



Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials

Mathieu Renouf, Pierre Alart

► To cite this version:

Mathieu Renouf, Pierre Alart. Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials. *Computer Methods in Applied Mechanics and Engineering*, 2005, 194 (18-20), pp.2019 - 2041. 10.1016/j.cma.2004.07.009 . hal-01393138

HAL Id: hal-01393138

<https://hal.science/hal-01393138>

Submitted on 6 Nov 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials

Mathieu Renouf, Pierre Alart

Laboratoire de Mécanique et Génie Civil, Equipe Systèmes Multi-Contacts, Université Montpellier 2, UMR 5508, CNRS,
cc048, Place Eugène Bataillon, 34095 Montpellier Cedex 05, France

This paper presents gradient type algorithms to solve frictional multi contact problems written as quasi optimization problems. A single loop scheme formally close to the classical conjugate gradient method is proposed with some adaptations of the iterate corrections and gradient projections. Since the convergence is difficult to prove, various tests in the field of granular media are performed with comparison with the non linear Gauss Seidel scheme.

Keywords: Conjugate gradient; Optimization; Constraints; Multi contact; Granular media

1. Introduction

In computational structural mechanics contact problems are often presented as difficult to solve even if the contact is localized on a limited part of the boundary of bodies. This is due to the strong non-linearity of the frictional contact laws involving multi-valued relationships between kinematic and static variables on the potential contact area. Such a difficulty imposed to develop a wide range of specific numerical tools; a large review is given in the context of finite element modeling in [42]. But the situation is more complex when we have to solve multi-contact problems. *Multi-contact problems* indicate situations when many contacts occur, especially when the number of contacts exceeds the number of degrees of freedom of the system. This last case refers essentially to granular media [10,22,35]. But this definition may be extended to cases when the contact is diffuse through the system on numerous contact zones even if the ratio contacts

nodes/degrees of freedom is small: it concerns assembly of structures [1,8], micro-cracked media [24] or cellular media [4].

Because the size of the problems we have to treat increases regularly, the parallel computational techniques appear as a first attempt to reduce the computational time and to deal with larger and more realistic problems. The direct simulation stays an inevitable way to simulate such structures even if it may lead to expansive (even impossible) computations. Indeed for cellular and granular materials effective behavior laws are not available via homogenization procedures. For granular media the definition of a representative sample is still an open question and some homogenization procedures may be only applied to static behavior of granulates [10,11,32]; recall that a granular medium may behave as a solid or a fluid according to the external solicitations. The evolution of a granular medium is an erratic process with multiple possible paths, localization phenomena (shear bands) and local dynamic crisis (arching collapses). We have to capture all these events to get a better understanding of the behavior of such materials and structures. Consequently we have to improve the numerical tools to solve frictional contact problems and especially the algorithms which may be intrinsically parallel.

A first study consisted in carrying out a parallel version of the non-linear Gauss Seidel algorithm associated with the *Non-Smooth Contact Dynamics* approach developed by Jean and Moreau [20,30]. This method is intrinsically sequential and we tried a simple multi-threading technique which consists in splitting the contact loop into several threads dedicated to different processors. This approach was tested and analyzed in [38]: the performance of the solver is weakly perturbed. But Gauss Seidel algorithms are not the most efficient methods to solve large linear systems. By analogy with the linear solvers *conjugate gradient* algorithms could be good candidates as non-smooth solvers for large multi-contact problems preserving moreover parallel treatment. The purpose of this paper is to investigate this topic. Conjugate gradient methods have been used to solve friction-less contact problems in the eighties [16,26] in the context of finite element modeling. An extension for frictional contact is proposed in [36], but it is restricted to structural problems using the displacement field as the main unknown; moreover a fixed point loop is introduced for satisfying the Coulomb law, and the friction law is regularized to be inserted as a non-quadratic but smooth additional term in the energy to minimize. We propose in the following a single loop algorithm very close to a standard conjugate gradient method with local adaptations consisting in iterate corrections and gradient projections which are in part theoretically justified in Sections 2 and 3 and detailed in Section 4. Section 5 is dedicated to the implementation and Section 6 collects numerical tests on the simulation of various processes involving granular media.

2. Constrained (quasi) optimization problems in contact mechanics

Using optimization techniques requires expressing contact problems as optimization problems by resorting a convex analysis formalism. Attention is restricted to discrete problems obtained either from an approximation technique (as the finite element method) or directly from the modeling of a finite-freedom mechanical system. As presented in a previous paper [4], the methods chosen to solve non-smooth equations depend on the features of the problem. It is then useful to distinguish two extreme typical problems involving both unilateral constraints.

2.1. Structural and granular type frictionless problems

In [4], we present two typical problems, called structural and granular type problems. The so called *structural type problem* consists in determining the equilibrium of an elastic body in grazing contact with a rigid foundation. The integer n notes the total number of degree of freedom of the elastic body, $m = n_c$ is the number of potential contact nodes. The equilibrium equations consists of a linear system (in a first approx-

imation): stiffness matrix \mathbb{K} , generalized displacement \mathbf{u} , normal contact reactions \mathbf{r} on the nodes of the contact area, the external forces \mathbf{f} ; this constitutes the first line of the system to solve

$$\begin{cases} \mathbb{K}\mathbf{u} - \mathbb{H}\mathbf{r} = \mathbf{f}, \\ \mathbb{H}^T\mathbf{u} \geq \mathbf{0}, \quad \mathbf{r} \geq \mathbf{0}, \quad \mathbf{r} \cdot \mathbb{H}^T\mathbf{u} = 0, \end{cases} \quad (1)$$

The matrix \mathbb{H} passes from the local frame to the global one for all potential contact nodes. We have to add the contact conditions written here with two inequalities and a complementarity condition between the contact reactions and the normal components of the displacements on the potential contact surface. In structural mechanics we have often to add other non-linearities issued from the body (the internal forces $\mathbb{K}\mathbf{u}$ become a non-linear function of the displacement $\mathbb{K}(\mathbf{u})$) or from the contact geometry when strong curvatures have to be taken into account: we have then $\mathcal{G}(\mathbf{u})$ instead of $\mathbb{H}^T\mathbf{u}$ and $\nabla(\mathcal{G})^T\mathbf{r}$ instead of $\mathbb{H}\mathbf{r}$. Such additional non-linearities lead to use Newton type methods, either extended to non-differentiable equations [3,14,33], or combined with specific contact methods: Gauss Seidel [20,21,36], Augmented Lagrangian [13,25].

The situation is quite different for *the granular type problem*. Following the approach of Moreau [28], the main object of the computation is the velocity and a time stepping method is used as time integrator. The reader is referred to [9,30] for a discussion about the different integrators. For comparison with the previous case we consider the problem on a single time step; we have to predict the velocities distribution in a collection of rigid bodies at the end of this step. The contact operates now as an interaction law between particles. The dynamic equations may be easily written at the contact points using the local frame (see below) by considering the local variables: the relative velocity \mathbf{u} between the two particles passing at the contact point at this time and the impulsion \mathbf{r} . The system to solve is then the dynamics reduced to contacts

$$\begin{cases} \mathbb{W}\mathbf{r} - \mathbf{u} = \mathbf{b}, \\ \mathbf{r} \geq \mathbf{0}, \quad \mathbf{u} \geq \mathbf{0}, \quad \mathbf{r} \cdot \mathbf{u} = 0. \end{cases} \quad (2)$$

The right-hand side of the first line in (2) takes into account the external forces and the relative velocity at the end of the previous step. In dense granulates, the number of contacts n_c is larger than the number of bodies n_b ; the mass matrix \mathbb{M} belongs to $\mathbb{R}^{n \times n}$ with $n = 3n_b$ or $6n_b$ according to the modeling (2D or 3D). Since the mass matrix \mathbb{M} is diagonal, it is easily invertible and the reduced system with the matrix \mathbb{W} may be considered to be solved, $\mathbb{W} = \mathbb{H}^T\mathbb{M}^{-1}\mathbb{H}$. But the \mathbb{W} matrix may be singular giving rise to “wedging” effects and indeterminacy of the impulsions. Generally $\mathbb{H}(\mathbf{q})$, depending on the configuration parameter \mathbf{q} of \mathbb{R}^n , is maintained constant over the time step. The structural problem (1) may be reformulated as an optimization problem where the solution \mathbf{u} is the argument of the minimum of the quadratic function $\phi(\cdot)$ over the set defined by the constraint $\mathbb{H}^T\mathbf{u} \geq \mathbf{0}$

$$\mathbf{u} = \arg \min_{\substack{\mathbb{H}^T\mathbf{u} \geq \mathbf{0}}} \frac{1}{2} \tilde{\mathbf{u}} \cdot \mathbb{K} \tilde{\mathbf{u}} - \mathbf{f} \cdot \tilde{\mathbf{u}}. \quad (3)$$

Generally the matrix \mathbb{K} is positive definite and the solution is unique; that explains the equality. The granular problem (2) is so reformulated as an optimization problem using the contact impulsions \mathbf{r} as the primal variable

$$\mathbf{r} \in \arg \min_{\mathbf{r} \geq \mathbf{0}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (4)$$

In granular systems the matrix \mathbb{W} is often only semi-positive definite leading to the indeterminacy of the impulsions; we have to determine one solution among several admissible ones. In fact the number of solutions is infinite and the algorithms have to reach one solution that we hope to be representative of the global behavior of the medium. A first study concerning the parallel treatment of the Gauss Seidel type method shows that the macroscopic features of the granular medium are preserved even if different deformation

paths are followed when several processors are used [38]. More generally the Gauss Seidel type algorithms revealed to be efficient for granular problems [20,30], they converge even if the matrix is singular [12], but the optimization techniques are recommended in this case as long as the function stays lower bounded, which may be proved.

2.2. Frictional conditions

At first we complete the contact description by splitting the matrix \mathbb{H} from the local frames to the global one into its normal and tangential components: $\mathbb{H} = [\mathbb{H}_n, \mathbb{H}_t]$. The friction laws link the tangential velocity to the tangential component of the contact impulsion. The Tresca law postulates a non-negative friction threshold, noted s_α for each contact locus α , below which the sliding is precluded. In an incremental form the frictional contact problem is yet a constrained optimization problem involving now a non-differentiable objective function

$$\mathbf{u} = \arg \min_{\mathbb{H}^T \mathbf{u} \geq 0} \phi(\tilde{\mathbf{u}}) + \sum_{\alpha=1}^{n_c} s_\alpha \|\mathbb{H}_{\alpha,t}^T(\tilde{\mathbf{u}} - \mathbf{u})\|, \quad (5)$$

where \mathbf{u} notes the displacement at the previous step and $\mathbb{H}_{\alpha,t}^T$ is the tangential contribution of the contact α in \mathbb{H}^T ; $\mathbb{H}_{\alpha,t}^T(\tilde{\mathbf{u}} - \mathbf{u})$ corresponds then to a local slip increment.

For granular type problem a Tresca like law provides a simpler formulation

$$\mathbf{r} \in \arg \min_{\mathbf{r} \in \mathcal{C}(s)} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}, \quad (6)$$

where only the constraint set is modified in comparison with the problem (4); it is the cartesian product of infinite half cylinders in \mathbb{R}^2 or \mathbb{R}^3 with a section $\mathcal{B}(0, s_\alpha)$, disk of radius s_α

$$\mathcal{C}(s) = \prod_{\alpha=1}^{n_c} \mathbb{R}^+ \times \mathcal{B}(0, s_\alpha). \quad (7)$$

But the more classical Coulomb law links the friction threshold to the normal component of the contact force (or impulsion) via a friction coefficient μ . The coupled frictional contact problem is not an optimization problem anymore. Formally it is always possible to formulate a “quasi”-optimization problem (in reference to more classical quasi-variational inequalities which derive from it) for which the constraint set depends on the normal components of the solution as a parameter; only the *granular type frictional contact problem* is given

$$\mathbf{r} \in \arg \min_{\mathbf{r} \in \mathcal{C}(\mu n)} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (8)$$

The non-uniqueness of the solution is due to the singularity of the matrix \mathbb{W} mentioned above and to the non-associated character of the slip rule according to several studies on this aspect [2,5,15,18,31]. The normality rule of the gradient of the objective function with respect to the conical constraint set is not satisfied as shown in Fig. 1; the gradient \mathbf{g} of the objective function ($\mathbf{g} = \mathbb{W} \mathbf{r} - \mathbf{b}$) is normal to an half-cylinder.

In the following we consider a *standard constrained quadratic problem* using the notations of the granular type problem

$$\mathbf{r} \in \arg \min_{\mathbf{r} \in \mathcal{C}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (9)$$

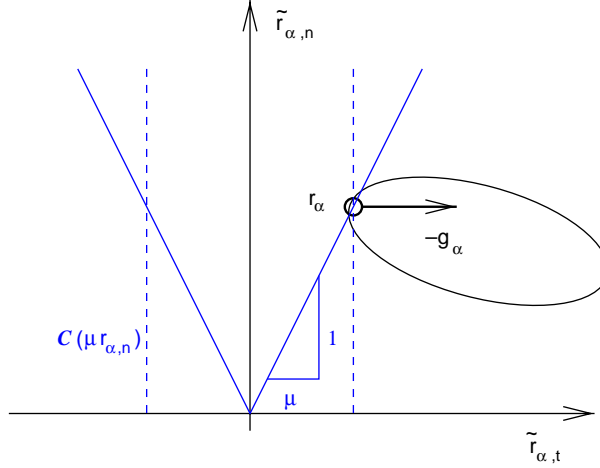


Fig. 1. Coulomb's cone and slip rule.

It is interesting to specify the geometric features of the constraint set \mathcal{C} according to the kind of the problem. If the unilateral contact is only accounted for, the set \mathcal{C} is the non-negative cone or the non-negative orthant. If friction is only considered (the contact is assumed to be maintained) the convex set is an hyper rectangle. For a Tresca like frictional contact problem in a bidimensional modeling the set \mathcal{C} is the cartesian product of infinite half-bands in \mathbb{R}^2 . For these three cases the set \mathcal{C} is a polyhedral convex set. For a three-dimensional Tresca like frictional contact problem the set \mathcal{C} is the cartesian product of infinite half-cylinders in \mathbb{R}^3 and it is no more polyhedral. For a Coulomb frictional contact law, the Tresca like frictional contact problem may be viewed as an intermediate problem in a numerical solution strategy.

3. (Projected) gradient methods

The simplest gradient method to solve (9) is the gradient method with projection defined by

$$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}[\mathbf{r}^k + \gamma(\mathbf{b} - \mathbb{W}\mathbf{r}^k)] \quad \forall \gamma > 0. \quad (10)$$

The gradient will be noted $\mathbf{g}^k = \mathbb{W}\mathbf{r}^k - \mathbf{b}$ and the descent direction is $\mathbf{u}^k = -\mathbf{g}^k$. The projection on the admissible set concerns the iterate but not the gradient. The scheme converges if the positive parameter γ is less than $\frac{2}{\lambda_{\max}(\mathbb{W})}$, where $\lambda_{\max}(\mathbb{W})$ is the largest eigenvalue of \mathbb{W} . The convergence rate is low especially if the matrix \mathbb{W} is not well-conditioned (cf. Fig. 2).

3.1. Generalized Rosen methods

The first class of improved algorithms consists of projected gradient methods. For a better understanding, let consider a single linear constraint, $\mathbf{d} \cdot \mathbf{r} \geq 0$, where \mathbf{d} is a unit vector. We can prove that the vector $\mathbf{p}^k := \text{proj}_{\mathcal{C}}(\mathbf{r}^k - \gamma\mathbf{g}^k) - \mathbf{r}^k$ is yet a descent direction ($\mathbf{p}^k \cdot \mathbf{g}^k < 0$) and, if $(\mathbf{r}^k - \gamma\mathbf{g}^k) \cdot \mathbf{d} < 0$, this new direction is the projection of the gradient on the hyper plane boundary of the half-space constraint set: $\mathbf{p}^k = \gamma[-\mathbf{g}^k + (\mathbf{d} \cdot \mathbf{g}^k)\mathbf{d}]$; the constraint is called active. One can appreciate the interest in restricting the research to a subspace of active constraints when we deal with numerous linear constraints. Following the ideas developed by Rosen for linear [39] and non-linear [40] constraints, Mehrez [41] named generalized Rosen method the scheme defined with two parameters α and γ

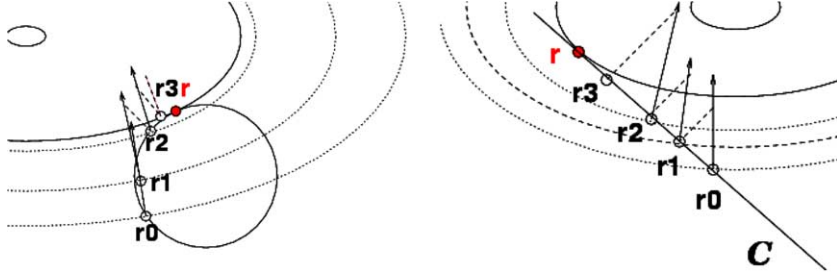


Fig. 2. Gradient method with projection for linear and non linear constraints.

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \alpha[\text{proj}_{\mathcal{C}}(\mathbf{r}^k - \gamma \mathbf{g}^k) - \mathbf{r}^k] \quad \forall \gamma > 0, \forall \alpha > 0. \quad (11)$$

The convergence is proved with some conditions for these parameters [27]; two extreme cases may be specified,

$$\begin{cases} \text{case 1: } \alpha = 1 & 0 < \gamma < \frac{2}{\lambda_{\max}(\mathbb{W})}, \\ \text{case 2: } 0 < \alpha < \min\left(\frac{2}{\lambda_{\max}}, \frac{2}{2 - \lambda_{\min}}\right) & \gamma = 1. \end{cases}$$

The convergence may be still slow if such conditions are imposed. Moreover the parameter α has to belong to the interval $[0, 1]$ insuring that \mathbf{r}^{k+1} stays in the constraint set. This restriction may be combined with a varying optimal step for the parameter α^k in order to accelerate the convergence. But such a strategy may lead to a non-optimal solution as proved by Wolfe with a simple counterexample [7]. It is surer to go outside and then to project once again on the admissible set to get the following scheme:

$$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}\{\mathbf{r}^k + \alpha^k[\text{proj}_{\mathcal{C}}(\mathbf{r}^k - \gamma^k \mathbf{g}^k) - \mathbf{r}^k]\}, \quad \forall \gamma^k > 0, \forall \alpha^k > 0. \quad (12)$$

This scheme is quite general but it does not use the polyhedral character of our convex set except in 3D and it does not allow introducing conjugate gradient. Moreover the optimal choice of the parameter γ is an open question. But Fig. 3 shows that for a small enough γ and for a polyhedral set, the descent direction \mathbf{p}^k does not depend on γ : the gradient \mathbf{g}^k , or more precisely the descent direction $\mathbf{u}^k = -\mathbf{g}^k$, is projected on the tangent cone to \mathcal{C} in \mathbf{r}^k . We note the tangent cone $T_{\mathcal{C}}$ as the polar cone to the normal cone $N_{\mathcal{C}}$. This remark leads to postulate a local version of a generalized Rosen algorithm or a Projected Gradient method, combining the previous projection and an optimal choice for the parameter α^k (cf. Table 1).

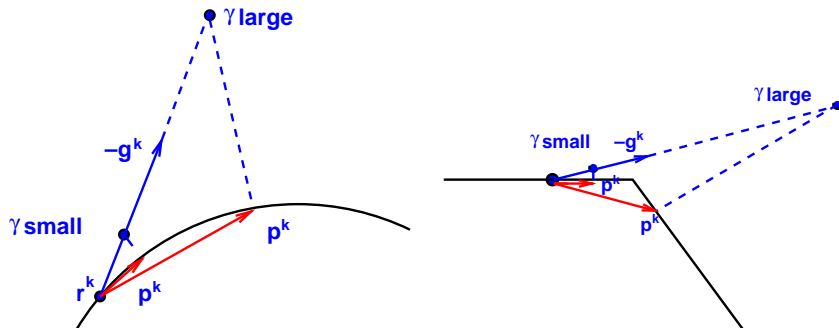


Fig. 3. Influence of the magnitude of γ on the descent direction for smooth or polyhedral constraint set.

Table 1
Projected gradient (PG) algorithm

$\mathbf{r}^0, \mathbf{u}^0$ and \mathbf{p}^0 given	
$k \rightarrow k+1$	
$\mathbf{r}^{k+\frac{1}{2}}$	$\mathbf{r}^k + \alpha^{k+1} \mathbf{p}^k$ with $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k}$
\mathbf{r}^{k+1}	$\text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$
\mathbf{u}^{k+1}	$\mathbf{b} - \mathbb{W} \mathbf{r}^{k+1}$
\mathbf{p}^{k+1}	$\text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$

3.2. A conjugate projected gradient algorithm

To improve the convergence rate we introduce a conjugate gradient type approach by conjugating the previous descent direction \mathbf{p}^{k-1} with the current gradient \mathbf{u}^k after projecting them on the tangent cone, and so defining a conjugate projected gradient (CPG) algorithm (cf. Table 2).

The conjugating process is a priori fully efficient if the iterates stay during several iterations in the same set of active constraints, that is to say on the same facet or edge of the constraint set. An interesting case is the set of active constraints to which the solution belongs; this set corresponds in contact mechanics to the set of the solution statuses contact by contact. After this set is found the Gauss-Seidel algorithm is slow to reach the solution; we can hope in this case a better behavior of the conjugate gradient algorithm; numerical

Table 2
Conjugate projected gradient algorithm

$\mathbf{r}^0, \mathbf{u}^0$ and \mathbf{p}^0 given	
$k \rightarrow k+1$	
$\mathbf{r}^{k+\frac{1}{2}}$	$\mathbf{r}^k + \alpha^{k+1} \mathbf{p}^k$ with $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k}$
\mathbf{r}^{k+1}	$\text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$
\mathbf{u}^{k+1}	$\mathbf{b} - \mathbb{W} \mathbf{r}^{k+1}$
\mathbf{w}^{k+1}	$\text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$
\mathbf{z}^{k+1}	$\text{proj}(\mathbf{p}^k; A(\mathbf{w}^{k+1}))$
\mathbf{p}^{k+1}	$\mathbf{w}^{k+1} + \beta^{k+1} \mathbf{z}^{k+1}$ with $\beta^{k+1} = \frac{\mathbf{w}^{k+1} \cdot \mathbb{W} \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k}$

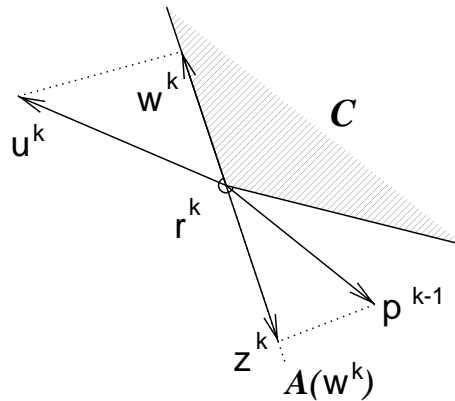


Fig. 4. Representation of the subspace of active constraints $A(\mathbf{w}^k)$.

tests have to confirm or to invalidate this prediction. Such a strategy was proposed by May [26] and Dilintas et al. [16] for frictionless contact problems, but our formulation is more general and synthetic. Indeed Dilintas et al. [16] fixes the set of active constraints to perform the *CPG* algorithm until convergence before updating the active constraints.

A last improvement consists in taking at best advantage of the conjugating process in a same subspace. The directions \mathbf{w}^k and \mathbf{z}^k belong to the same cone but not necessary to the same subspace or edge of the constraint set as shown in Fig. 4. We propose to project \mathbf{p}^{k-1} on the subspace (of active constraints) to which \mathbf{w}^k belongs, after projection of \mathbf{u}^k , noted $A(\mathbf{w}^k)$,

$$\mathbf{z}^k = \text{proj}(\mathbf{p}^{k-1}; A(\mathbf{w}^k)).$$

3.3. Application to contact/friction problems

For the standard constrained quadratic problem (9), the previous algorithms take advantage of the simple form of the constraint set. For the unilateral contact or the bidimensional friction the constraint set is a cartesian product of intervals of \mathbb{R} . Consequently the projections are easy to implement component by component, as specified below

<i>Unilateral contact:</i> $\mathcal{C} = \mathbb{R}_+^{n_c}$	
$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$	$\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$
for $\alpha = 1, n_c$	for $\alpha = 1, n_c$
$r_{\alpha}^{k+1} = \max\left(r_{\alpha}^{k+\frac{1}{2}}, 0\right)$	if $r_{\alpha}^{k+1} = 0$ then $w_{\alpha}^{k+1} = \max(u_{\alpha}^{k+1}, 0)$ else $w_{\alpha}^{k+1} = u_{\alpha}^{k+1}$

<i>2D Friction:</i> $\mathcal{C} = \prod_{\alpha=1}^{n_c} [-s_{\alpha}, s_{\alpha}]$	
$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$	$\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$
for $\alpha = 1, n_c$	for $\alpha = 1, n_c$
if $r_{\alpha}^{k+\frac{1}{2}} > s_{\alpha}$ then $r_{\alpha}^{k+1} = s_{\alpha}$	if $r_{\alpha}^{k+1} = s_{\alpha}$ then $w_{\alpha}^{k+1} = \min(u_{\alpha}^{k+1}, 0)$
if $r_{\alpha}^{k+\frac{1}{2}} < -s_{\alpha}$ then $r_{\alpha}^{k+1} = -s_{\alpha}$	if $r_{\alpha}^{k+1} = -s_{\alpha}$ then $w_{\alpha}^{k+1} = \max(u_{\alpha}^{k+1}, 0)$
else $r_{\alpha}^{k+1} = r_{\alpha}^{k+\frac{1}{2}}$	else $w_{\alpha}^{k+1} = u_{\alpha}^{k+1}$

4. A specific algorithm for frictional contact problems

The coupling of unilateral contact and Coulomb friction is expressed in (8) with a quasi-optimization problem. A classical approach consists in carrying out a fixed point algorithm on the friction threshold.

4.1. A fixed point approach

The problem (8) is approximated by a succession of optimization problem (6) with a fixed threshold s . The resulting algorithm is summarized in the following scheme (cf. Table 3). The solution of the Tresca like frictional contact problem is obtained by a combination of the procedures for *unilateral contact* and *2D friction* detailed in Section 3.3 and illustrated in Fig. 5.

In order to well understand the next diagonalized version, it is interesting to detail this algorithm in Table 4.

4.2. A diagonalized version

This new algorithm consists in suppressing the loop indexed by l according to a diagonalization process. Two principles are used. The projections of the gradients are still performed on the tangent cone of a cartesian product of infinite half cylinders approximating a cartesian product of convex cones because the normality rule satisfied in an optimization problem has to be conserved. Secondly the projection of the iterate is replaced by a correction. This correction has to concentrate three steps of the previous scheme underlined in Table 4: the updating of the threshold, the initialization of the iterate and the projection itself. This leads to the single loop algorithm of Table 5.

The modifications are presented contact by contact (locally) and restricted to the two-dimensional modeling, for which the projected gradient strategy is meaningful according to the remarks in Sections 3.1 and 3.2.

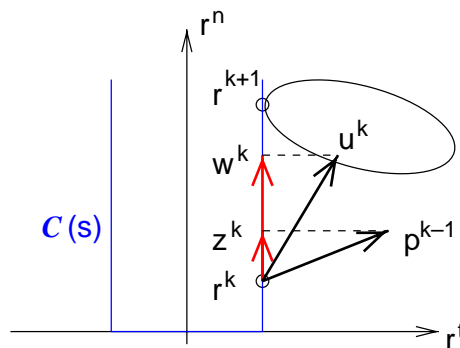
Table 3
Fixed point algorithm
$$\begin{aligned} & \text{Initializing the threshold } \mathbf{s} \\ & \mathbf{s}^0 = (s_x^0, \alpha - 1, n_c) \text{ with } s_x^0 \geq 0 \\ & (\mathbf{r}^0 = \mathbf{0}) \\ & l = l + 1 \\ & \left[\begin{array}{ll} s_x^l & \mu_{x,n}^{l-1}, \alpha - 1, n_c \quad (\text{fixed point}) \quad \mathcal{C}^l \quad \mathcal{C}(s^l) \\ \mathbf{r}^l & \arg \min_{\mathbf{r} \in \mathcal{C}^l} \frac{1}{2} \mathbf{r}^\top \mathbb{W} \mathbf{r} + \mathbf{b}^\top \mathbf{r} \quad (\text{Tresca like frictional contact}) \end{array} \right. \end{aligned}$$


Fig. 5. Projected gradients for a Tresca like problem (the index α is omitted).

Table 4

Detailed fixed point algorithm

Initializing the threshold: $\mathbf{s}^0 = (s_\alpha^0, \alpha = 1, n_c)$ with $s_\alpha^0 \geq 0$ ($\mathbf{r}^0 = \mathbf{0}$)	
$l = l + 1$	
$s_\alpha^l = \mu r_{\alpha,n}^{l-1}, \alpha = 1, n_c$	(updating the threshold) $\mathcal{C}^l = \mathcal{C}(s^l)$
$\mathbf{r}^{l,0} = \mathbf{r}^{l-1,\text{conv}}$	(initializing the iterate)
$\mathbf{u}^0 = \mathbf{b} - \mathbb{W}\mathbf{r}^{l,0}, \mathbf{p}^0 = \mathbf{u}^0$	
$k = k + 1$	
$\mathbf{r}^{l,k+\frac{1}{2}} = \mathbf{r}^{l,k} + \alpha^{k+1} \mathbf{p}^k$ with $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	
$\mathbf{r}^{l,k+1} = \text{proj}_{\mathcal{C}(s^l)}(\mathbf{r}^{l,k+\frac{1}{2}})$	(projecting the iterate)
$\mathbf{u}^{k+1} = \mathbf{b} - \mathbb{W}\mathbf{r}^{l,k+1}$	(pre conditioner: $\mathbf{u}^{k+1} = \mathbb{P}(\mathbf{b} - \mathbb{W}\mathbf{r}^{l,k+1})$)
if convergence then $\mathbf{r}^l = \mathbf{r}^{l,k+1}$ and exit	
$\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}^l}(\mathbf{r}^{l,k+1}))$	
$\mathbf{z}^{k+1} = \text{proj}(\mathbf{p}^k; A(\mathbf{w}^{k+1}))$	
$\mathbf{p}^{k+1} = \mathbf{w}^{k+1} + \beta^{k+1} \mathbf{z}^{k+1}$ with $\beta^{k+1} = \frac{\mathbf{w}^{k+1} \cdot \mathbb{W}\mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	

Table 5

Conjugate projected gradient algorithm (CPG), frictional contact version

Initialization of $\mathbf{u}^0 = \mathbf{b} - \mathbb{W}\mathbf{r}^0, \mathbf{w}^0 = \mathbf{u}^0, \mathbf{p}^0 = \mathbf{u}^0$	
$k = k + 1$	
$\mathbf{r}^{k+\frac{1}{2}} = \mathbf{r}^k + \alpha^{k+1} \mathbf{p}^k$ with $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	
Correction of the iterate: $\mathbf{r}^{k+\frac{1}{2}} \rightarrow \mathbf{r}^{k+1}$	
$\mathbf{u}^{k+1} = \mathbf{b} - \mathbb{W}\mathbf{r}^{k+1}$	(preconditioner: $\mathbf{u}^{k+1} = \mathbb{P}(\mathbf{b} - \mathbb{W}\mathbf{r}^{k+1})$)
Convergence criterion: if true then $\mathbf{r}^{\text{conv}} = \mathbf{r}^{k+1}$	
Projection of gradients: $\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}^{k+1}}(\mathbf{r}^{k+1}))$ $\mathbf{z}^{k+1} = \text{proj}(\mathbf{p}^k; A(\mathbf{w}^{k+1}))$	
$\mathbf{p}^{k+1} = \mathbf{w}^{k+1} + \beta^{k+1} \mathbf{z}^{k+1}$ with $\beta^{k+1} = \frac{\mathbf{w}^{k+1} \cdot \mathbb{W}\mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	

4.2.1. Correction of the iterate

At first the updating of the threshold differs from that of the double-loop algorithm because $r_{\alpha,n}^{k+\frac{1}{2}}$ is not necessary non-negative,

$$s_\alpha^{k+1} = \max(0, \mu r_{\alpha,n}^{k+\frac{1}{2}}), \quad \mathcal{C}_\alpha^{k+1} = \mathbb{R}^+ \times [-s_\alpha^{k+1}, s_\alpha^{k+1}].$$

The projection of the iterate $\mathbf{r}_\alpha^{k+\frac{1}{2}}$ is then carried out on this new local convex set if the iterate does not belong to it. But when the iterate is inside \mathcal{C}_α^{k+1} , the situation is quite complicated. A strict application of a projection procedure would lead to confirm $\mathbf{r}_\alpha^{k+\frac{1}{2}}$ as \mathbf{r}_α^{k+1} and so consider a “stick” status for this contact, which is not optimal if the previous status was “slip”. Indeed, assume that the solution status is “slip” (backward or forward), the iterate may oscillate between “slip” and “stick” because of the conical form of the Coulomb criterion. For taking advantage of the conjugating process it is more convenient to keep the descent directions in the same subspace corresponding to the “slip” status. Consequently if the current local descent direction \mathbf{p}_α^k has no tangential component, the local contact status, which was then “slip” previously, has to be maintained even if the iterate is inside the friction cone and inside its approximated cylindrical set \mathcal{C}_α^{k+1} ; such a situation is shown with the case (b) in Fig. 6. In this way we can summarize in the correction the three different steps underlined in the double-loop algorithm. The detailed correction

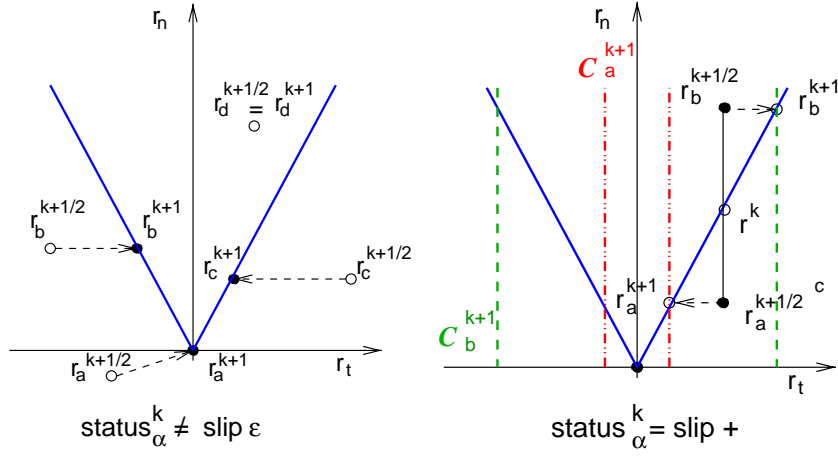


Fig. 6. Iterate correction (some situations according to the previous status).

Table 6
Correction of the iterate

if $r_{\alpha,n}^{k+1/2} < 0$	then $r_\alpha^{k+1} = 0$	(status $_\alpha^{k+1}$ gap)
else	$r_{\alpha,n}^{k+1} = r_{\alpha,n}^{k+1/2} \in \text{sign}(r_{\alpha,t}^{k+1/2})$	
	$s_\alpha^{k+1} = \max(0, \mu r_{\alpha,n}^{k+1/2})$	
if $ r_{\alpha,t}^{k+1/2} > s_\alpha^{k+1}$	then $r_{\alpha,t}^{k+1} = \varepsilon s_\alpha^{k+1}$	(status $_\alpha^{k+1}$ slipe)
else		
if $p_{\alpha,t}^k = 0$	then $r_{\alpha,t}^{k+1} = \varepsilon s_\alpha^{k+1}$	(status $_\alpha^{k+1}$ slipe)
else	$r_{\alpha,t}^{k+1} = r_{\alpha,t}^{k+1/2}$	(status $_\alpha^{k+1}$ stick)

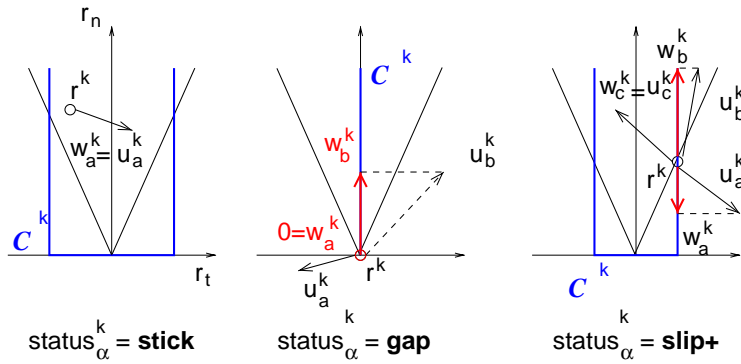


Fig. 7. Gradient projection (some situations according to the previous status).

Table 7
Projection of the gradients

if status _{α} ^k gap	then	if $u_{\alpha,n}^k \leq 0$ then w_{α}^k	0
		else $w_{\alpha,n}^k$	$u_{\alpha,n}^k$ and $w_{\alpha,t}^k$ 0
if status _{α} ^k stick	then	w_{α}^k	u_{α}^k
if status _{α} ^k slip ⁺	then	$w_{\alpha,n}^k$	$u_{\alpha,n}^k$ and $w_{\alpha,t}^k$ $\min(0, u_{\alpha,t}^k)$
if status _{α} ^k slip ⁻	then	$w_{\alpha,n}^k$	$u_{\alpha,n}^k$ and $w_{\alpha,t}^k$ $\max(0, u_{\alpha,t}^k)$

procedure (cf. Table 6) uses the notion of (local) contact status which is so useful for understanding the next projection of the gradients. The status provides the pertinent information on the iterate at which the tangent cone $T_{\mathcal{C}}$ has to be evaluated (Fig. 7).

4.2.2. Projection of the gradients

Since the approximating convex set is well defined the projection of the gradient on the tangent cone is a simple combination of the procedure described above for the unilateral contact and the 2D friction. But the more convenient way consists in using the previous contact status first stored in memory as specified in Table 7.

We can extend this approach to the three-dimensional modeling, but the following numerical tests do not account for this extension and the projections on a non-polyhedral constraint set allow different variants that we have to try out.

5. Pre-conditioning and convergence criteria

The algorithms presented above may be used for all contact or frictional contact problems, structural or granular type ones. But their efficiency depends on the features of the problem and on the implementation of the methods. Some remarks about implementation are necessary to explain the following choices for pre-conditioners and convergence criteria.

5.1. Some remarks about implementation

In the CPG algorithm two matrix vector products are performed ($\mathbb{W}\mathbf{p}^k, \mathbb{W}\mathbf{r}^{k+1}$) instead of a single for a CG method applied to unconstrained quadratic problems ($\mathbb{W}\mathbf{p}^k$). An alternative consists in carrying out $\mathbb{W}\mathbf{r}^{k+1}$ only if at least one local projection has been activated in the previous correction step ($\mathbf{r}^{k+\frac{1}{2}} \rightarrow \mathbf{r}^{k+1}$); the gradient \mathbf{u}^{k+1} is then updated by the classical formula using $\mathbb{W}\mathbf{p}^k$. This strategy may be efficient for frictionless contact [16] and frictional contact with a Tresca law, but it is not relevant for Coulomb frictional contact because the correction of $\mathbf{r}^{k+\frac{1}{2}}$ is almost always activated as soon as sliding statuses are predicted. Moreover in granular type problems the number of contacts is large enough to produce many status changes.

All instructions in the CPG scheme may be treated by parallel computing, either because they use vector algebra or because the projections or corrections are performed component-wise. Then a pre-conditioner has to be the simplest one with possible parallel implementation to avoid to introduce sequential procedures.

The storage of the \mathbb{W} matrix depends on the type of problem. For granular media, this matrix is not gathered like a finite element matrix. In the original *LMGC 90* software [17] using essentially a Gauss Seidel like solver (*NSCD* method), the \mathbb{W} matrix was not initially formed; the matrix vector product was only

performed transferring information from the local level (contacts) to the global one (bodies) and vice versa. In [37] a numerical comparison, including parallel computing, is made between this original version and a new version with a stored \mathbb{W} matrix. Moreover the new version is chosen to identify the convergence criteria between the CPG algorithm and the NSCD method.

5.2. Simple pre-conditioners

What does a pre-conditioner mean for a constrained quadratic optimization solver? Recall that pre-conditioning a CG algorithm applied to an unconstrained quadratic problem consists only in multiplying the current gradient \mathbf{u}^{k+1} by a pre-conditioning matrix \mathbb{P} . This simple additional instruction amounts to replace the initial ill-conditioned problem by a better conditioned one, consisting to minimize $\frac{1}{2} \mathbf{s} \cdot \overline{\mathbb{W}} \mathbf{s} - \mathbf{b} \cdot \mathbf{s}$ with the following change of variables:

$$\mathbf{s} = \mathbb{P}^{\frac{1}{2}} \mathbf{s} \quad \mathbf{b} = \mathbb{P}^{-\frac{1}{2}} \mathbf{b} \quad \text{and} \quad \overline{\mathbb{W}} = \mathbb{P}^{\frac{1}{2}} \mathbb{W} \mathbb{P}^{\frac{1}{2}}.$$

To recover the pre-conditioned scheme we have only to identify each instruction of the CG algorithm applied to the pre-conditioned problem using the previous definitions. For a constrained quadratic problem the pre-conditioned problem is

$$\inf_{\mathbf{s} \in \mathcal{C}} \frac{1}{2} \mathbf{s} \cdot \overline{\mathbb{W}} \mathbf{s} - \mathbf{b} \cdot \mathbf{s}, \quad (13)$$

with moreover

$$\mathcal{C} = \mathbb{P}^{-\frac{1}{2}} \mathcal{C}.$$

The projection in the CPG scheme (Table 2) applied to the pre-conditioned problem (13), dealing with the new over-lined variables, have to be defined in such away we can identify the Euclidean projections in the algorithm expressed with the initial variables. In other words what is $\text{proj}(\cdot; \mathcal{C})$ in $\mathbf{a} = \text{proj}(\mathbf{b}, \mathcal{C})$ in such a way that $\mathbf{a} = \text{proj}(\mathbf{b}; \mathcal{C})$? We can easily prove that $\text{proj}(\cdot; \mathcal{C})$ has to be defined with the norm associated with \mathbb{P} , $\|\mathbf{x}\|_{\mathbb{P}}^2 = \mathbf{x} \cdot \mathbb{P} \mathbf{x}$,

$$\mathbf{a} = \text{proj}(\mathbf{b}; \mathcal{C}) \iff \|\mathbf{b} - \mathbf{a}\|_{\mathbb{P}}^2 \leq \|\mathbf{b} - \mathbf{x}\|_{\mathbb{P}}^2, \quad \forall \mathbf{x} \in \mathcal{C}.$$

The projection on the tangent cone is defined in the same manner. Such definitions are not practically carried out with the new variables, but they justify to carry out the pre-conditioned conjugate projected gradient (PCPG) algorithm using the initial variables and introducing a simple matrix vector product $\mathbb{P} \mathbf{u}^{k+1}$ before projecting the gradients (cf. Tables 4 and 5). With the intention of preserving parallel computing for the whole solver, we chose to carry out the simplest pre-conditioners. Moreover the efficiency of pre-conditioning procedures are not proved efficient for constrained quadratic optimization problems with inequality constraints; sophisticated pre-conditioners are then excluded in a first approach.

5.2.1. Diagonal pre-conditioners

If the \mathbb{W} matrix is split into its diagonal part and its non-diagonal one, $\mathbb{W} = \mathbb{D} - \mathbb{E}$, the pre-conditioner is the inverse of the diagonal part, $\mathbb{P} = \mathbb{D}^{-1}$. This procedure is not expansive because the matrix vector product is then very simple. This first strategy may be often sufficient to improve the convergence, especially when a polydisperse sample of a granular medium is considered: the sample then contains big and small grains leading to a large difference between the (diagonal) terms of the mass matrix and then of the \mathbb{W} matrix. A variant consists in considering a block splitting of the \mathbb{W} matrix,

where each diagonal block corresponds to one contact. For granular type problems this distinction is only relevant for polygonal or polyhedral grains. Indeed, for spherical grains, the block-diagonal part of \mathbb{W} is still diagonal.

5.2.2. Polynomial pre-conditioners

Among the more sophisticated pre-conditioners, the polynomial pre-conditioners remain good candidates because the preparation step is not too expansive. The \mathbb{P} matrix conserves the same structure than \mathbb{W} ; for example the pre-conditioner of level one may be built while \mathbb{W} is built itself, $\mathbb{P} = \mathbb{D}^{-1} + \mathbb{D}^{-1}\mathbb{E}\mathbb{D}^{-1}$. But this pre-conditioner needs a third matrix vector product at each iteration; it may be too expansive compared to the gain that we can hope in terms of number of iterations. Some tests concerning small examples with some disks prove that the condition number is only divided by two from the diagonal pre-conditioner (polynomial of level zero) to the level one pre-conditioner.

5.3. Convergence criteria for non-smooth systems

The question of accuracy and the choice of the convergence criteria are not easy tasks. Whatever the considered problem and overall in the context of large dense collection of rigid contacting bodies. Indeed, the exact solution is usually unknown and the only information which might be exhibited is some distance to it. Upper and lower bounds of such a distance may be obtained using mathematical convergence properties of the algorithm, which means that a unique solution has been proved to exist and that the convergence of the algorithm is established. Except in some quite peculiar cases, these circumstances are not realized. At first the non-uniqueness is not an exception but the rule. Indeed it may be caused by two features. The first one is of a geometrical nature. The network of the forces between particles in a frictionless dense granulate may be viewed as the same network in a truss composed with bars linking the center of particles. Such a system is modeled in the \mathbb{W} matrix, it may be hyper-static and then the \mathbb{W} matrix becomes singular (positive semi-definite). Note that we check the nullity of the term $\mathbf{p} \cdot \mathbb{W}\mathbf{p}$ which may vanish even if \mathbf{p} is different from $\mathbf{0}$. This precaution, theoretically necessary is practically never activated. On the other hand, the friction laws induces also a multiplicity of the solution of the contact impulses. The (numerical) solution depends on the history of loading and numerically on the choice of the initialization of the algorithm; the previous contact forces are naturally considered [29].

In the simulation of granular material with the *NSCD* approach using a *NLGS* algorithm, Jean introduces an energetic criterion applied to a Jacobi correction to the current iterate [10]. To obtain an equivalent test for the *CPG*, useful for the forthcoming comparison, we consider the relative velocity variation $\Delta\mathbf{v}$ issued from two successive reaction iterates and related to the descent direction

$$\Delta\mathbf{v} = \mathbf{v}^{k+1} - \mathbf{v}^k = \mathbb{W}(\mathbf{r}^{k+1} - \mathbf{r}^k) \simeq \mathbb{W}(\mathbf{r}^{k+\frac{1}{2}} - \mathbf{r}^k) = \alpha^k \mathbb{W}(\mathbf{p}^{k+1}).$$

We introduce a reference energy

$$E(\mathbf{r}_\alpha^k) = \sum_\alpha \mathbf{r}_\alpha^k \cdot \mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^k,$$

and two convergence criteria to be both satisfied,

$$\begin{cases} e_Q = \frac{\alpha}{E(\mathbf{r}_\alpha^k)} \sqrt{N_{\text{active}}} \sum_\alpha (\mathbf{p}_\alpha^k \cdot \mathbf{p}_\alpha^k) \times (\mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^k \cdot \mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^k), \\ e_m = \frac{\alpha}{E(\mathbf{r}_\alpha^k)} \left| \sum_\alpha (\mathbf{p}_\alpha^k \cdot \mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^k) \right|, \end{cases}$$

where e_m is a mean value over all contacts and e_Q a quadratic version more drastic in numerous cases, N_{active} counts the active contacts (stick or slip status).

6. Numerical tests in granular mechanics

To conduct a complete study of the gradient type algorithms applied to the simulation of granular media, we first investigate their behavior on a single time step. In a second section different whole processes are considered to take into account the variety of granular problems [6,23,34,35]. In these sections the *CPG* algorithm is essentially compared with the *NLGS* method generally associated with the *NSCD* approach [20,30]. All simulations are performed with the *LMGC90* software [17] dedicated to multi-contact problems, very useful for granular materials.

6.1. On a single step

One step is isolated at the end of a depositing process of particles in a box in the field of gravity. This amounts to determine the distribution of the contact forces in a equilibrium state of a granular packing. As the number of contacts is an important parameter of the study several numerical samples are created in the same box ($1 \text{ m} \times 1 \text{ m}$) with different sizes of circular particles. Thus the capacity of the box goes from 1000 to 33,000 disks. For a frictionless problem we verify first that all methods are descent like for the energy to minimize. Fig. 8 gives the evolution of the objective function $E^k = \frac{1}{2} \mathbf{r}^k \cdot \mathbb{W} \mathbf{r}^k - \mathbf{r}^k \cdot \mathbf{b}$ during the iterations for the following methods: non-linear Gauss Seidel (*NLGS*), conjugate projected gradient (*CPG*), pre-conditioned conjugate projected gradient (*PCPG*). The better behavior of conjugate gradient methods is not surprising but has to be confirmed for frictional contact problems with the adaptations proposed above.

The Coulomb frictional contact situation requires the convergence criteria introduced in Section 5.3 to carry out the comparisons. For simplicity reason the e_m criterion is omitted; the e_Q one is generally more difficult to satisfy and thus more reliable. A fourth algorithm is considered: the Projected Gradient without conjugating (*PG*). The sample for Figs. 9 and 10 comprises 15,000 contacts. The convergence is reached for $e_Q < \epsilon = 1.66 \times 10^{-6}$. The conjugating process reveals to be necessary for a good behavior of gradient method in comparison with the Gauss Seidel one. The *NLGS* solver provides a smooth convergence since the *CPG* methods converge faster but with large perturbations and a final drop. The large oscillations with

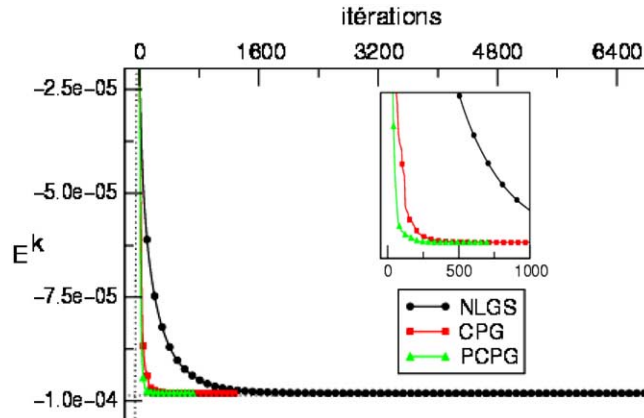


Fig. 8. Decreasing of energy E^k during iterations ($n_c = 28,000$).

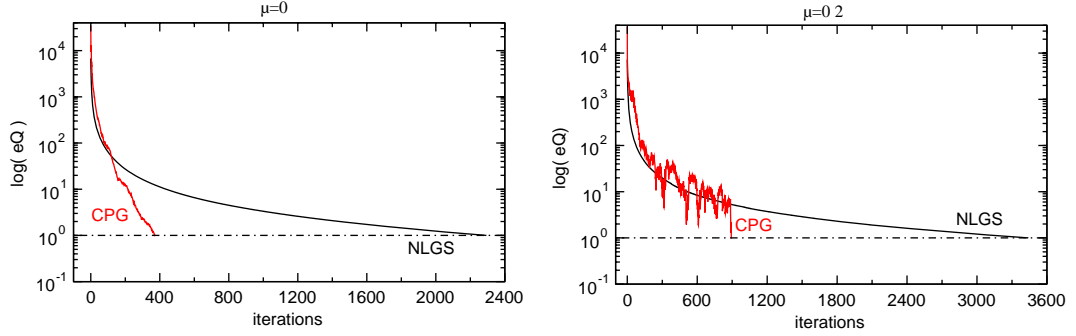


Fig. 9. Convergence behavior of CPG/NLGS ($n_c = 15,000$) for two friction coefficients.

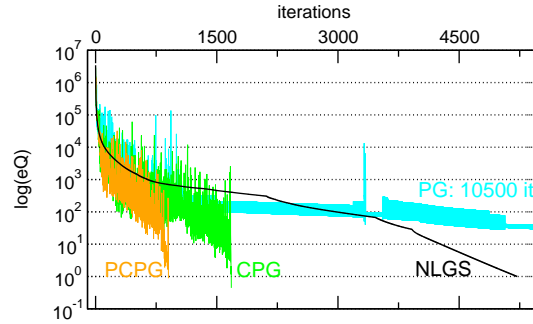


Fig. 10. Convergence behavior of the fourth algorithms ($n_c = 15,000$) for $\mu = 0.4$.

the gradient methods can be attributed to the corrections of the iterate specific to the Coulomb frictional contact law as proved by the smoother curves obtained with a zero friction coefficient (Fig. 9).

We verify in Fig. 10 the increase of the oscillations with a stronger friction coefficient equal to 0.4. In this example the pre-conditioner improves significantly the algorithm. Two parameters are relevant to appreciate the performance of the gradient methods: the size of the system characterized by the number of contacts n_c and the required accuracy ϵ .

We present in Fig. 11 the evolution of the gain of iterations from the NLGS to the CPG or PCPG ($\delta = (\text{number of NLGS iterations})/(\text{number of (P)CPG iterations})$) according to the number of contacts and for the three required accuracies. The gain is at least equal to 2.98 ($n_c = 15,249$ and $\epsilon = 1.66 \times 10^{-5}$) and may reach 9.3 for CPG ($n_c = 15,249$ and $\epsilon = 1.66 \times 10^{-4}$) and even 9.47 for PCPG ($n_c = 28,014$ and $\epsilon = 1.66 \times 10^{-6}$).

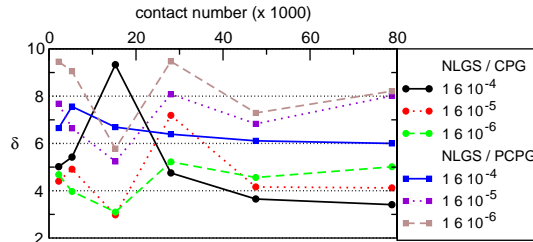


Fig. 11. Gain of iterations, δ , of the CPG and PCPG in regard of the NLGS.

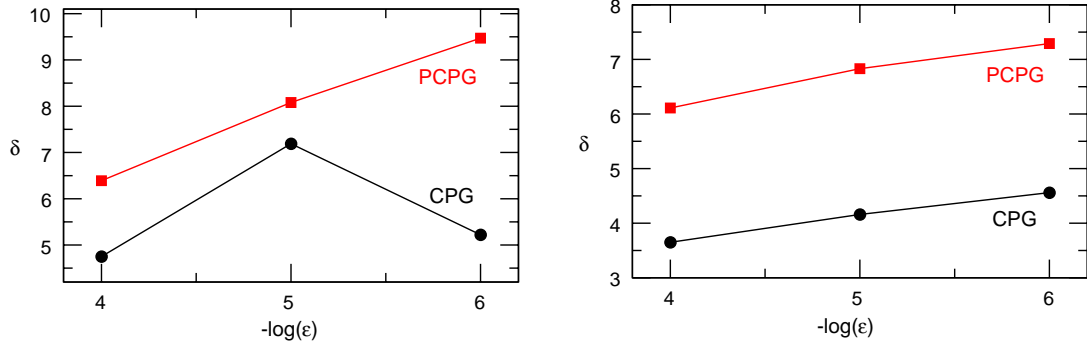


Fig. 12. Gain for $n_c = 28,014$ (left) and $n_c = 47,531$ (right).

Due to the erratic convergence behavior underlined in Fig. 10, it is not simple to conclude below 40,000 contacts. Beyond this value the gain seems to stabilize around 4 for the *CPG* algorithm and 7 for the pre-conditioned one. Finally the gain is all the more important since the required accuracy is high as proved in Fig. 12 where two sizes of samples are extracted of the previous figure. But such a study has to be confirmed in considering a whole evolution process.

6.2. On a whole process

To cover numerous cases, we study dynamic and quasi-static processes. During this kind of simulations, we usually chose the solution of the previous time step to initialize the algorithm. Due to the multiplicity of solutions in granular materials and the numerical error, the solution at the beginning of the second time step will be generally different if we use different methods to solve our problem. In consequence, we can only perform a global analysis of our process. To avoid or limit this difficulty we could initialize with zero at each time step. But this procedure penalizes all methods without favoring an algorithm in comparison with an other. To compare our results in terms of quality, some macroscopic quantities like compactness, anisotropy, mean velocity are introduced.

6.2.1. Rotating drum

The rotating drum is a dynamic process at least close to the free surface where avalanches and even a regular flow occur (cf. Fig. 13). Our model is similar to the experimental one of Bonamy et al. [6]: 7200 disks (mean diameter equal to 3 mm) are packed in a drum with a diameter of 450 mm.

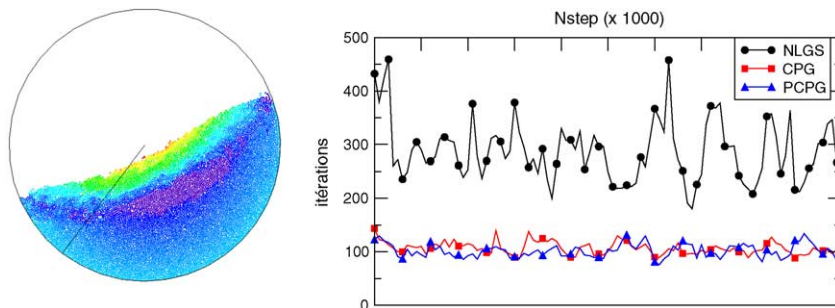


Fig. 13. Numerical rotating drum simulation: on the left we observe the magnitude of the mean velocity field between two time steps for each particle. On the right side, the evolution of the iterations for the different simulations.

Table 8

Performance of the *CPG* algorithm with respect to the *NLGS* method for different processes: n_p represents the number of particles and ϵ the accuracy

Process	n_p	ϵ	$\langle it \rangle$ ratio	<i>CPU</i> ratio
Rotating drum	7200 disks	1.6×10^{-4}	2.75	0.95
Biaxial test	1024 disks	1.6×10^{-4}	3.26	1.69
	1024 disks	1.6×10^{-6}	7.09	3.38
	9216 disks	1.6×10^{-4}	2.17	0.81
	9216 disks	1.6×10^{-5}	3.02	1.49
Shear test	1024 disks	1.6×10^{-4}	3.32	2.23
	9216 disks	1.6×10^{-4}	3.66	1.70
Free surface	1000 disks	1.6×10^{-4}	2.99	1.56
Compaction	10,000 disks	1.6×10^{-4}	1.31	0.82

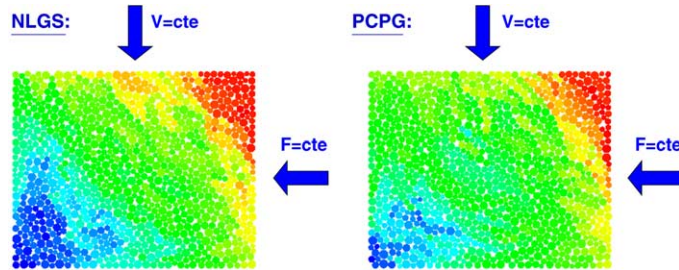


Fig. 14. Displacement field of particles between two time steps at the end of the simulation (1024 disks): contrary to the local level where there is many differences (position of particles), at the global level we distinguish identical global displacement.

In addition to a frictional contact law, a normal restitution shock law (coefficient equal to 0.92) is postulated and accentuates the dynamic effects. The drum is submitted to an angular velocity equal to 3 rpm (round per minute). Over the whole process (Fig. 13) the mean number of iterations per step is equal to 290 for *NLGS* and 102 for the *PCPG*, that is to say a gain of 2.75. Moreover the variation of the number of iterations during the process is weaker for the *CPG* and *PCPG* algorithms than for the *NLGS* one. Here the pre-conditioner is not useful: the iterations number does not decrease. Since the cost of a *CPG* iteration is higher than that of one *NLGS* iteration, the gain in *CPU* time is null and even negative (ratio equal to 0.95 in Table 8).

6.2.2. Biaxial tests

It consists in imposing a global biaxial deformation to a square box: a constant pressure is applied on the right boundary since a constant velocity is imposed to the upper side of the sample (cf. Fig. 14). The process is carried out until 10% deformation. Fig. 14 shows a first interesting difference between simulations: the displacement increment field is more homogeneous with the conjugate gradient methods; the smallest and largest values are very concentrated in two corners.

Such a difference may be explained by the global transmission of the information over the whole domain assured in the gradient methods by the matrix-vector products at each iteration since the *NLGS* algorithm spreads slowly the information as a wave. Indeed the biaxial tests are performed with four rigid bodies for the lateral, upper and lower walls. Consequently the bandwidth of the \mathbb{W} matrix is large due to the coupling

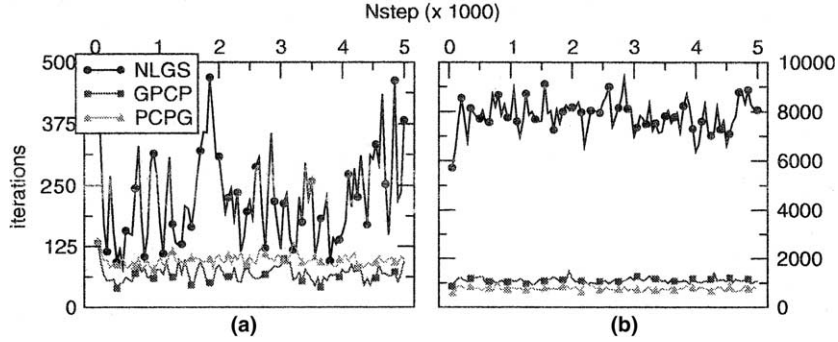


Fig. 15. Evolution of the number of iterations obtained with the three algorithms (biaxial test with 1024 disks) with an accuracy of 1.66×10^{-4} (a) and 1.66×10^{-6} (b).

of all the disks in contact with the same wall. In this case an investigation of the stress transmissions through the domain during the iterations shows the propagation from the walls to the center is much faster with CPG than NLGS. The local differences do not modify the macroscopic behavior of the sample which may be analyzed with some relevant mean quantities [34,41]. A first coarse sample with 1024 disks is performed with two required accuracies. The evolution of the number of iterations are copied out in Fig. 15 and the mean gains in terms of iterations and *CPU* time are summarized in Table 8. A finer sample with 9216 disks is considered to provide similar average indications in Table 8. We obtain better gains with a higher accuracy in such a way that for the fine sample it is necessary to require a strict accuracy to get a positive gain in *CPU* time (1.49).

6.2.3. Shear test

In a simple shear test an angular velocity is imposed to the lateral sides of the square since a constant pressure is maintained on the upper side. Two coarse and fine samples are tested and the results are transferred in Fig. 16 and Table 8.

In this case even if the gain in iterations increases slowly from 3.32 to 3.66 when a finer sample is considered, the gain in *CPU* time decreases from 2.23 to 1.70. This proves that the efficiency in *CPU* time is not easy to extrapolate from an analysis in terms of iterations only. For large problems the memory management in a computer may modify strongly the prediction. To illustrate the global behavior of the samples, we present in Fig. 17 the evolution of macroscopic parameters obtained by averaging: the compactness is the

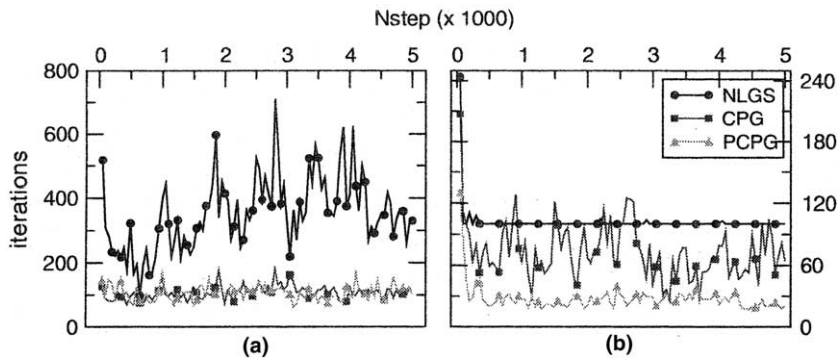


Fig. 16. Evolution of the number of iterations in a simple shear test with 1024 (a) and 9216 (b) disks (required accuracy: 1.6×10^{-4}).

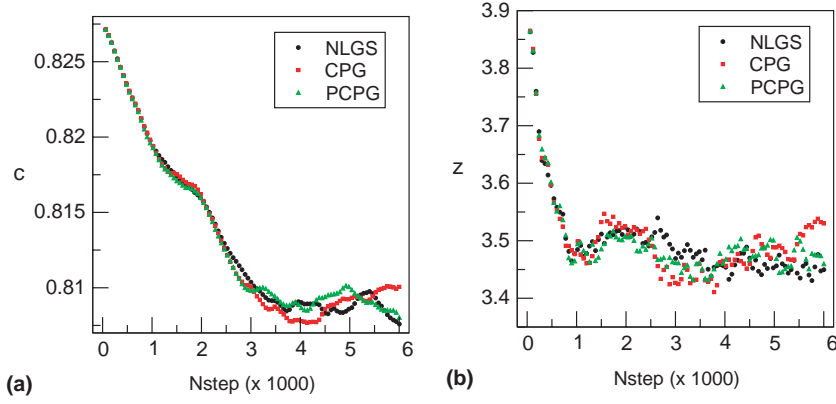


Fig. 17. Evolution of characteristic macroscopic properties (compactness (left) and coordination number (right)) for the different algorithms in a simple shear test.

volume of the particles divided by the total volume; the mean coordination number is the mean number of contacts per particle. The evolution of such quantities is not really perturbed by the use of the different algorithms.

6.2.4. Free surface compaction tests

This example differs strongly from the previous ones because the solicitations are not constant in time. After depositing particles in a box, the sample is compacted in moving one of the lateral walls with a non-monotone speed. Such conditions lead to a non-constant evolution of the number of iterations as shown in Fig. 18 (contrary to the previous examples). For the smaller example (1000 disks) we recover similar results with a gain in iterations around 3 and a gain in *CPU* time around 1.5 (cf. Table 8).

For the finer example (10,000 disks), Fig. 18(a) shows apparently a better behavior of the *NLGS* algorithm. In fact the *NLGS* method converges too fast at the beginning of the process and this lead to an accumulation of the interpenetration of the particles as proved in Fig. 18(b). The *NLGS* algorithm requires a minimum number of iterations to diffuse information through the whole domain. To avoid such a situation a minimum of iterations is imposed (20 for the standard version). But this threshold is not sufficient: to recover an identical level of interpenetration, 200 iterations at least are required. No rule is avail-

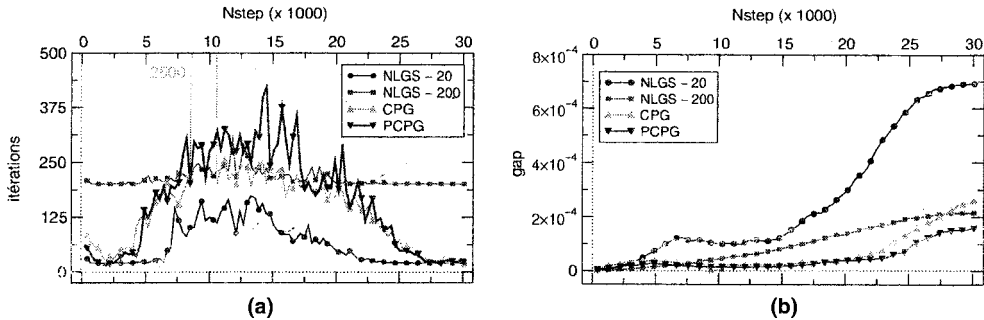


Fig. 18. For a 10,000 particles free surface compaction: (a) evolution of iterations number (left); (b) evolution of the mean gap (right).

able to choose such a parameter. Then the gradient methods become again more efficient in terms of mean iterations number (1.31), but the *CPU* time gain remains less than one.

6.2.5. Summary

In conclusion to this study the gradient methods require in general three times less iterations than the *NLGS* solver (for a reasonable required accuracy). The *CPU* time gain is difficult to evaluate but is around 1.5. The conjugate gradients algorithms seem to be more robust to solve dynamic or quasi-static problems; the user has not to impose an extra parameter to converge toward an acceptable solution. If a high accuracy is required the gains may jump to high values.

6.3. Multi-threading

Gradient algorithms are intrinsically parallel due to the fact that they are essentially composed of products between vectors and matrices. The implementation of a parallel version is thus not a difficulty in itself. Like for the *NLGS* algorithm [38], we opt for a multi-threading on shared memory computer to avoid the message passing. The implementation is so easily performed via OpenMP directives [19]. Although our algorithm is intrinsically parallel, using parallel treatment modifies slightly the solution on a whole process. On one time step sequential and parallel versions give the same solution. But during a process we accumulate numerical errors which can have more visible consequences in a granular material, and this for the reasons evoked previously. Recall that the infinity of solutions makes the simulations very sensitive to the numerical error accumulations; these errors may be due to the parallelization but also to the implementation procedure as proved in [37] where two implementations of the *NLGS* algorithm are performed. So during the process, the evolution of the iterations number with sequential and parallel treatment diverges after some time steps. To illustrate efficiency of parallel computing, we introduce the relative speed-up, S_p , defined by

$$S_p = \frac{T_{\text{seq}}}{T_{\text{par}}} \frac{I_p}{I_{\text{seq}}}. \quad (14)$$

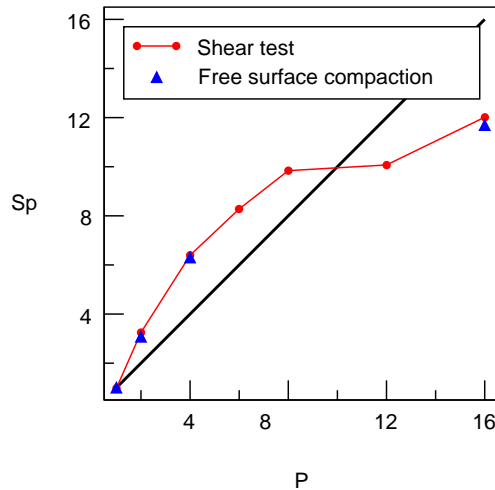


Fig. 19. Evolution of the relative speed up. Simulations have been performed on a SGI Origin 3800 using processors R14000/500 MHz.

P is the number of processors, T_P the CPU time used by a P -parallel simulation and It_P the mean number of iterations of a P -parallel simulation. Fig. 19 represents the evolution of S_P for a simple shear test and free surface compaction.

Evolution of speed-up is quite curious. This phenomenon is due to the increase of size memory usable. For a sequential computation only 500 Mo are available. This capacity is multiplied by P for a P -parallel simulation. So to the efficiency of parallel computing, we add the benefit of a large memory stack, very useful for large simulation. In consequence, superlinear speed-up can be explained by this phenomenon.

7. Conclusion

The proposed numerical scheme reveals to be more efficient than the non-linear Gauss Seidel algorithm but the gain is not always important. To impose it we need to pursue the investigations about the convergence behavior. For instance, even if the conjugating process seems to be essential (cf. Fig. 10), we observe a large number of changes of contact status at each iteration which might cancel the effect of the conjugating. Some specific features of granular problems may be so exploited as bimodal character of the network of contact forces in granular packing underlining scale effects [34]. The extension to the three dimensional modeling with isotropic friction is an open question because the convex constraint set is no more polyhedral. The concept of active constraints related to the projection of the gradient and mentioned in Section 3.2 is no more relevant. A way may consist in combining a gradient method with other ones. For instance we can combine the gradient algorithm for the normal components with the Gauss Seidel method for correcting the tangential components.

Acknowledgment

This work is supported by the Centre d'Informatique National de l'Enseignement Supérieur (CINES) project mgc2547.

References

- [1] K. Ach, P. Alart, Numerical simulation of a multi jointed structure, Philos. Trans. Roy. Soc. Lond. A 359 (2001) 2557–2573.
- [2] P. Alart, Critère d'injectivité et de surjectivité pour certaines applications de \mathbb{R}^n dans lui-même; application à la mécanique du contact, Math. Mod. Num. Anal. 27 (2) (1993) 203–222.
- [3] P. Alart, Méthode de Newton généralisée en mécanique du contact, J. Math. Pures Appl. 76 (1997) 83–108.
- [4] P. Alart, M. Barboteu, M. Renouf, Parallel computational strategies for multi contact problems: applications to cellular and granular media, Int. J. Mult. Scales Comput. Engrg. 1 (4) (2003) 419–430.
- [5] P. Alart, F. Lebon, F. Quittau, K. Rey, Frictional contact problem in elastostatics: revisiting the uniqueness condition, in: M. Jean, M. Raous, J. J. Moreau (Eds.), Contact Mechanics, Plenum Press, New York, 1995, pp. 63–70, ISBN 0 306 45065 8.
- [6] D. Bonanry, F. Daviaud, L. Laurent, Experimental study of granular surface flows via a fast camera: a continuous description, Phys. Fluids 14 (5) (2002) 1666–1673.
- [7] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, C.A. Sugastizabal, Numerical Optimization Theoretical and Practical Aspect, Springer, 2003.
- [8] P.A. Boucard, L. Champaney, A suitable computational strategy for the parametric analysis of problems with multiple contact, Int. J. Numer. Methods Engrg. 57 (2003) 1259–1281.
- [9] B. Brogliato, A.A. ten Dam, L. Paoli, F. Génot, M. Abadie, Numerical simulation of finite dimensional multibody nonsmooth mechanical systems, Appl. Mech. Rev. 55 (2) (2002) 107–149.
- [10] B. Cambou, M. Jean, Micromécanique des Matériaux Granulaires, Hermès Science, 2001.
- [11] D. Cambou, Ph. Dubujet, Difficulties and limitation of statistical homogenization in granular materials, in: Vermeer et al. (Eds.), Continuous and Discontinuous Modeling of Cohesive Frictional Materials, vol. 568, Springer, 2001, pp. 205–214.

- [12] Z. H. Cao, On the convergence of iterative methods for solving singular linear systems, *J. Comput. Appl. Math.* 145 (2002) 1–9.
- [13] P. Chabrand, F. Dubois, M. Raous, Various numerical methods for solving unilateral contact problems with friction, *Math. Comput. Modell.* 28 (4–8) (1999) 97–108.
- [14] P.W. Christensen, A semi smooth Newton method for elasto plastic contact problems, *Int. J. Solids Struct.* 39 (2002) 2323–2341.
- [15] M. Cocu, Existence of solutions of Signorini problems with friction, *Int. J. Engrg. Sci.* 22 (5) (1984) 567–575.
- [16] G. Dilintas, P. Laurent Gengoux, D. Trystram, A conjugate projected gradient method with preconditioning for unilateral contact problems, *Comput. Struct.* 29 (4) (1988) 675–680.
- [17] F. Dubois, M. Jean, LMGC90 une plateforme de développement dédiée à la modélisation des problèmes d'interaction, in: *Actes du sixième colloque national en calcul des structures*, vol. 1, CSMA AFM LMS, 2003, pp. 111–118.
- [18] G. Duvaut, Loi de frottement non locale, *J. Méc. Th. et Appl. (numéro spécial)* (1982) 73–78.
- [19] E. Gondet, P. F. Lavalée, Coûts OpenMP. IDRIS (September) (2000).
- [20] M. Jean, The non smooth contact dynamics method, *Comput. Methods Appl. Mech. Engrg.* 177 (1999) 235–257.
- [21] F. Jourdan, P. Alart, M. Jean, A Gauss–Seidel like algorithm to solve frictional contact problem, *Comput. Methods Appl. Mech. Engrg.* 155 (1998) 31–47.
- [22] Y. Kishino, in: *Mechanics of Granular Materials: An Introduction*, Balkema, 2000, pp. 147–159, Chapter 3.
- [23] Y. Kishino, H. Akaizawa, K. Kaneko, On the plastic flow of granular materials, in: Y. Kishino (Ed.), *Powder and Grains*, 2001, pp. 199–202.
- [24] P. Ladevèze, A. Nouy, O. Loiseau, A multiscale computational approach for contact problems, *Comput. Methods Appl. Mech. Engrg.* 43 (2002) 4869–4891.
- [25] T.A. Laursen, J.C. Simo, Algorithmic symmetrization of Coulomb frictional problems using augmented Lagrangians, *Comput. Methods Appl. Mech. Engrg.* 108 (1993) 133–146.
- [26] H. O. May, The conjugate gradient method for unilateral problems, *Comput. Struct.* 12 (4) (1986) 595–598.
- [27] F. Mehrez, *Modélisation du Contact Frottement dans la simulation numérique de l'emboutissage des têtes*, Ph.D. thesis, Université Pierre et Marie Curie, Paris 6, juin 1991.
- [28] J. J. Moreau, Unilateral contact and dry friction in finite freedom dynamics, in: J. J. Moreau, P. D. Panagiotopoulos (Eds.), *Non Smooth Mechanics and Applications*, CISM Courses and Lectures, vol. 302, Springer Verlag, Wien, New York, 1998, pp. 1–82.
- [29] J. J. Moreau, Indétermination liée au frottement sec dans le calcul des granulats, in: *Actes du Sixième Colloque National en Calcul des Structures*, vol. 3, CSMA AFM LMS, 2003, pp. 465–472.
- [30] J.J. Moreau, Numerical aspects of sweeping process, *Comput. Methods Appl. Mech. Engrg.* 177 (1999) 329–349.
- [31] J. Necas, J. Jarusek, J. Haslinger, On the solution of the variational inequality to the Signorini problem with small friction, *Bolletino U.M.I.* 5 (17.B) (1980) 796–811.
- [32] S. Nemat Nasser, J. Zhang, Constitutive relations for cohesionless frictional granular materials, *Int. J. Plast.* 18 (2002) 531–547.
- [33] J. S. Pang, Newton's method for B differentiable equations, *Math. Op. Res.* 15 (1990) 311–341.
- [34] F. Radjai, D.E. Wolf, M. Jean, J. J. Moreau, Bimodal character of stress transmission in granular packings, *Phys. Rev. Lett.* 80 (1) (1998) 61–64.
- [35] J. Rajchenbach, Granular flows, *Adv. Phys.* 49 (2) (2000) 229–256.
- [36] M. Raous, S. Barbarin, Conjugate gradient for frictional contact, in: A. Curnier (Ed.), *Proc. Contact Mechanics Int. Symp.*, PPUR, 1992, pp. 423–432.
- [37] M. Renouf, P. Alart, Solveurs parallèles pour la simulation de systèmes multi contacts, *Rev. Eur. Elem. Finis.* 13 (5–7) (2004) 691–702.
- [38] M. Renouf, F. Dubois, P. Alart, A parallel version of the non smooth contact dynamics algorithm applied to the simulation of granular media, *J. Comput. Appl. Math.* 168 (2004) 375–382.
- [39] J.B. Rosen, The gradient projection method for nonlinear programming. Part 1—linear constraints, *J. Soc. Indust. Appl. Math.* 8 (1) (1960) 181–217.
- [40] J.B. Rosen, The gradient projection method for nonlinear programming. Part 2—non linear constraints, *J. Soc. Indust. Appl. Math.* 9 (4) (1961) 514–532.
- [41] H. Troadec, F. Radjai, S. Roux, J.C. Charmet, Model for granular texture with steric exclusion, *Phys. Rev. E* 66 (2002) 041305.
- [42] P. Wriggers, Finite element algorithms for contact problems, *Arch. Comput. Meth. Engrg.* 2 (1995) 1–49.