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-structured algorithm, LCP, Direct method, Iterative method

Abstract. *Numerous works in Computational Mechanics are dedicated to the simulation of multi-body systems with contact and friction. The case of dense multi-contact assemblies is one of the more complex one: the problem have often a large number of unknowns and the non uniqueness of solutions is usual. Moreover this problem becomes harder when the Coulomb's friction or more complex laws are introduced. In this context, fast and robust solvers are required to carry out relevant mechanical studies. In a general way, these performances can be improved if the special structure of the problem can be exploited. For multi-body systems with contact and friction, the sparse block structured matrices involved in the time-discretised problem is one such structure.*

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

Methods are illustrated with the simulation of granular assemblies. All of these 3D frictional contact simulations are performed with ConF&TiS and the SICONOS/Numerics library.

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 the associated specific constraints leads to use a large panel of methods to simulate static and/or dynamic evolutions. The case of dense multi-contact assemblies is one of the more complex one. The problem have often a large number of unknown and the non uniqueness of solutions is usual. The first reason for the non-uniqueness is the redundancy of unilateral constraints leading to severe hyperstaticity, and non-definiteness of the matrices describing the system. The second reason is inherent to the Coulomb's friction law allowing multiple solution for the reaction forces at contact. Granular materials are typical examples of large and dense multi-body systems, where the previous mathematical troubles are encountered. To this mathematical difficulty, a physical one must be added. Indeed, very different behaviors which can jump from a solid one, in quasi-static evolution or low dynamic evolution of dense assemblies [2] to a liquid behavior during surface flows [6] have to be taken into account.

Due to the geometric structure of a multibody systems, the matrix describing the time discretized ad possibly linearized problem enjoys a block structure. This is mainly due to the fact that the connection between two bodies is only possible between adjacent bodies. In[7], a standard block splitting method is used, called the Non Smooth Block Gauss-Seidel algorithm. The numerical efficiency is critical (robust but slowly convergence) justifying studies and improvements [8]. Projected Conjugate Gradient[9] is faster for two-dimensional problems but does not allow in the case of three-dimensional frictional contact problems to obtain a large CPU time gain.

Usual LCP and QP methods for the frictionless case can be also extended to the frictional case using a approximation of the friction cone [5]. In this case, the block structure of the matrices is still obtained. Direct methods such as Lemke's method can also be used but need special pivot research [10]. In the multi-contact case, a pivoting method appears to be slower than iterative ones, especially for a large number of unknown [4]. Quadratic programming solvers are also an alternative but present similar efficiency than Lemke's algorithm. For small systems, direct methods are often more efficient and faster than iterative ones especially for very difficult and ill-conditioned problem.

The aim of this paper is to present a combination of algorithms which takes benefit from the block structure of three-dimensional frictional contact problem. This combination is performed using different local algorithms for each blocks (direct or iterative methods) within a global standard block splitting method. After a presentation in Section 2 of the framework used for our three-dimensional frictional contact problem, the block structure formulation is presented in details in Section 3 with and without an approximation of the friction cone. Section 4 gives an overview of the different algorithms. Several results are presented in Section 5 on granular materials samples and Section 6 concludes the paper.

2 FORMULATION OF THE FRICTIONAL CONTACT PROBLEMS

In Computational Mechanics, among well-suited approaches for the rigid body dynamics with contact, friction and impact, two opposite approaches are found: compliant versus unilateral contact model and event-driven versus time-stepping schemes. In the context of granular materials, where large collections of bodies are encountered, Cundall [11] was the first to propose a numerical tool based on an Euler scheme and where contacts are governed by a compliant model. With a definitely different approach, Moreau [12] and Jean [7] exposed a numerical treatment of rigid and deformable body collections with unilateral contact, Coulomb's dry friction and impact in the framework of the Nonsmooth Mechanics and Convex Analysis. This framework yields a time-stepping scheme (without explicit event-handling) where velocities and impulses are the primary variables. Still in a nonsmooth framework, Pfeiffer and Glocker [1] and Stewart and Trinkle [13] designed event-driven algorithms for the time integration and proposed a general formulation for the dynamics at the acceleration-force level. The resolution of the time-discretized problem is performed using a direct method for LCP [5], a quadratic programing solver [14] or a MCP solver [15]. Focusing on the resolution of the time-discretized problem, the choice of the time integration scheme is out the scope of this paper.

The method used in this work to simulate our multi-body systems is based on the *Non Smooth Contact Dynamics* (NSCD) framework previously evoked. This time-stepping scheme is a very efficient numerical tool for a great number of applications that are well-known for theirs difficulties [3, 2]. The headlines of the method are presented and we refer to [12, 7] for detailed explanations.

2.1 Discretization of the equations of motion

The formalism of the NSCD method relies on a special formulation of the equation of motion. When multi-contact systems are considered, shocks may be expected during the evolution of the system. These shocks lead to velocity discontinuities and preclude to define the acceleration as the usual second time derivative of the configuration parameter, denoted in this paper by $\mathbf{q}$. Consequently the equations 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 $d\mathbf{R}$ is a non-negative real measure, representing the reaction forces and impulses. In the equation (1), the matrix $\mathbb{M}$ represents the mass matrix and $\mathbf{F}(t, \mathbf{q}, \dot{\mathbf{q}})$ the internal and external forces acting on the system. For the sake of readability, $\mathbf{F}$ is only given by external loads depending on time. The general case can be easily tackled through a linearizing procedure such as a Newton's method.

In this way, the equation (1) is integrated on each time interval $]t_i, t_{i+1}]$ and approximated using a θ method, an implicit first-order scheme, using the configuration parameter and its first derivative [7]. A 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

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

The vector $\dot{\mathbf{q}}^{free}$ denotes the free velocity (computed velocities 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 $\mathbb{H}^*$. The local forces vector $\mathbf{r}$, expressed in the local frame, are related to $\mathbf{R}$ by the relation

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

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 summarized 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 and α is the index over the contact set. The right-hand-side of the first equation in (5) represents the free relative velocity. 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 BLOCK STRUCTURE FORMULATION OF THE DELASSUS OPERATOR

Within the NSCD framework, the Delassus operator enjoys a block structure. When this theoretical structure is linked to the physical structure of a granular material, each block on the diagonal of the operator contains the local information of each contact, and the blocks out of the diagonal represent the connectivity between coupled contacts. The explicit formulation of the Delassus operator used in the problem (5) is presented in this section. This explicit formulation is addressed first for a one contact case in a non approximated problem and after for an approximated one and then for the multi-contact case.

3.1 The one-contact case

3.1.1 Standard Delassus operator

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 $\mathbb{W}$ one obtain

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

Here the contact index is omitted to keep a pleasant reading.

3.1.2 Inner approximation of the friction cone

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 made. with such an approximation, some existence of solutions can be exhibited [13].

Different kinds of approximations can be performed: uniform or not, inner or outer. The global form of the matrix is described, using an regular inner approximation as it may be represented on Figure 1. First the approximated friction cone is defined 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

$$\mathbb{D} = \left[\ \mathbf{d}_1 \,\middle|\, \ldots \,\middle|\, \mathbf{d}_\nu\ \right].$$

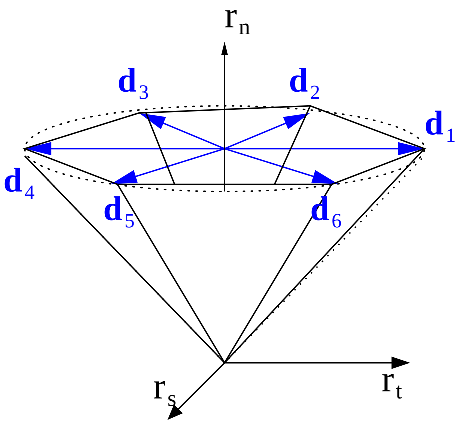

Figure 1: Inner approximation

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

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

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

Note that this kind of approximation leads to increase the number of unknowns due to the introduction of slack variables. On the behavioral point of view, such approximation may generate some problems with a strong impact in special applications such as haptic control [4] due to the introduction of an anisotropy of the approximated friction cone.

3.2 Multi-contact strategy

When multi-contact assemblies are considered, $\mathbb{W}$ presents a block structure ($n_c \times n_c$ blocks, if n_c denotes the contact number). This structure is a concatenation of both null and $\mathbb{W}_{\alpha,\beta}$ matrices where the last ones are defined as

$$\mathbb{W}_{\alpha\beta} = \begin{bmatrix} \mathbb{H}^*_{T,\alpha}\mathbb{M}^{-1}\mathbb{H}_{T,\beta} & \mathbb{H}^*_{T,\alpha}\mathbb{M}^{-1}\mathbb{H}_{N,\beta} \\ \mathbb{H}^*_{N,\alpha}\mathbb{M}^{-1}\mathbb{H}_{T,\beta} & \mathbb{H}^*_{N,\alpha}\mathbb{M}^{-1}\mathbb{H}_{N,\beta} \end{bmatrix}.$$

When $\alpha = \beta$, the $\mathbb{W}_{\alpha\beta}$ are equal to matrices defined in the one contact case. Using this definition, the global matrix is equal to

$$\mathbb{W} = \begin{bmatrix} \mathbb{W}_{11} & \mathbb{W}_{12} & \mathbb{W}_{13} & \dots & \mathbb{O} \\ \mathbb{W}_{21} & \mathbb{W}_{22} & \mathbb{O} & \dots & \mathbb{W}_{2n_c} \\ \mathbb{W}_{31} & \mathbb{O} & \mathbb{W}_{33} & \dots & \mathbb{W}_{3n_c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbb{O} & \mathbb{W}_{n_c2} & \mathbb{W}_{n_c3} & \dots & \mathbb{W}_{n_cn_c} \end{bmatrix}.$$

Non null matrices on each block line represent the connectivity between the set of contacts. When approximation of the friction cone is used, the size is related to the accuracy of the approximation, i.e. to the dimension of the matrix, $\mathbb{D}$. Usually when pivoting methods are used to solve the problem the whole matrix must be considered at each operation. The idea here is to keep the structure found in [7] and to extend it towards combinations using LCP or QP solvers. Consequently if an approximated problem is considered, the approximation is performed at the local level only. The size of the problem is reduced by considering only the approximation of the local Delassus operator and not of the whole matrix. Furthermore, if only small problems are considered , problems where pivoting methods can be efficient. Moreover the construction of each $\tilde{\mathbb{W}}$ can be performed only one times to preserve CPU time. Only the right hand side of problem (5) has to be updated to take into account the evolution of the system. From this point of view, this method is different from the theoretical results and the methodology presented in [19].

4 BLOCK SPLITTING METHOD AND LOCAL ALGORITHMS PANEL

4.1 Block Splitting method

To solve problem (5) a general Block Splitting Method (BSM) is applied on matrix $\mathbb{W}$ or $\tilde{\mathbb{W}}$. The idea is to solve the local frictional contact problem using an algorithm such as generalized Newton methods [20], Lemke's method [10] and to use a block splitting method to solve contact by contact each local problems. Method such as CPG algorithm [9] or solvers based on NCP formulation [21] will be not considered here.
Using the notation of system (5), the global splitting scheme is written down as follows

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

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, the algorithm used to solve the local frictional contact problem must be defined.

```
Combined Block Splitting Scheme

Step 0 :  Initialization
          Compute block matrices $\mathbb{W}_{\alpha\alpha}, \mathbb{W}_{\alpha\beta},\ \alpha = 1,..,n, \beta = 1,..,n$
Step 1 :  CBSM iteration (index $k$)
          Start block iteration (index $\alpha$)
            | Compute   $\mathbf{b}_\alpha^k$ right hand side of equation (6)
            | Solve     $Frictional\ Contact\ Problem(\mathbb{W}_{\alpha\alpha}, \mathbf{b}_\alpha^k)$
Step 2 :  Check convergence
          If convergence or $k = k_{max}$ Stop else go to Step 1
```

4.2 Lexicographic Lemke

Pivoting methods are often used to solve LCP [10], but obtaining good results in terms of efficiency with such methods depends strongly on the properties of the matrix $\mathbb{W}$. In multi-contact assemblies (of rigid bodies), the matrix $\mathbb{W}$ is almost always a positive but only semi-definite (PSD) matrix. Especially, The redundancy of constraints leads to degenerate problems. In this case, classical pivoting methods cannot be used and some adaptation in pivoting algorithms

must be taken into account to avoid cycling during the pivoting process.
To face this problem with the Lemke's 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 interest of Quadratic Programming solver relies on the minimization principle and the descent method which ensures a strong stability to the algorithm. Moreover, reliable modifications of such algorithms are usual to overcome the non-definiteness of the matrix. To face this problem a QP solver based on the generalized Fletcher's method [23] is used. These improvements ensure an algorithm termination extending to the case of round-off error. However, to guarantee that the solution of the LCP is also a solution of the QP, the matrix of the LCP need to be monotone. Especially, for frictional simulation, the non symmetry of the Delassus operator leads to double the number of constraints and the monotonicity of the Delassus operator is not ensured.

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) may be formulated as a root finding problem for a non smooth function $\mathcal{F}(\mathbf{v},\mathbf{r})$ defined 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 $\gamma = \mathbf{r} - \rho\mathbf{u} = \gamma_n\mathbf{n} + \gamma_t, \rho > 0$ et $\gamma_n^+ = proj(\gamma_n; \mathbb{R}^+)$ et $\gamma_t \in \mathbb{R}^2$. 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, the different iterate $(\mathbf{X}^p)_p$ are defined 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 the iteration index p is introduced 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, the equation (9) is introduced in an iterative scheme function of $\mathbb{A}^p$ and $\mathbb{B}^p$.

To determine $\mathbb{A}^p$, $\mathbb{B}^p$ and to solve our problem, the different components of $\mathcal{F}$ as well as their partial subgradients must be defined 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 take benefit of the geometry of particles to simplify the problem. When spherical particles are considered, it is not necessary to used a complex solver because the resolution become explicit with or without friction. These dedicated strategies, well known in the granular materials community, have been presented in [4] where more details can be obtained. It's noteworthy that for more complex systems such as masonry or complex rigid body structure, this kind of explicit resolution cannot be used. Therefore previous methods present a large interest in the case of more general multi-body simulations.

5 SIMULATION RESULTS

To start our comparisons, a sphere packing is simulated. Although the geometry of particles is simple and smooth, the process of packing is interesting form a numerical point of view because of the transition from a dynamic flow to a static equilibrium. Moreover the contact number increases quickly during the simulation.

Figure 2 depicts results obtained with a dedicated solver (as a reference result), QP solver, Lemke and Newton algorithm during a sphere settling. The time of BSM resolution is displayed in function of the size of $\mathbb{W}$ i.e. the contact number.

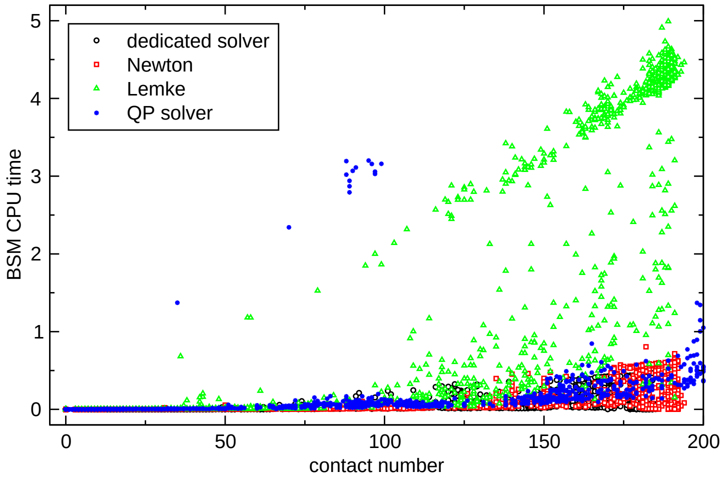

Figure 2: Evolution of time of BSM resolution in regard of the contact number

For this sample, the combination with the generalized Newton method appears to be the more efficient. The Lemke's method is the worse in terms of CPU time. QP solver provides us with correct results but is less efficient than the Newton's method. It's noteworthy that for pivoting methods the maximal number of pivots is crucial. Indeed, direct methods must complete all pivoting steps before giving a solution. Otherwise, the result do not have any physical sense. This value is difficult to size because no result of convergence exists for PSD matrices. Nevertheless, reasonable precautions assure to find a proper solution reducing the CPU time.

The different methods are also compared on the simulation of a well-known mechanical phenomenon: the Brazil nut effect (See Figure 5). To check the validity of the results, several simulations with the different algorithms presented in Section 4 are performed and comparisons in terms of CPU time as well as mechanical behavior are done.

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 and small particles lay on the big one. A periodic displacement is prescribed to the floor and generate fluctuations in the sample. The motion of the big particle is tracked during the simulation.

Figure 4 depicts the evolution of the BSM resolution time during the simulation process. As the obtained results with the Lemke's method are too much expensive in CPU time, their evolution is not plotted. Results obtain with the QP solver are more expensive that the ones obtained with Newton or dedicated solver. However, in each case the mechanical behavior is reproduced.

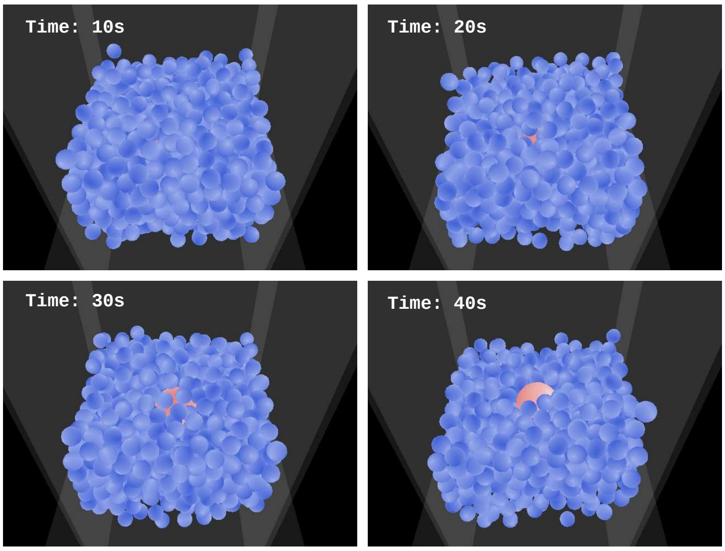

Figure 3: Reproduction of the Brazil nut effect

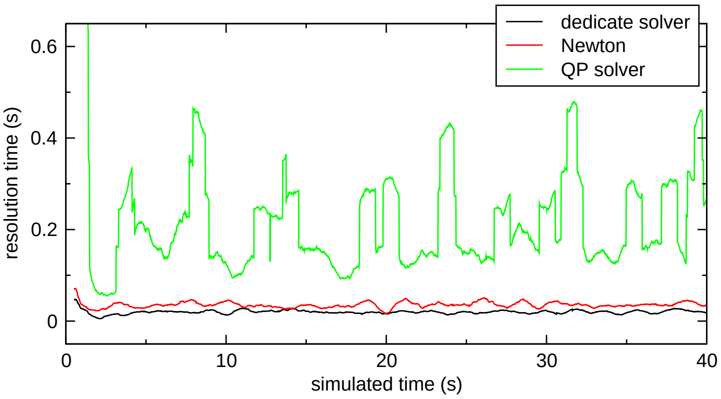

Figure 4: Evolution of the time take by the resolution of the BSM during the simulation process.

As mention in Section 4, spherical assemblies do not need complex algorithms due to the specific simplification induced by their geometry. Then masonry [3] or assemblies composed of more complex shape [2] for which this simplification is not relevant are considered.

The example of masonry simulations as shown on figure 5 is very sensitive. The result must be the most accurate possible to ensure the stability of the structure. Figure 5 depicts the numerical reproduction of the experiment performed by M. Jean [3]. The ground is composed of two separate part and one of the part settles vertically. Block detachments and fractures in specific area of the wall are observed. Experiment and simulation give similar results.

The block splitting method combined to the Newton algorithm allow us to perform the simulation in 1200s. Lemke's method and QP solver are slower than Newton method but results keep a correct physical behavior.

6 CONCLUSION

A brief overview of combined algorithms has been presented and tested for the simulation of granular materials. In a previous work the authors have presented results underlining the

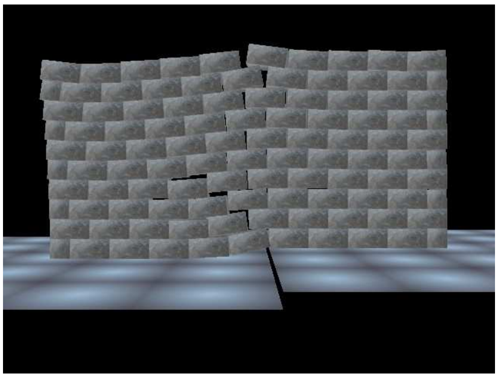

Figure 5: Numerical reproduction of Chateau Gombert experiment

efficiency of iterative methods for the simulation of multi-body systems. The conclusions of this short study are similar. Although direct methods appear to be efficient on small systems, their integration in a iterative system does not lead to improve efficiency. If the result keeps a physical meaning, the CPU time of simulation is not reasonable. Perhaps, such method may be interesting in very ill-conditioned problem. The Newton's method appears to be the most well-suited algorithm for a coupling with a general splitting method. The CPU time as well as mechanical behavior are satisfying. To complete the study, combinations with the PATH solver [15] and NCP [21] will be performed.

Acknowledgment

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

[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. Stewart. 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 parallèles pour la simulation de systèmes 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.