase. In the context of granular material, the case of this paper, numerical optimisation have been performed to preserve the CPU time during computation [18]. When we use approximation of the friction cone, the size increase with the accuracy of the approximation. Moreover with pivot method the matrix must be considered in its all dimension .
The idea here is to keep the structure found in [7] and extend it to combination with LCP or QP solver. Thus when we considered an approximate problem, the approximation is performed at the local level on the local Delassus operator and not on its all structure. Then we considered only small problem where pivot or direct can be efficient and we can take advantage of this kind of method. Moreover the construction of each $\tilde{\mathbb{W}}$ can be performed one times only to preserve CPU time. The only the right hand side of problem (5) must be updated to take into account the evolution of the system. This method is different from the theoretical results and the methodology presented in [19].

4 ALGORITHM PANEL

4.1 Block Splitting method

To solve problem (5) we use a general block splitting method on matrix $\mathbb{W}$ or $\tilde{\mathbb{W}}$. The idea is to solve the local frictional contact problem using an algorithm such as Newton algorithm [20], Lemke [10] and to use a block splitting method to solve contact by contact each local problem. Method such as CPG algorithm [9] or NCP solver [21] will be not considered here.
Using the notation of system (5), the global splitting scheme is equal to

$$\begin{array}{ll} \mathbf{v}_\alpha^{k+1} - \mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^{k+1} & = \mathbf{v}_{\alpha,free} + \sum_{\beta<\alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\beta^{k+1} + \sum_{\beta>\alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\beta^{k} \\ & = \mathbf{b}^k \end{array} \tag{6}$$

where the index k refers to the splitting method iterations. The time index is omitted to make pleasant reading. Using the global scheme to propagate the information, we must now define what kind of algorithm used to the local problem.

4.2 Lexicographic Lemke

Pivoting methods are often used to solve LCP($\mathbf{q}$,$\mathbb{W}$) [], but the good results obtain with such method depend strongly of the properties of the matrix $\mathbb{W}$. In multi-contact assemblies (of rigid bodies), the matrix $\mathbb{W}$ is almost always a positive semi-definite (PSD) matrix and so leads to degenerate problem. In this case classical pivoting methods cannot be used and we must have resorts to some adaptation in pivoting algorithms to avoid cycling during the pivoting process. To face this problem with the Lemke method, a solution is to base the choice of the pivot and the minimum ratio on a lexicographic ordering [10, 22]. Thus the choice of the pivot variable is unique and allows to obtain a solution when the problem is degenerated.

4.3 Quadratic Programming Solver

The interests of Quadratic Programming Solver leads on a trivial reformulation of the contact problem as an optimisation problem. Moreover, minimisation algorithm used ensure a strong stability and are reliable for semi-definite positive matrices. For frictional simulation, Non Symmetric Quadratic Programming solver must be used. The formulation of the frictional contact problem leads to a non symmetric matrix. To face this problem we use an QP solver based on the generalised Fletcher's method [23]. These improvements ensure an algorithm termination extending to the case of round-off error

4.4 Dedicated Local Newton Method

The non smooth frictional contact problem can also be solved by a Newton method [20]. Resolution of system (5) can be reduce to looking for the zero of a function $\mathcal{F}(\mathbf{v},\mathbf{r})$ define as

$$\mathcal{F}(\mathbf{v},\mathbf{r}) = \mathbf{0} \Leftrightarrow \left\{ \begin{array}{l} \mathbf{v} - \mathbf{v}_{free} - \mathbb{W}\mathbf{r} = \mathbf{0}, \\ \mathbf{r} - \gamma_n^+\mathbf{n} - proj(\gamma_t; \mathcal{C}(\gamma_n^+))\mathbf{t} = \mathbf{0}, \end{array} \right. \tag{7}$$

with $\gamma = \mathbf{r} - \rho\mathbf{u} = \gamma_n\mathbf{n} + \gamma_t$ et $\gamma_n^+ = proj(\gamma_n; \mathbb{R}^+)$ et $\gamma_t \in \mathbb{R}^2$. Problem (7) is equivalent to problem presented in [20] with partial properties of convergence. The principle of the Newton's method is to determine the couple $\mathbf{X} = (\tilde{\mathbf{v}}, \tilde{\mathbf{r}})$ which satisfied $\mathcal{F}(\mathbf{X}) = \mathbf{0}$. For that, we define the different iterate $(\mathbf{X}^p)_p$ as

$$\partial\mathcal{F}(\mathbf{X}^p)\Delta\mathbf{X} = -\mathcal{F}(\mathbf{X}^p), \tag{8}$$

where $\partial\mathcal{F}(\mathbf{X}^p)$ is one of the Jacobian matrices, element of the Jacobian base of $\mathcal{F}(\mathbf{X}^p)$. Then we introduced the iteration indice p to obtain

$$\nabla\mathcal{F} = \left[\begin{array}{cc} \mathbb{I} & -\mathbb{W} \\ \mathbb{A}^p & \mathbb{B}^p \end{array} \right], \tag{9}$$

where $\mathbb{I}$, $\mathbb{A}^p$ et $\mathbb{B}^p \in \mathbb{R}^3$. Using a formulation based on the definition of Newton iterations, we can introduced (9) in a iterative scheme function of $\mathbb{A}^p$ and $\mathbb{B}^p$.

To determine $\mathbb{A}^p$, $\mathbb{B}^p$ and solve our problem, we must determine the different components of $\mathcal{F}$ as well as their partial derivative in regards of r_n, v_n, $\mathbf{r}_t$ and $\mathbf{v}_t$.

4.5 About granular dedicated solver

When granular assemblies are considered, it is possible to benefit of the geometry of particles to simplify the problem. When spherical particles are considered, it is not necessary to used complex solver because the resolution become explicit with or without friction. These dedicated strategies, well known in the numerical granular community, have been presented in [4] and we refer to this work for more details. Note that for more complex systems such as masonry or complex rigid body structure, this kind of explicit resolution cannot be used. Their excessive use can generate instabilities and leads to non physical behaviour. Thus previous methods present a large interest in the case of more general multi-body simulations.

5 SIMULATION RESULTS

To start our comparisons, we simulate a sphere packing. Also the geometry of particles is smooth, the process reaches from a dynamic phase to a static one. Moreover the contact number increases during the simulation. Thus algorithms must be well-suitable to this change of state. Figure presents results obtain with a dedicate solver (as reference result), QP solver, Lemke and Newton algorithm.

For this example the combination with the Newton method appears to be the more efficient. The QP method is the worse in term of CPU time. Lemke give good result but less efficient than the Newton method. Note that for this last one the choice in the maximal number of pivot is very important. Direct method must finish all pivot before giving a solution. On the over case, the result do not have any physical sense. This value is difficult to determine because any result of convergence exist for PSD matrices. Nevertheless, reasonable precaution assure to found a solution.

We also compare the different method on the simulation of a well-known mechanical phenomenon: the brazil nut effect (figure 3). To check the good mechanical result of simulations, we perform its with the different algorithm presented in section 4, compare the CPU time as well as the mechanical behaviour.

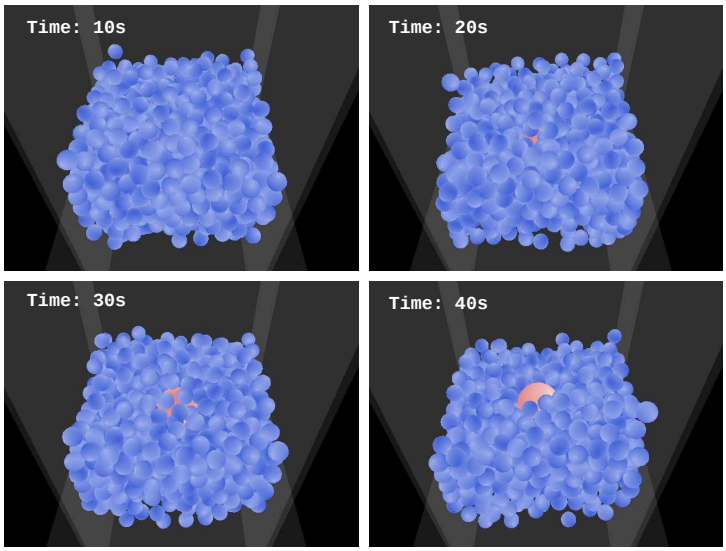

Figure 2: Reproduction of the brazil nut effect

The sample is composed of 800 spheres of radius equal to 1 and a big sphere of radius 2. All particles are in a box. Small particle lay on the big one. A periodic displacement is given to the floor to generate fluctuation in the sample. The motion of the big particle is tracked during the simulation.

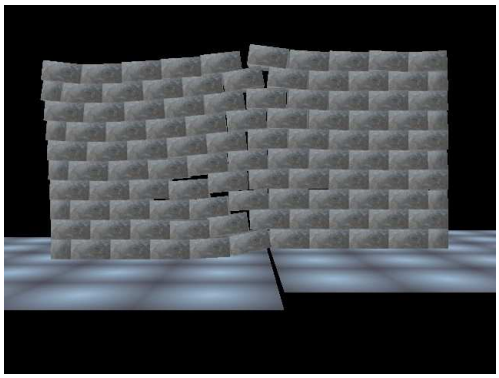

Figure 3: Numerical reproduction of Chateau Gombert experiment

As we mention on Sec. 4, spherical assemblies do not need complex algorithms due to the specific simplification induces by their geometry. But we want to consider masonry [3] or assemblies composed of more complex shape [2], this simplification cannot be considered. The example of masonry simulations as shown on figure 3b are very sensitive. The result must be the most accurate one to ensure the stability of the structure. Figure 3 represents the numerical reproduction of the experiment performed at Chateau Gombert [3]. The ground is composed of two part and one of the part go down. Thus we can observe block detachments in specific area of the wall. Experiment and simulation give similar results. The block splitting method combined to the Newton algorithm allow to perform the simulation in 1200s. Lemke and QP solver are slower than Newton method but results keep a correct physical behaviour.

6 CONCLUSION

A brief overview of combined algorithm have been presented and tested for the simulation of granular material. In a previous work the author have presented results underlining the efficiency of iterative method for the simulation of multi-body systems. The conclusions of this short study appear as the same. Also direct methods appear as efficient on small system, their integration in a iterative system does not lead to so good results. If result keep a physical sense, the CPU time of simulation is not reasonable. The Newton method appear as the only one to be the most well-suitable algorithm for a coupling with an general splitting method. CPU time as well as mechanical behaviour are preserved. To complete the study, combinations with the PATH solver [**?**] and NCP [21] will be performed.

Acknowledgment

Authors want to acknowledge SiCoNoS developers (http://siconos.gforge.inria.fr) for their help in the integration of the Numerics library in ConF&TiS (http://confeti.gforge.inria.fr), a three-dimensional rigid body simulation tool developed by the authors dedicated to virtual reality applications.

REFERENCES

[1] F. Pfeiffer and C Glocker. *Multibody dynamics with unilateral contacts*. Non-linear Dynamics. John Wiley and Sons, 1996.

[2] G. Saussine, F. Dubois, C. Bohatier, C. Cholet, P.E. Gautier, and J.J. Moreau. Modelling ballast behaviour under dynamic loading, part 1: a 2d polygonal discrete element method approach. *to appear in Computer methods in applied mechanics and engineering*, 2004.

[3] V. Acary and M. Jean. Numerical modeling of three dimensional divided structures by the non smooth contact dynamics method: Application to masonry structure. In B.H.V. Topping, editor, *The Fifth international Conference on Computational Structures Technology 2000*, pages 211–222. Civil-Comp Press, 6-8 September 2000.

[4] M. Renouf, V. Acary, and G. Dumont. Comparison of algorithms for collisions, contact and friction in view of real-time applications. In *Multibody Dynamics 2005 proceedings*, International Conference on Advances in Computational Multibody, Madrid, 21-24 June 2005 2005.

[5] J.-S. Pang and J.C. Trinkle. Complementarity formulations and existence of solutions of dynamic multi-rigid-body contact problems with Coulomb friction. *Math. Program.*, 73:199–226, 1996.

[6] M. Renouf, D. Bonamy, F. Dubois, and P. Alart. Numerical simulation of two-dimensional steady granular flows in rotating drum: On surface flow rheology. *Phys. Fluids*, 17:103303, 2005.

[7] M. Jean. The non smooth contact dynamics method. *Compt. Methods Appl. Math. Engrg.*, 177:235–257, 1999.

[8] M. Renouf, F. Dubois, and P. Alart. A parallel version of the Non Smooth Contact Dynamics algorithm applied to the simulation of granular media. *J. Comput. Appl. Math.*, 168:375–38, 2004.

[9] M. Renouf and P. Alart. Conjugate gradient type algorithms for frictional multicontact problems: applications to granular materials. *Comput. Methods Appl. Mech. Engrg.*, 194(18-20):2019–2041, 2004.

[10] R. W. Cottle, J. Pang, and R. E. Stone. *The linear complementarity problem*. Academic Press, Inc., Boston, MA, 1992.

[11] P. A. Cundall. A computer model for simulating progressive large scale movements of blocky rock systems. In *Proceedings of the symposium of the international society of rock mechanics*, volume 1, pages 132–150, 1971.

[12] J. J. Moreau. Unilateral contact and dry friction in finite freedom dynamics. In J.J. Moreau and eds. P.-D. Panagiotopoulos, editors, *Non Smooth Mechanics and Applications, CISM Courses and Lectures*, volume 302 (Springer-Verlag, Wien, New York), pages 1–82, 1988.

[13] D. E. Stewart and J. C. Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and Coulomb friction. *Int. J. Num. Methods Engrg.*, 39:2673–2691, 1996.

[14] M. Anitescu, F. A. Potra, and D. E. Steward. Time-stepping for the three dimensional rigid body dynamics. *Comput. Methods Appl. Mech. Engrg.*, 177:183–4197, 1999.

[15] S. P. Dirkse and M. C. Ferris. The path solver: A non-monotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software*, 5:123–156, 1995.

[16] A. Klarbring. A mathematical programming approach to three-dimensional contact problem with friction. *Compt. Methods Appl. Math. Engrg.*, 58:175–200, 1986.

[17] A. M. Al-Fahed, G. E. Stavroulakis, and P. D. Panagiotopulos. Hard and soft fingered robot grippers. the linear complementarity approach. *Z. angew. Math. Mech.*, 71:257–265, 1991.

[18] M. Renouf and P. Alart. Solveurs parallles pour la simulation de systmes multi-contacts. *Rev. Eur. Elem. Finis*, 13:691–702, 2004.

[19] S. Takriti and K. G. Murty. On the convergence of the block principal pivotal algorithm for the lcp. *Eur. J. Operational Research*, 102:657–666, 1997.

[20] P. Alart and A. Curnier. A mixed formulation for frictional contact problems prone to Newton like solution method. *Computer Methods in Applied Mechanics and Engineering*, 92(3):353–375, 1991.

[21] C. Glocker. Formulation of spatial contact situations in rigid multibody systems. *Comput. Methods Appl. Mech. Engrg.*, 177:199–214, 1999.

[22] K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Helderman-Verlag, Berlin, 1988.

[23] R. Fletcher. Resolving degeneracy in quadratic programming. *Annals of Operations Research*, 46–47(1–4):307–334, 1993.