



HAL
open science

Optimisation numérique et calcul parallèle pour l'étude de milieux divisés bi- et tridimensionnels

Mathieu Renouf

► **To cite this version:**

Mathieu Renouf. Optimisation numérique et calcul parallèle pour l'étude de milieux divisés bi- et tridimensionnels. Modélisation et simulation. Université Montpellier II - Sciences et Techniques du Languedoc, 2004. Français. NNT: . tel-00007686

HAL Id: tel-00007686

<https://theses.hal.science/tel-00007686>

Submitted on 8 Dec 2004

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.

ACADEMIE DE MONTPELLIER

UNIVERSITE MONTPELLIER II

— SCIENCES ET TECHNIQUES DU LANGUEDOC —

THÈSE

Présentée à l'Université des Sciences et Techniques du Languedoc
pour obtenir le diplôme de DOCTORAT

Spécialité : Mécanique, Génie mécanique et Génie Civil
Formation Doctorale : Mécanique des matériaux et des milieux complexes,
des structures et systèmes.
Ecole Doctorale : Informations, Structures, Systèmes.

OPTIMISATION NUMERIQUE ET CALCUL PARALLELE POUR L'ETUDE DES MILIEUX DIVISES BI- ET TRIDIMENSIONNELS

présenté par

Mathieu RENOUF

soutenu le 14 Septembre 2004 devant le jury composé de :

MM.	Pierre Ladevèze	Professeur L.M.T. Cachan	Président du Jury
	Christian Rey	Professeur L.M.T. Cachan	Rapporteur
	Jean-Pierre Vilotte	Professeur IPG Paris	Rapporteur
	Jean-Claude Bénet	Professeur LMGC - Université Montpellier II	Examineur
	Pierre ALart	Professeur LMGC - Université Montpellier II	Directeur de thèse
	Frédéric Dubois	Ingénieur de Recherche CNRS - LMGC	co-encadrant
	Claude Lemaréchal	Professeur INRIA Rhône Alpes	Invité
	Mickaël Barbotou	Maître de Conférence LTS Perpignan	Invité

*Il n'y a point d'art mécanique si petit
et si méprisable qui ne puisse fournir
quelques observations
ou considérations remarquables.*

Leibniz

à mes grand-parents

Remerciements

Pour sa présence, sa confiance et son enthousiasme, pour ces critiques constructives et nos nombreuses discussions, je tiens à exprimer mes remerciements les plus sincères à Pierre Alart, mon directeur de thèse durant ces trois années. Mes remerciements vont également à Frédéric Dubois pour avoir co-encadré ce travail, pour ces remarques pertinentes, son enthousiasme ainsi que son dynamisme. Ils ont su tout deux m'aider à développer mon goût pour la recherche et je leur en suis très reconnaissant.

2003年の一年間私を研究所に受け入れてくださった
岸野ゆうじ教授に心から感謝しております。
岸野教授の暖かい歓待と心やさしいお気遣いのおかげ
で研究に専念することができました。
本当にありがとうございました。

Je tiens à exprimer mes vifs remerciements au Pr. Yuji Kishino, de l'université de Tohoku à Sendai (Japon) pour sa gentillesse et son accueil durant l'été 2003 dans le cadre du programme de collaboration JSPS. Je tiens également à remercier le MEXT et CNRS pour m'avoir permis de participer à ce programme.

Je remercie le professeur Pierre Ladevèze d'avoir accepté la présidence de mon jury ainsi que Christian Rey et Jean-Pierre Vilotte pour avoir accepté d'être mes rapporteurs. Leurs remarques et leur intérêt portés à ce travail mon énormément touché.

Je tiens à remercier vivement Michel Jean pour les discussions durant les Carrigolades et ses remarques toujours intéressantes et l'intérêt qu'il a porté de près et de loin à ce travail.

Pour nos nombreuses discussions, son aide et ses connaissances sur les écoulements granulaires je tiens à remercier Daniel Bonamy et son tambour tournant. Je n'oublie pas les membres du GDR-MiDi, avec une pensée plus particulière à Farhang Radjai, Olivier Dauchot et Alfredo Taboada.

Pour leurs aides, leurs conseils et remarques toujours objectives pendant mes trois années de monitorat, je remercie Claude Do, Louis Paris et Franck Jourdan. Leur pédagogie et leurs connaissances m'ont été très utiles lors de mes trois années de monitorats.

Je remercie énormément ma mère pour sa confiance, son soutien et ses tartes aux fruits tout au long de ma "longue" scolarité, du CP à la Thèse.

Une énorme pensée à mon instit préférée, Katia, tout simplement par ce qu'elle est là.

J'ai une très forte pensée à toute ma famille et en particulier à mon père, Serge, Emilie, Marine, Enzo, et mes grands parents.

Je n'oublierais pas de remercier du fond du coeur *Milf production (c)*, alias Sophie, Pierrot et Cyril pour *Pop Mortem*, *Otaku* et *Thé Dansant* pour le Rock'n Roll, pour le Cinéma, pour notre Cinéma, ... en n'oubliant pas que rien n'est fait et que tout reste à faire.

Un merci à Gilles Saussine avec qui j'ai pu partager les joies du développement et du debuggage. Je n'oublie pas mes consoeurs et confrère doctorants alias Aude ("la grande"), Aude ("la petite"), Sylvie, Jean-Michel, Ken'ichi et Matthieu pour nos grands bols d'air. Une dernière pensée à tout ceux que je n'ai pas cité, de Montpellier ou d'ailleurs, Universitaire ou non, que j'ai rencontré durant ces trois années.

Table des matières

Introduction générale	1
I Granulats numériques	5
1 Méthodes numériques pour éléments discrets	7
1.1 Introduction	7
1.2 Cadre d'étude	8
1.3 Molecular Dynamics (<i>MD</i>)	12
1.3.1 Schéma prédicteur-correcteur de Gear	12
1.3.2 Forces de contact	13
1.3.3 Choix des paramètres	14
1.4 Granular Element Method (GEM)	15
1.4.1 Schéma de Newmark	15
1.4.2 Forces de contact	16
1.5 Non Smooth Contact Dynamics (<i>NSCD</i>)	18
1.5.1 Schéma temporel : θ -méthode	18
1.5.2 Loi de contact en vitesse	20
1.5.3 Un solveur non régulier	21
1.5.4 Critère de convergence et qualité de solution.	24
1.6 Comparaisons numériques	26
1.6.1 LMGC90 : plate-forme de développement	27
1.6.2 MD vs NSCD	27
1.6.3 GEM vs NSCD	29
1.7 Un choix parmi d'autre	30
2 Milieux granulaires : paramètres d'étude	31
2.1 Compacité et nombre de coordination	31
2.2 Tenseur de contrainte discret	32
2.3 Anisotropie et tenseur de fabrication	34
2.4 Réseaux forts et faibles	34
2.5 Conclusion	37

II	Optimisation numérique	39
3	Implémentations et Multithreading	41
3.1	Analyse du temps de calcul	42
3.2	Implémentation du solveur Gauss-Seidel non linéaire	45
3.2.1	Méthode ELG (sans assemblage)	45
3.2.2	Méthode SDL (avec assemblage)	47
3.2.3	Que choisir ?	47
3.3	Multithreading	50
3.3.1	Bref tour d’horizon	50
3.3.2	Mais pourquoi OpenMP ?	53
3.3.3	Analyse des tests	55
3.3.4	Influence sur la solution	60
3.4	vers un code 100% parallèle ?	63
4	Algorithmes de type gradient	65
4.1	(Quasi) optimisation en mécanique du contact	66
4.1.1	Problèmes types; <i>Structures</i> et <i>Granulat</i>	66
4.1.2	Conditions de frottement	67
4.2	Les méthodes de gradient (projeté)	69
4.2.1	Les méthodes de Rosen généralisées	69
4.2.2	Un algorithme de gradient projeté conjugué	71
4.2.3	Application au contact unilatéral, au frottement.	73
4.3	Un algorithme spécifique pour le contact frottant	74
4.3.1	Algorithme de point fixe	74
4.3.2	Version diagonalisée	74
4.4	Preconditionneur et critères de convergence	78
4.4.1	Quelques remarques sur l’implémentation	79
4.4.2	Préconditionneurs de base	79
4.4.3	Critères de convergence pour systèmes non réguliers	81
4.5	Tests numériques en mécanique des milieux granulaires	82
4.5.1	Sur un seul pas de temps	82
4.5.2	Influence du frottement	85
4.5.3	Sur un processus complet	86
4.5.4	Multithreading	91
4.6	Conclusion	93
5	Modélisation tridimensionnelle	95
5.1	NSCD, Newton et Multithreading	95
5.1.1	Cas général : Méthode de Newton Généralisée	96
5.1.2	Cas particulier : échantillons de sphères	98
5.1.3	Implémentations	100
5.1.4	Multithreading	101
5.2	Extension du GCP au 3D	102

5.2.1	Une première approche	103
5.2.2	Approximation du cône de Coulomb	104
5.2.3	Schéma hybride	107
5.3	Tests numériques	109
5.3.1	Cycles	109
5.3.2	Sur un pas de temps	110
5.4	Conclusion	113
 III Applications : entre fluide et solide		115
6	Ecoulements granulaires	119
6.1	Panorama	119
6.2	Cadre théorique	120
6.3	Analyse 2D	121
6.3.1	Méthodologie	121
6.3.2	Frottement dynamique	122
6.3.3	Profil de vitesse	123
6.3.4	Vitesse angulaire	127
6.3.5	Compacité	127
6.3.6	Coordinence	128
6.3.7	Contraintes	129
6.3.8	Discussion	129
6.4	Analyse 3D	131
6.4.1	Méthodologie	131
6.4.2	Profils de vitesses	132
6.4.3	Coordinence	132
6.5	Frottement latéral et vitesse de rotation	134
7	Mouvements tectoniques	135
7.1	Etude du terrain	135
7.2	Investigations numériques	138
7.2.1	Choix du nombre de particules	139
7.2.2	Géométries	141
7.3	Propagation de failles	145
7.3.1	Modèle	145
7.3.2	Méthodologie	146
7.3.3	Analyses	146
7.4	Formation de plis	152
7.4.1	Entrée en matière	152
7.4.2	Observations	153
7.4.3	Conclusion	157
8	Conclusion	159

Bibliographie

167

Introduction générale

En calcul des structures les problèmes de contact sont souvent présentés comme difficiles à résoudre et ceci même si le contact est localisé sur une petite part de la frontière des corps en contact. Ceci provient essentiellement de deux difficultés. La première tient à la description elle-même des surfaces potentielles de contact des deux corps antagonistes, inconnues à part entière du problème ; dans une modélisation éléments finis selon la cinématique envisagée l'attirail de la mécanique des grandes transformations doit être utilisé ; la modélisation du contact entre corps rigides polyédriques pose des difficultés techniques comparables [30]. Mais, une fois les variables cinématiques et statiques définies, la deuxième difficulté est due à la forte non linéarité des lois de comportement de l'interface où la prise en compte de l'unilatéralité du contact, du frottement ou même de l'adhésion implique des relations multivoques entre les variables statiques et cinématiques sur l'aire potentielle de contact. De telles difficultés ont conduit à développer une large variété d'outils numériques. Dans le contexte de la modélisation par éléments finis, où le nombre de contacts est la plupart du temps inférieur au nombre de degrés de liberté, une large revue des approches et techniques est donnée dans [155]. Une fois les surfaces potentielles de contacts identifiées, la résolution du problème de contact consiste en la détermination des surfaces réelles de contact, de la répartition des statuts (glissant, adhérent, si le frottement est considéré) de sous régions les composant et enfin des efforts qui y règnent. Si la situation étudiée comporte une multitude de surfaces potentielles de contact, on parle alors de problème *multicontact*. Entrent dans cette définition les milieux granulaires considérés comme collections de solides rigides [26, 78, 117] ; dans ce cas ces derniers interagissent essentiellement par des contacts ponctuels (ou modélisés comme tels) et non des surfaces ; il n'en reste pas moins qu'alors, d'un point de vue numérique, le nombre de contacts excède le nombre de degrés de liberté du système donnant tout son sens au terme "multicontact". On peut cependant considérer comme tels d'autres systèmes comportant de nombreuses zones de contact potentielles et ceci même si le rapport (contacts)/(degrés de liberté) reste petit : ceci concerne ainsi certains assemblages de structures [2, 20], les milieux micro-fissurés [87] ou les matériaux cellulaires [5].

Un milieu granulaire est par définition un ensemble de particules, de taille plus ou moins grande (du grain de sable au bloc de pierre de plusieurs mètres), de forme plus ou moins complexe (de la sphère "parfaite" au polyèdre anguleux), qui interagissent par contacts. Ces interactions mettent en jeu, en sus du contact unilatéral, du frottement, voire de l'adhésion de diverses origines (corrosion, capillarité,...). Un tas de sable ou de

gravats, du béton, un échantillon d'argile, une maçonnerie, une formation géologique, un ensemble de pilules pharmaceutiques peuvent être vu comme des milieux granulaires. On retrouve ces matériaux dans de nombreux secteurs industriels aux problématiques diverses : stockage et vidange des silos de blé [52], tassement d'un ballast au passage répété d'un train [140], mélange de pilules pharmaceutiques, ruine d'un édifice monumental sous sollicitations sismiques, etc. Cette diversité renvoie au comportement atypique d'un milieu granulaire, tantôt solide (béton, plaques tectoniques, ballast) tantôt fluide (avalanche, éboulement, écoulement en silos). Comprendre et modéliser les milieux granulaires apparaît donc comme un enjeu non seulement pratique, mais également scientifique.

En complément de l'approche expérimentale et des essais conventionnels (essai biaxial et/ou de cisaillement) permettant de relier la déformation du milieu à la contrainte qu'il subit, les simulations numériques apparaissent aujourd'hui comme un outil incontournable pour venir enrichir les connaissances apportées par les expériences. Elles permettent d'obtenir des informations impossibles à obtenir expérimentalement, notamment les forces de contact locales, de simuler des milieux parfaits et de jouer plus facilement avec les nombreux paramètres physiques (coefficient de frottement, de restitution). A la fin des années 70, des méthodes numériques dédiées aux éléments discrets voient le jour sous l'impulsion de P. A. Cundall qui développe la première méthode numérique dédiée à la simulation des milieux discrets (*Distinct Element Method* [38]). Rapidement d'autres méthodes voient le jour avec toutes pour même objectif la simulation des milieux discrets. On distingue deux grandes classes de méthode : les méthodes régulières plus souvent qualifiées par le terme anglais "*Smooth*" [9, 77, 151] et les non régulières dites aussi "*Non Smooth*" [40, 71, 55].

Dans le but de simuler des systèmes moins académiques et plus réalistes, la taille des problèmes à traiter augmente régulièrement pouvant parfois conduire à des calculs énormément coûteux en temps de calcul, voire impossible à résoudre. Oeuvrer pour la diminution du temps de calcul apparaît comme un enjeu important. Ce fait est d'autant plus primordial que pour des matériaux tels que les matériaux granulaires (également pour les matériaux cellulaires) on ne peut pas forcément définir le comportement macroscopique par des procédures d'homogénéisation. Pour les matériaux granulaires la définition d'un volume élémentaire représentatif (V.E.R.) est toujours une question ouverte et certaines techniques d'homogénéisation s'appliquent uniquement au comportement statique des granulats [26, 27, 114]. L'évolution d'un milieu granulaire est un processus erratique pouvant avoir une multitude de chemins possibles lors d'une évolution, des phénomènes de localisation (bandes de cisaillement) et des crises dynamiques locales (ruptures d'arches). Capturer tous ces événements aide à mieux comprendre le comportement de tels milieux, et en l'absence d'une théorie micro-macro qui permettrait de les prendre en compte sans les reproduire, offre une garantie sur la pertinence de la simulation.

En conséquence nous devons posséder des outils numériques performants alliant robustesse et précision pour résoudre les problèmes multicontacts impliquant des algorithmes pouvant être efficacement parallélisés. En effet les techniques de calcul parallèle largement utilisées dans le domaine des éléments finis, apparaissent comme une première façon

d'aborder la problématique du temps de calcul. Certaines méthodes par éléments discrets bénéficient de telles techniques. Ces techniques, bien mises en œuvre et appliquées aux opérations les plus consommatrices en temps de calcul, sont susceptibles de diminuer de manière importante le temps de calcul, si une machine multiprocesseur est disponible. Une autre façon d'aborder la diminution du temps de calcul est le développement d'algorithmes plus performants. Ces développements sont d'autant plus intéressants que l'on peut, si ceux-ci sont judicieusement choisis, leur appliquer les techniques de calcul parallèle bénéficiant ainsi d'une double optimisation. Les travaux qui suivent présentent des développements entrant dans ce cadre d'étude.

Le mémoire est décomposé en trois grandes parties et sept chapitres. La première partie présente la problématique générale des "**granulats numériques**", avec un premier chapitre consacré aux différentes méthodes numériques et un second aux paramètres physiques que l'on peut extraire des simulations. La deuxième partie constitue le coeur du mémoire puisque concernant **l'optimisation numérique**. Le chapitre 3 s'attache à tester différentes implémentations séquentielles et parallèles de l'algorithme de Gauss-Seidel non linéaire par blocs en général attaché à l'approche *Non Smooth Contact Dynamics* développée par J.-J Moreau & M. Jean [70, 104]. On introduit dans le chapitre 4 un nouvel algorithme de type gradient projeté conjugué adapté au contact frottant en modélisation bidimensionnelle dont les performances sont étudiées tant en séquentiel qu'en parallèle. Le chapitre 5 étend à la modélisation tridimensionnelle les approches développées dans les chapitres précédents. La troisième et dernière partie est consacrée aux **applications** ayant motivé les développements numériques précédents et ayant suscité, pour certaines, des collaborations ; deux dynamiques **entre fluide et solide** sont étudiées. Le chapitre III réinvestit le thème récurrent des écoulements granulaires en tambour tournant avec des simulations bi- et tridimensionnelles. Le chapitre 7 s'attaque à la simulation des "boîtes à sable" utilisées par les géophysiciens dans l'étude de l'évolution des plaques tectoniques.

Première partie

Granulats numériques

Chapitre 1

Méthodes numériques pour éléments discrets

1.1 Introduction

De nombreuses méthodes numériques sont utilisées pour étudier le comportement dynamique ou quasi-statique de collections de corps considérés comme parfaitement rigides. Les domaines d'application de telles méthodes incluent la robotique [15], la dynamique des maçonneries [1], les évolutions géophysiques [42] et la mécanique des milieux granulaires [26]. Le traitement de corps rigides impose des stratégies de résolution différentes de celles liées à la méthode des éléments finis utilisée pour étudier l'évolution de corps déformables. Lorsque le nombre de contacts entre corps se révèle être peu élevé, les méthodes de type *Event Driven* s'avèrent être de très bonnes candidates. Ces méthodes consistent, lors d'une évolution, à déterminer les moments de transitions (contact, décollement, ...) des moments où les statuts de contact restent inchangés. Sur ces derniers on peut alors utiliser des méthodes d'intégrations d'ordre élevé permettant ainsi des mouvements précis et rapides entre les transitions. Les systèmes de tensegrité [91] et la robotique [55] sont des domaines où de telles méthodes s'avèrent efficaces et ont montré de très bons résultats. Les problèmes de contact sont alors reformulés en terme de problèmes de complémentarité (*L.C.P.* pour *Linear Complementarity Problem*) bénéficiant ainsi des outils de l'analyse non régulière. Lorsque le nombre de contacts devient élevé, ces méthodes deviennent alors coûteuses voire impraticables pour des contacts forcément quasi-simultanés. Or les milieux granulaires sont des milieux typiquement multi-corps générant un grand nombre de contacts simultanés; ainsi on parlera volontiers de systèmes multicontacts. Pour traiter l'évolution de collections de particules en contact, on fait alors appel aux méthodes par éléments discrets communément appelées "*Discrete Element Methods*" par analogie aux "*Finite Element Methods*" (FEM). Le développement des méthodes par éléments discrets voit le jour grâce aux travaux novateurs de P.A. Cundall basés sur le développement d'une méthode numérique dédiée aux éléments discrets - *the Distinct Element Method* (DEM) - développée initialement pour l'étude de systèmes composés de rocs [38] et étendue à l'étude des milieux granulaires [39]. L'extension des méthodes utilisées pour modéliser le

comportement dynamique des molécules aux particules macroscopiques, appelée à juste titre dynamique moléculaire (Molecular Dynamics - MD) [9], ajoute une contribution à ces développements. Ces deux méthodes régularisent le contact entre particules. En effet elles utilisent des lois de répulsions raides reliant forces et paramètres de configuration pour approcher la contrainte mécanique de non-pénétrabilité (conduisant numériquement à une légère inter-pénétration des particules). La raideur des lois poussent à introduire des paramètres de dissipation reliés aux vitesses des corps et à garder des pas de temps petits pour assurer la stabilité du schéma d'intégration temporel. Un dernier type de méthode se distinguant de cette stratégie est la méthode *Contact Dynamics* développé par Moreau et Jean [71, 104, 105] qui renonce à régulariser les lois : ainsi le contact est réellement unilatéral. On parle alors de méthodes "Smooth" pour DEM et MD et de "Non Smooth" pour CD où les auteurs précités ont par ailleurs complété l'intitulé de leur approche pour devenir "Non Smooth Contact Dynamics" (NSCD) [70]. Le caractère non-régulier apparaît dans trois aspects. Primo une *non-linéarité spatiale* due à la condition géométrique de non-pénétration conduisant à traiter des inégalités au lieu d'égalités. Secundo une *non-linéarité temporelle* liée aux chocs entre corps créant ainsi des discontinuités dans les vitesses. Tertio une *non-linéarité en loi de contact* apparaissant à travers les lois non-régulières reliant les forces aux paramètres de configuration (contact unilatéral).

Il paraît alors intéressant de comparer un peu plus en détail ces trois méthodes en éclairant les points communs ainsi que les différences. Après une rapide présentation des différentes étapes d'un code dédié aux éléments discrets, nous présenterons la méthode MD, la "Granular Element method" (GEM) développé par Kishino [77] s'appuyant sur DEM, puis pour finir la méthode NSCD. Quelques comparaisons numériques seront faites et nous expliquerons les points qui nous ont poussé à opter pour la méthode NSCD. Par la suite nous utiliserons souvent la terminologie anglaise pour présenter les méthodes permettant de conserver la compréhension des observations.

1.2 Cadre d'étude

Les trois méthodes (MD, GEM et NSCD) ont le même cadre de départ. On considère un système composé de n_b particules parfaitement rigides ; on note n le nombre de paramètres de configuration : $n = 3 * n_b$ ou $n = 6 * n_b$ suivant la nature du problème (2D et 3D). Nous supposons pour la suite de ce chapitre, pour simplifier la présentation et les illustrations, que *les particules sont circulaires et que le problème est plan*. On définit alors un paramétrage $t \rightarrow \mathbf{q}(t)$ ($\in \mathbb{R}^n$) définissant la position des particules à un instant donné. Considérons pour l'instant que \mathbf{q} est suffisamment régulier pour nous permettre d'écrire l'équation de la dynamique pour chaque particule, obtenant le système suivant

$$\mathbb{M}\ddot{\mathbf{q}}(t) = \mathbf{F}_{ext}(t) + \mathbf{R}, \tag{1.1}$$

où \mathbb{M} ($\in \mathbb{R}^{n \times n}$) est la matrice de masse du système (comprenant les composantes de masse et d'inertie), $\ddot{\mathbf{q}}$ le vecteur accélération ($\in \mathbb{R}^n$) dérivée seconde du vecteur configuration \mathbf{q} , \mathbf{F}_{ext} les forces extérieures en l'absence de contact et \mathbf{R} les forces de contacts. La somme de ces deux derniers termes est notée \mathbf{F} . Les trois méthodes s'appuient sur l'équation (1.1)

mais en prenant toutes les trois des itinéraires différents pour déterminer le mouvement du système. L'équation (1.1) est discrétisée en temps via un schéma d'intégration temporel. Mais pour avoir les idées plus claires sur les différents points pouvant être propres à chaque méthode ou communs, nous allons considérer ce qui peut apparaître comme un algorithme général œuvrant pour la simulation des éléments discrets (voir figure 1.1) en distinguant trois étapes fondamentales. Une fois ces étapes dissociées nous verrons comment elles sont traitées par chaque approche.

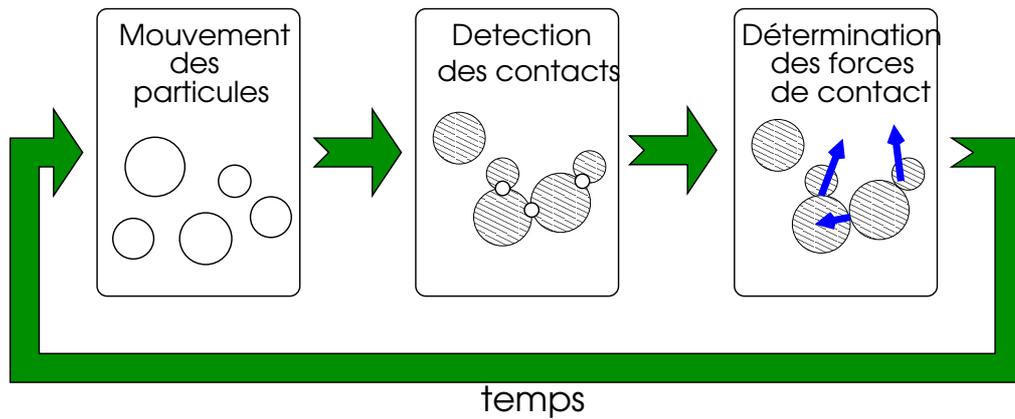


FIG. 1.1 – Schéma caractéristique de l'algorithme général d'une méthode œuvrant pour la simulation du mouvement d'éléments discrets. Seules les trois étapes principales sont représentées.

Ces trois parties sont le mouvement des particules, la détection des contacts et la résolution des forces de contact. Passons les en revue.

Mouvement des particules

A chaque particule i du système, correspond un vecteur \mathbf{q}_i qui peut se décomposer en deux parties : un vecteur translation \mathbf{x}_i et une rotation ω_i (figure 1.2) ; on rappelle que nous limitons la présentation dans ce chapitre à une modélisation bidimensionnelle. De même le torseur des efforts appliqués à chaque particule, noté \mathbf{F}_i , comprenant les forces extérieures ainsi que les forces de contact, est décomposé en sa résultante \mathbf{N}_i et son moment M_i (figure 1.4).

La détermination du mouvement des particules est liée au schéma d'intégration. Nous sommes amenés à travailler sur des intervalles de temps $[t^i, t^{i+1}]$. Le principe est de prédire une position ou d'effectuer directement un déplacement via un schéma d'intégration temporel. Ne connaissant qu'une partie des forces au temps t^i , nous effectuons une correction de cette position lorsque l'ensemble des forces est connu, nous permettant ainsi de connaître la position du système à l'instant t^{i+1} . Le schéma d'intégration dépend de la méthode

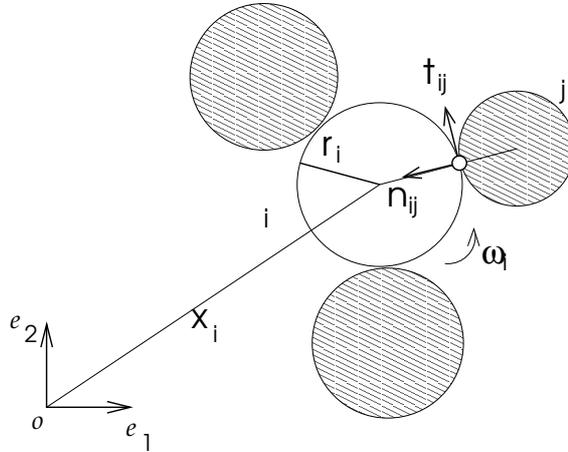


FIG. 1.2 – Définition du repère direct local (\mathbf{t}, \mathbf{n})

utilisée et les différentes méthodes ne se basent pas sur les mêmes schémas d'intégration temporel (méthode de Gear, schéma de Newmark, θ -méthode).

Détection des contacts

Une fois la position des particules prédite, de nouveaux contacts peuvent apparaître tandis que des contacts présents à l'instant précédent ont pu disparaître. La méthode dressant la liste des contacts potentiels est indépendante de la méthode (MD, GEM ou NSCD). Nous pouvons donc très bien utiliser le même algorithme de détection pour les trois méthodes : une méthode des boîtes de Manhattan [50], une triangulation de Delaunay [31] ou encore des méthodes plus exotiques [24, 48]. Pour dire simplement quelques mots par rapport aux méthodes utilisées, nous présentons le principe de la méthode que nous utilisons dans les différentes simulations présentées ainsi qu'une autre méthode plus souvent utilisée qui est une méthode de détection par triangulation de Delaunay. La méthode des boîtes de Manhattan consiste à découper le système en boîte de mêmes tailles. Ce découpage se fait en fonction du diamètre des particules. Une fois le découpage effectué, on répartit les particules dans les boîtes correspondantes. Ensuite on procède à la détection des contacts entre les particules d'une même boîte, ainsi qu'entre les particules appartenant à deux boîtes adjacentes (boîtes hachurées sur la figure 1.3) : toutes les boîtes adjacentes ne sont pas parcourues car suivant l'ordre de parcours des boîtes, des détections entre boîtes ont pu déjà être faites, ce qui nous permet de limiter le nombre de tests effectués. La triangulation de Delaunay consiste quant à elle à réaliser un maillage entre les centres des corps du système, nous permettant ainsi d'obtenir directement la liste des arêtes correspondant aux contacts potentiels (c.f. partie droite de la figure 1.3).

Quoi qu'il en soit ces méthodes doivent être efficaces pour ne pas consommer la majeure partie du temps. On trouve ainsi de nombreuses méthodes robustes, apparaissant souvent

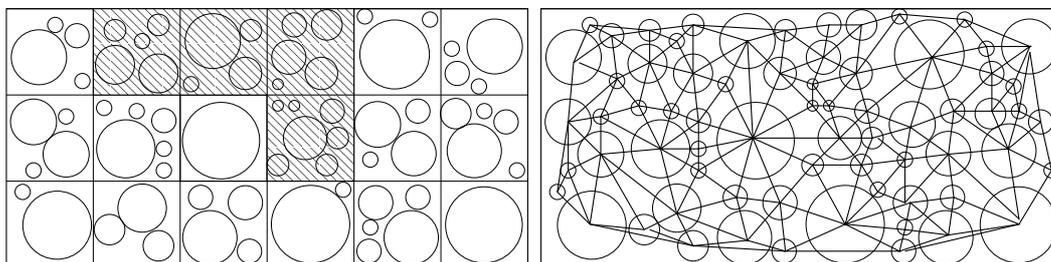


FIG. 1.3 – Algorithme de détection des contacts : méthodes des boîtes (gauche) et triangulation de Delaunay (droite)

comme des extensions de celles présentées précédemment, dont certaines possèdent des versions parallèles permettant de bénéficier des super-calculateurs [31]. Nous ne détaillerons pas plus les différents algorithmes de détection même si ceux-ci s'avèrent être d'un très grand intérêt. Pour les trois méthodes, lors de cette phase, nous définissons pour chaque paire de particules en contact un repère local (illustré par la figure 1.2) dans lequel sont exprimées les grandeurs locales, vitesse relative et force de contact locale.

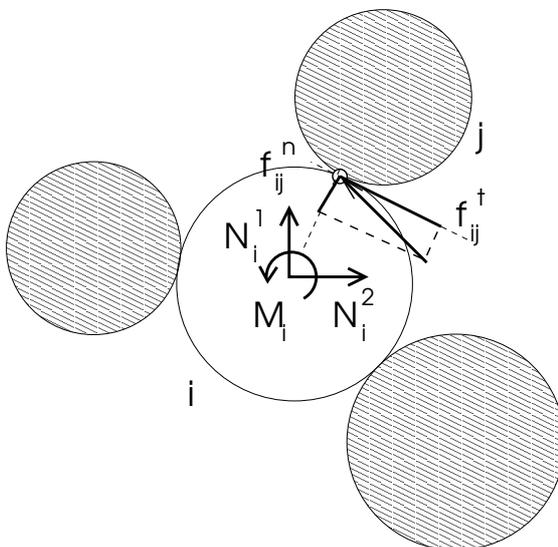


FIG. 1.4 – Représentation des forces locales et globales

Pour chaque contact α , on définit un repère direct $(I_\alpha; \mathbf{t}_\alpha, \mathbf{n}_\alpha)$, dans lequel on exprime la force ou l'impulsion de contact locale \mathbf{r}_α décomposée en une partie normale $r_{\alpha,n}$ et une partie tangentielle $r_{\alpha,t}$ (figure 1.2); cette dernière quantité est vectorielle en 3D.

Détermination des forces de contacts

Pour finir ou corriger le déplacement des corps composant le système, il nous faut procéder à la phase consistant à exprimer les forces de contact, seules grandeurs qui restent inconnues. Cette détermination peut se faire de manière explicite en utilisant des lois régularisantes, raides, reliant configuration et force (caractère *smooth*) comme pour les méthodes *MD* et *DEM*, ou de façon implicite nécessitant un algorithme itératif pour résoudre le problème non linéaire local comme le fait la méthode *NSCD* (caractère *non smooth*) en utilisant classiquement un algorithme de type Gauss-Seidel mais pouvant aussi utiliser une approche de type bi-potentiel [51, 53]. Si l'on définit δ comme la différence entre la distance inter-centres et la somme des rayons des particules, $\delta = d_{ij} - (l_i + l_j)$, les méthodes *smooth* écrivent $r_{\alpha,n}$ comme une fonction de δ (par exemple $r_{\alpha,n} = k_n \max(0, -\delta)$) tandis que les méthodes *non smooth* travaillerons avec des inégalités à satisfaire du type $\delta \geq 0$, $r_{\alpha,n} \geq 0$ pour du contact unilatéral. Les forces de contact peuvent n'être que normales en l'absence de frottement, posséder une composante tangentielle lorsque le frottement est pris en compte, ou encore plus complexes faisant intervenir des effets thermiques, des phénomènes d'usure ou encore des interactions à distance. Le choix de la loi de contact dépend du problème traité. Malgré un formalisme assez général sur certaines lois, nous ne traiterons par la suite que des problèmes de contact avec ou sans frottement. La cohésion entre particules provenant des phénomènes de capillarité ou autre point de colle n'a pas été utilisée dans les simulations présentées dans ce manuscrit.

Nous nous proposons maintenant de regarder plus en détails, pour chaque méthode, les parties concernant le schéma d'intégration temporel ainsi que la détermination des forces de contact. Nous porterons une attention plus particulière à l'approche *NSCD* que nous avons utilisé pour étudier le comportement des milieux granulaires.

1.3 Molecular Dynamics (*MD*)

L'approche dynamique moléculaire est l'adaptation à l'échelle macroscopique [9] de méthodes utilisées couramment pour la modélisation de systèmes moléculaires. A cette échelle (Angström), les interactions entre molécules dérivent de potentiels mettant en jeu des forces en général négligées à l'échelle du mécanicien. Ces derniers sont alors remplacés par des potentiels modélisant l'interaction élastique entre particules dont le plus connu est le modèle de Hertz. Parmi les nombreux travaux utilisant cette approche pour la modélisation dynamique ou statique de collections de corps en contact, on peut citer les travaux de Walton [150, 151], Ristow [133] et Hermann [61, 62]. Après une présentation succincte de la discrétisation temporelle, nous présentons une formulation générale du calcul des forces de contact.

1.3.1 Schéma prédicteur-correcteur de Gear

La discrétisation des équations de la dynamique se fait via une méthode de différences finies associée à un algorithme prédicteur-correcteur de Gear. Le principe en est le sui-

vant. Partant des positions, vitesses et autres grandeurs dynamiques à un instant t^i ($\mathbf{q}(t^i)$ sera noté $\mathbf{q}(i)$, notation adoptée pour les autres quantités dépendant du temps), on veut déterminer ces mêmes quantités à l'instant $t^{i+1} = t^i + \Delta t$ pour une certaine précision. Suivant un développement classique de type Taylor nous pouvons écrire,

$$\left\{ \begin{array}{l} \mathbf{q}^p(i+1) = \mathbf{q}(i) + \Delta t \dot{\mathbf{q}}(i) + \frac{(\Delta t)^2}{2!} \ddot{\mathbf{q}}(i) + \frac{(\Delta t)^3}{3!} \mathbf{b}(i) + \frac{(\Delta t)^4}{4!} \mathbf{c}(i) + \dots \\ \dot{\mathbf{q}}^p(i+1) = \dot{\mathbf{q}}(i) + \Delta t \ddot{\mathbf{q}}(i) + \frac{(\Delta t)^2}{2!} \mathbf{b}(i) + \frac{(\Delta t)^3}{3!} \mathbf{c}(i) + \dots \\ \ddot{\mathbf{q}}^p(i+1) = \ddot{\mathbf{q}}(i) + \Delta t \mathbf{b}(i) + \frac{(\Delta t)^2}{2!} \mathbf{c}(i) \dots \\ \mathbf{b}^p(i+1) = \mathbf{b}(i) + \Delta t \mathbf{c}(i) + \dots \\ \mathbf{c}^p(i+1) = \mathbf{c}(i) + \dots \end{array} \right. , \quad (1.2)$$

où \mathbf{b} et \mathbf{c} sont respectivement les dérivées première et seconde de l'accélération. L'algorithme de Gear seul ne permet pas de déterminer la position des corps. En effet l'équation de la dynamique n'étant pas encore résolue, on ne peut déterminer exactement la position des corps sans prendre en compte les efforts de contact : on parle alors de grandeurs prédites (indexées par p). Ainsi une fois les forces de contact déterminées, l'accélération corrigée peut être calculée ($\ddot{\mathbf{q}}^c(i+1) = \mathbb{M}^{-1} \mathbf{F}(i+1)$) fournissant un terme correctif noté $\Delta \ddot{\mathbf{q}}$ égal à $\ddot{\mathbf{q}}^c(i+1) - \ddot{\mathbf{q}}^p(i+1)$ permettant d'actualiser les différentes grandeurs prédites :

$$\left\{ \begin{array}{l} \mathbf{q}^c(i+1) = \mathbf{q}^p(i+1) + c_0 \Delta \ddot{\mathbf{q}} \\ \dot{\mathbf{q}}^c(i+1) = \dot{\mathbf{q}}^p(i+1) + c_1 \Delta \ddot{\mathbf{q}} \\ \ddot{\mathbf{q}}^c(i+1) = \ddot{\mathbf{q}}^p(i+1) + c_2 \Delta \ddot{\mathbf{q}} \\ \mathbf{b}^c(i+1) = \mathbf{b}^p(i+1) + c_3 \Delta \ddot{\mathbf{q}} \\ \mathbf{c}^c(i+1) = \mathbf{c}^p(i+1) + c_4 \Delta \ddot{\mathbf{q}} \end{array} \right. . \quad (1.3)$$

Les coefficients c_0 , c_1 , c_2 , c_3 et c_4 sont appelés coefficients de Gear et sont reliés à l'ordre de l'équation différentielle (1.1) ainsi qu'au nombre de quantités considérées [9]. Le tableau 1.1 donne des exemples de valeurs suivant l'ordre utilisé dans le développement.

valeurs	c_0	c_1	c_2	c_3	c_4
3	0	1	1	0	0
4	1/6	5/6	1	1/3	0
5	19/90	3/4	1	1/2	1/12

TAB. 1.1 – Coefficients de Gear pour le schéma prédictor-correcteur.

1.3.2 Forces de contact

En se basant sur l'équation (1.1), il nous faut déterminer les forces de contact exercées sur chaque corps notées $\mathbf{R} = (\mathbf{R}_1, \dots, \mathbf{R}_{n_b})$ contribuant au second membre de (1.1). Celles-ci sont calculées en fonction des forces de contact locales \mathbf{r}_α , fonctions de l'interpénétration δ (si δ est négatif) dans une démarche régularisante. Ainsi les forces normales de contact

peuvent suivre une simple loi de Hertz, i.e. $r_n \approx k\delta^{\frac{3}{2}}$ dans le cas 2D, ou utiliser d'autres lois comme le propose Johnson *et al.* [72] avec le modèle JKR prenant en compte les effets de cohésion entre particules, effets largement étudiés dans [123]. Nous présentons un modèle utilisé dans les écoulements denses [132, 133] s'apparentant à une loi de Hooke. En définissant $\mathbf{d}_{ij}(= d_{ij}\mathbf{n}_{ij})$ comme le vecteur inter-centres des particules i et j

$$\begin{cases} r_{ij,n} = \{k_n\delta + \gamma_n \frac{m_{eff}\mathbf{u}_{ij}\cdot\mathbf{d}_{ij}}{l_{eff}d_{ij}}\} \\ r_{ij,t} = \min\{-\gamma_t m_{eff}v_{ij,t}, \text{sign}(-\gamma_t m_{eff}v_{ij,t})\mu|f_{ij,n}|\} \end{cases} \quad (1.4)$$

où

$$\begin{aligned} m_{eff} &= \frac{m_i m_j}{m_i + m_j} \\ l_{eff} &= \frac{l_i l_j}{l_i + l_j} \end{aligned} \quad ,$$

m_{eff} et l_{eff} représentent respectivement la masse effective et le rayon effectif associés au contact ij . Ce modèle impose d'introduire trois coefficients d'ajustement bien que qualifiés mécaniquement : k_n , coefficient de raideur locale, γ_n et γ_t , coefficient de viscosité. \mathbf{u}_{ij} représente la vitesse relative entre les particules. D'autres variantes pour les forces de contacts sont possibles [96, 142]. On peut ainsi suivant la position des particules dans l'échantillon déterminer les forces attractives (si la cohésion est prise en compte) ou répulsives entre particules en contact et procéder à la correction nous permettant de placer les particules dans leur position à l'instant t^{i+1} . Notons que cette approche fait intervenir des termes élevés en ordre de dérivation dans le schéma d'intégration temporel (dérivées première et seconde de l'accélération), ce qui suppose des évolutions locales très régulières quelque peu contradictoires avec les crises dynamiques locales (y compris pour des évolutions globales quasi-statiques) observées numériquement et expérimentalement (bruits acoustiques).

1.3.3 Choix des paramètres

L'utilisation de loi régularisante peut paraître fort sympathique puisque cela permet de traiter le contact de façon explicite. Cependant certaines précautions sont à prendre pour que les résultats des simulations aient un sens mécanique. L'une de ces précautions concerne le choix du pas de temps. Le pas de temps Δt dépend des paramètres mécaniques du système. Δt doit garder une taille suffisamment petite pour que la détermination des forces de contact soient pertinents. En particulier, Δt doit prendre des valeurs largement inférieure au temps de durée d'un contact, durée qui est fonction des paramètres k_n , m_{eff} et γ_n [123]. Une autre précaution repose sur le choix de la raideur normale. La valeur de k_n ne doit pas être trop élevée pour ne pas prendre des pas de temps trop petits, mais ne doit pas non plus être trop lâche entraînant des valeurs de δ trop importante et sans plus aucune signification. Ces précautions sont primordiales pour obtenir des résultats ayant un sens mécanique en particulier la loi de conservation du volume.

1.4 Granular Element Method (GEM)

La méthode GEM (Granular Element method) [77] est une variante de la méthode DEM développée par P.A. Cundall [38]. Comme MD, GEM appartient aux méthodes dites *smooth* ; le caractère *smooth* est lié à l'utilisation de lois régularisantes. GEM diffère de DEM par le schéma de résolution qui s'apparente à une *stiffness method*, schéma proposé tout d'abord par Kawai [75] et étendu par Kishino pour la simulation quasi-statique des milieux granulaires [77]. Une extension de la méthode sera proposée plus tard par le même auteur pour l'étude de situations dynamiques [80]. Nous proposons ici de présenter brièvement le schéma d'intégration temporel dans le cadre général puis de détailler plus longuement la méthode utilisée pour résoudre le problème de contact local restreint alors aux évolutions quasi-statiques.

1.4.1 Schéma de Newmark

Reprenons l'équation (1.1) en explicitant les forces de contacts $\mathbf{R}(\mathbf{q}, \dot{\mathbf{q}})$, que l'on suppose pouvoir s'écrire de la façon suivante

$$\mathbf{R}(\mathbf{q}, \dot{\mathbf{q}}) = -\mathbb{C}\dot{\mathbf{q}} - \mathbb{K}(\mathbf{q})\mathbf{q}, \quad (1.5)$$

où \mathbb{C} est une matrice de viscosité, $\mathbb{K}(\mathbf{q})$ une matrice de rigidité dépendant de la configuration, toutes deux définies positives, $\dot{\mathbf{q}}$ représentant la vitesse généralisée. On retrouve alors l'équation communément utilisée en dynamique des structures,

$$\mathbb{M}\ddot{\mathbf{q}} + \mathbb{C}\dot{\mathbf{q}} + \mathbb{K}\mathbf{q} = \mathbf{F}_{ext}. \quad (1.6)$$

La matrice \mathbb{K} est alors considérée comme constante sur l'incrément supposant que l'ensemble des contacts gardent leurs statuts. Nous verrons que ce n'est pas toujours le cas, mais cette hypothèse peut être toutefois conservée en prenant les précautions nécessaires. L'équation (1.6) est alors intégrée via un schéma de Newmark utilisant un paramètre $\beta > 0$ pris entre $\frac{1}{2}$ et 1 pour des raisons de stabilité numérique. Ainsi on peut approcher la vitesse et l'accélération par

$$\begin{cases} \dot{\mathbf{q}}(i+1) = \frac{\Delta\mathbf{q}}{\beta\Delta t} - \frac{1-\beta}{\beta}\dot{\mathbf{q}}(i) \\ \ddot{\mathbf{q}}(i+1) = \frac{\Delta\dot{\mathbf{q}}}{\beta\Delta t} - \frac{1-\beta}{\beta}\ddot{\mathbf{q}}(i) \end{cases}. \quad (1.7)$$

Après avoir réalisé une pondération de l'équation (1.6) au temps t^i et t^{i+1} , et substitué successivement les équations du système (1.7), on obtient la relation

$$\begin{aligned} & \left[\frac{1}{\beta\Delta t^2}\mathbb{M} + \frac{1}{\Delta t}\mathbb{C} + \beta\mathbb{K} \right] \mathbf{q}(i+1) = \\ & \left[\frac{1}{\beta\Delta t^2}\mathbb{M} + \frac{1}{\Delta t}\mathbb{C} - (1-\beta)\mathbb{K} \right] \mathbf{q}(i) + \frac{1}{\beta\Delta t}\mathbb{M}\dot{\mathbf{q}}(i) + [\beta\mathbf{F}(i+1) - (1-\beta)\mathbf{F}(i)] \end{aligned} \quad (1.8)$$

qui nous permet ainsi d'exprimer la configuration à l'instant t^{i+1} en fonction de la configuration à l'instant t^i (connue), de ces dérivées premières et secondes et des forces (extérieures et de contact). Pour pouvoir déplacer les particules, il nous reste à exprimer les forces de contact à l'instant t^{i+1} .

1.4.2 Forces de contact

Régularisation

Pour chaque particule i , la liste des particules en contact étant établie, nous décomposons le déplacement relatif entre la particules i et la particule j , candidate au contact, en une partie normale $\Delta x_{ij,n}$ et une partie tangentielle $\Delta x_{ij,t}$. Les efforts de contacts locaux sont alors déterminés en fonction de ces deux quantités,

$$\begin{cases} r_{ij,n} = k_n \max(0, -\Delta x_{ij,n}) = \max(0, -k_n \delta) \\ r_{ij,t} = k_t \Delta x_{ij,t} \end{cases}, \quad (1.9)$$

δ étant l'interpénétration définie dans la section précédente. Prenant en compte le frottement de Coulomb, si il y a glissement, autrement dit si $|r_{ij,t}| > c + \mu r_{ij,n}$ (c coefficient de cohésion et μ coefficient de frottement entre particules), la force tangentielle est égale à

$$r_{ij,t} = \frac{r_{ij,t}}{|r_{ij,t}|} (c + \mu r_{ij,n}) \quad (1.10)$$

La méthode de DEM classique consiste alors à faire remonter ces forces locales au niveau global actualisant les torseurs des efforts exercés sur chaque particule avant de passer au pas suivant. Ce que propose Kishino avec la méthode GEM [77] est différent puisque la méthode est implicite s'appuyant sur les matrices de rigidité utilisées dans la méthode des éléments finis.

Processus itératif

Nous ne présenterons que la méthode pour une étude de problème quasi statique [78], cadre fort agréable puisque l'équation, écrite de façon incrémentale, est réduite à

$$\begin{aligned} \mathbb{K}(\mathbf{q}(i+1))(\mathbf{q}(i+1) - \mathbf{q}(i)) &= \mathbf{F}_{ext}(i+1) - \mathbf{F}_{ext}(i) \\ \Leftrightarrow \mathbb{K}(\mathbf{q}(i+1))\Delta\mathbf{q} &= \Delta\mathbf{F}_{ext} \end{aligned}, \quad (1.11)$$

et permet d'explicitier plus aisément la démarche. La vitesse est considérée constante sur chaque pas de temps impliquant que l'incrément $\Delta\dot{\mathbf{q}}$ est nul. La matrice \mathbb{K} s'écrit

$$\mathbb{K}(\mathbf{q}(i+1)) = \mathbb{H}(\mathbf{q}(i+1))\mathbb{K}_c\mathbb{H}^T(\mathbf{q}(i+1)) \quad (1.12)$$

où

$$\mathbb{K}_c = \text{diag}(\mathbb{K}_e, \alpha = 1, n_c) \text{ et } \mathbb{K}_e = \begin{bmatrix} k_n & 0 \\ 0 & k_t \end{bmatrix}, \quad (1.13)$$

la matrice $\mathbb{H} \in \mathbb{R}^{3n_b \times 2n_c}$ est un assemblage de matrices $\mathbb{H}_{\alpha,i}$ et $\mathbb{H}_{\alpha,j}$ ($\alpha = ij$) défini par,

$$\begin{aligned} \text{Si } \alpha \text{ n'est pas actif} \quad & \mathbb{H}_{\alpha,i} = \mathbb{O} & \mathbb{H}_{\alpha,j} = \mathbb{O} \\ \text{Si } \alpha \text{ est actif} \quad & \mathbb{H}_{\alpha,i} = \begin{bmatrix} \mathbf{n} & \mathbf{t} \\ 0 & 1 \end{bmatrix} & \mathbb{H}_{\alpha,j} = \begin{bmatrix} -\mathbf{n} & -\mathbf{t} \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (1.14)$$

Le principe de la méthode ne consiste pas simplement à passer au pas suivant en inversant la matrice \mathbb{K} , comme le ferait l'approche MD. Afin de prendre en compte l'équilibre du système, un procédé itératif est mis en place. Partant d'une position initiale, les forces de contacts sont calculées suivant le système (1.9) afin de calculer l'équilibre global ainsi que l'équilibre de chaque particule. Si l'équilibre est établi, nous passons au pas de temps suivant. Dans le cas contraire, les particules non équilibrées sont déplacées afin de corriger l'instabilité du système. Les forces de contacts sont ensuite réévaluées en fonction de ce nouvel incrément de déplacement. La matrice \mathbb{K} est gardée constante sur le pas de temps conformément à la remarque faite précédemment. Toutefois il se peut qu'une particule perde l'ensemble de ses contacts au cours du processus itératif auquel cas, la matrice de rigidité est recalculée, afin de poursuivre le processus itératif. La méthode incrémentale s'apparente à un algorithme de Jacobi, au sens où pour calculer l'incrément de déplacement $\Delta \mathbf{q}_i$, il n'est pas nécessaire d'avoir calculé les incréments de déplacement des particules précédentes. Ainsi dans une optique multithread, cet algorithme est intrinsèquement parallèle [131]. L'approche est présentée plus en détail dans l'annexe A. De nombreux résultats ont été obtenus grâce à l'approche GEM [81, 82], un des derniers exemples étant la modélisation d'un essai tri-axial (figure 1.5) pour étudier les différents types de mode de déformation en plasticité. La version initiale de GEM, proposée pour des analyses quasi-statiques bénéficie aujourd'hui d'une version dynamique [80].

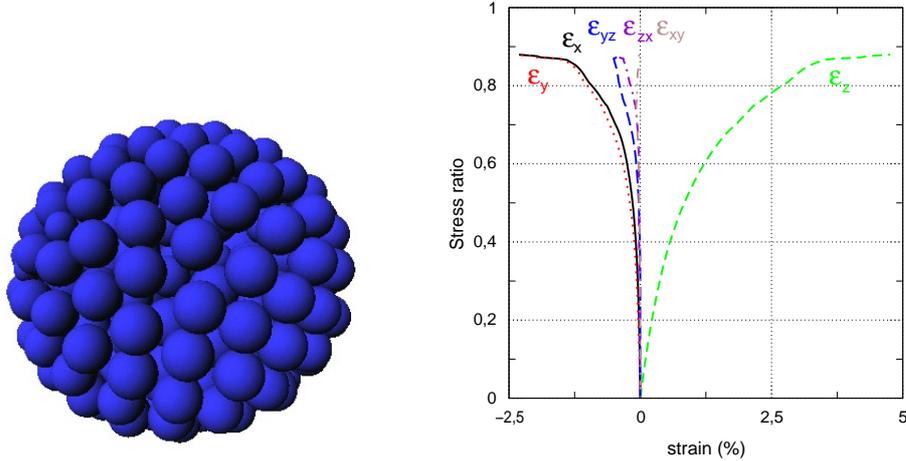


FIG. 1.5 – relation contrainte-déformation lors d'un essai tri-axial : ratio de déformation = valeur absolue du déviateur du tenseur de contrainte / contrainte moyenne

1.5 Non Smooth Contact Dynamics (*NSCD*)

La méthode *Non Smooth Contact Dynamics (NSCD)* n'est pas construite en ayant recours à une régularisation des relations de contact comme les deux approches précédentes et traite au même titre que les méthodes de types *Even Driven* une mise en équation exacte des problèmes à liaisons unilatérales. Issue des travaux de J.-J. Moreau [103, 105] et M. Jean [70, 71], nous présentons dans un premier temps la méthode en essayant de rester fidèle au formalisme développé par les auteurs. Nous présentons une fois l'équation (1.1) remaniée, la discrétisation via une θ -méthode, formalisons les notions de contact unilatéral et frottement de Coulomb pour finir par l'algorithme utilisé pour résoudre le problème de contact. Nous apportons également une attention particulière au critère de convergence de cette méthode.

1.5.1 Schéma temporel : θ -méthode

Le schéma d'intégration utilisé pour discrétiser l'équation (1.1) est une θ -méthode. Lors de l'étude de systèmes multi-contacts interviennent des chocs entre particules. On préfère alors écrire l'équation (1.1) en terme de mesures, et ainsi prendre les dérivées au sens des distribution. En ce sens (1.1) se réécrit

$$\mathbb{M}d\dot{\mathbf{q}} = \mathbf{F}_{ext}(\mathbf{q}, \dot{\mathbf{q}}, t)dt + \mathbf{R}d\nu \quad (1.15)$$

où dt est la mesure de Lebesgue, $d\dot{\mathbf{q}}$ une mesure différentielle (mesure accélération) et $d\nu$ une mesure non négative pour laquelle $d\dot{\mathbf{q}}$ possède une densité de mesure. \mathbf{R} est la densité d'impulsion de contact. Ce formalisme permet de modéliser des chocs par des contributions atomiques dans les mesures $d\dot{\mathbf{q}}$ (discontinuité de vitesse) et $d\nu$ (impulsion instantanée). En se plaçant sur l'intervalle $]t^i, t^{i+1}]$, (1.1) est équivalente à

$$\begin{cases} \mathbb{M}(\dot{\mathbf{q}}(i+1) - \dot{\mathbf{q}}(i)) = \int_{t^i}^{t^{i+1}} \mathbf{F}_{ext}(\mathbf{q}, \dot{\mathbf{q}}, s)ds + \int_{]t^i, t^{i+1}]} \mathbf{R}d\nu \\ \mathbf{q}(i+1) = \mathbf{q}(i) + \int_{t^i}^{t^{i+1}} \dot{\mathbf{q}}(s)ds \end{cases} \quad (1.16)$$

en supposant toutefois que les vitesses possèdent les bonnes propriétés. Devant traiter de chocs multiples et de discontinuités de vitesse, la discrétisation des intégrales ne peut se faire qu'en utilisant le paramétrage et sa dérivée première. Des ordres de dérivation supérieurs ne sont pas nécessaires. *L'impulsion (de contact) moyenne*

$$\mathbf{R}(i+1) = \frac{1}{h} \int_{]t^i, t^{i+1}]} \mathbf{R}d\nu, \quad (1.17)$$

comme pour les autres approches, apparaît comme l'inconnue principale, puisque c'est elle qui va nous permettre de déterminer la vitesse après chocs et ainsi actualiser la configuration du système. La θ -méthode s'apparente à une pondération en fonction des quantités obtenues en début et en fin de pas de temps (i.e. des temps t^i et t^{i+1}). En appliquant ce

principe aux deux intégrales du système (1.16), on écrit

$$\int_{t^i}^{t^{i+1}} \mathbf{F}_{ext}(\mathbf{q}, \dot{\mathbf{q}}, t) dt = h\theta \mathbf{F}_{ext}(i+1) + h(1-\theta) \mathbf{F}_{ext}(i) \quad (1.18)$$

$$\mathbf{q}(i+1) = \mathbf{q}(i) + h\theta \dot{\mathbf{q}}(i+1) + h(1-\theta) \dot{\mathbf{q}}(i)$$

Comme pour le schéma de Newmark, les valeurs de θ doivent être comprises entre $\frac{1}{2}$ et 1 pour permettre au schéma d'intégration d'être inconditionnellement stable. Une valeur de $\theta = 1$ permet de retomber sur le schéma d'Euler implicite. En substituant les équations (1.17) et (1.18) dans les expressions de (1.16), on peut écrire les deux équations

$$\dot{\mathbf{q}}(i+1) = \dot{\mathbf{q}}(i)_{libre} + (\mathbb{M}^{-1})h\mathbf{R}(i+1) \quad (1.19)$$

où

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

et

$$\mathbf{q}(i+1) = \mathbf{q}(i) + h\theta \dot{\mathbf{q}}(i+1) + h(1-\theta) \dot{\mathbf{q}}(i) \quad (1.20)$$

Par définition de \mathbf{F}_{ext} , représentant les forces extérieures en absence de contact, $\dot{\mathbf{q}}_{libre}$ dénote la "vitesse libre". Les inconnues à déterminer pour passer des instants t^i à t^{i+1} sont donc $\dot{\mathbf{q}}(i+1)$ et $\mathbf{R}(i+1)$ variables généralisées. Or les lois d'interaction non encore prises en compte sont locales; elles relient des variables locales relatives aux contacts. On rappelle ici que seules des lois de contact et frottement sont considérées. Cette réécriture nécessite l'introduction de variables locales : \mathbf{u}_α la vitesse relative, \mathbf{r}_α l'impulsion moyenne de contact (α signe le numéro du contact) reliées aux variables généralisées (globales) $\dot{\mathbf{q}}$ et \mathbf{R} par les relations

$$h \begin{bmatrix} \mathbf{R}_i \\ \mathbf{R}_j \end{bmatrix} = \mathbb{H}_\alpha \mathbf{r}_\alpha \text{ et } \mathbf{u}_\alpha = \mathbb{H}_\alpha^t \begin{bmatrix} \dot{\mathbf{q}}_i \\ \dot{\mathbf{q}}_j \end{bmatrix} \quad (1.21)$$

avec

$$\mathbb{H}_\alpha = \begin{bmatrix} \mathbb{H}(\mathbf{q}_i) \\ \mathbb{H}(\mathbf{q}_j) \end{bmatrix}, \quad (1.22)$$

comme défini pour la méthode GEM. En utilisant (1.21) dans la première équation du système (1.19), nous obtenons une condensation de (1.19) aux points de contact,

$$\mathbf{u}_\alpha(i+1) = \mathbf{u}_{\alpha, libre}(i) + \sum_{\beta=1}^{n_c} \mathbb{W}_{\alpha\beta} \mathbf{r}_\beta(i+1) \quad \alpha \in \{1, \dots, n_c\}, \quad (1.23)$$

où $\mathbb{W}_{\alpha\beta} = \mathbb{H}_\alpha^t (\mathbb{M}^{-1})(i) \mathbb{H}_\beta$. Le calcul de $\mathbb{W}_{\alpha\beta}$ peut être réalisé par différentes méthodes que nous détaillerons plus tard. \mathbb{H} et \mathbb{H}^t sont des applications linéaires, \mathbb{H}^t est l'application transposée de \mathbb{H} , impliquant un changement de variables entre les niveaux local et global.

Remarque : par abus de langage et commodité, nous appellerons par la suite \mathbf{r}_α la force ou réaction au contact α et non impulsion moyenne.

1.5.2 Loi de contact en vitesse

Contact unilatéral

La méthode *NSCD* diffère des deux autres méthodes par, entre autre, le traitement implicite des forces de contact. Contrairement aux méthodes *smooth*, l'interpénétration est ici proscrite et le contact est traité par des relations unilatérales. Pour la suite nous omettrons l'indice α pour alléger les notations. Suivant les notations définies précédemment, la condition d'impénétrabilité se traduit par $\delta \geq 0$. Ne considérant pas de cohésion entre les particules, la composante normale de l'impulsion locale reste elle aussi positive se traduisant par $r_n \geq 0$. Cette grandeur est nulle lorsqu'il n'y a pas contact, autrement dit : $\delta > 0 \Rightarrow r_n = 0$. Ces deux relations sont alors condensées en le système suivant plus communément appelée *condition de Signorini* :

$$\delta \geq 0 \quad r_n \geq 0 \quad \delta r_n = 0 \tag{1.24}$$

que l'on peut écrire aussi grâce au formalisme de l'analyse convexe

$$r_n = \text{proj}_{\mathbb{R}^+}(r_n - \rho\delta), \quad \rho > 0. \tag{1.25}$$

Usuellement nous ne travaillons pas avec δ (distance ou écart de contact) mais avec la vitesse relative au contact $\mathbf{u} = (u_n, u_t)$. Ainsi si celle-ci existe (i.e. que le mouvement est assez régulier), on peut écrire la *condition de Signorini en vitesse* définie par la relation,

$$\begin{aligned} &\text{Pour } t_0 \text{ tel que } \delta(t_0) \geq 0 \\ &\forall t \in \text{I} \quad \delta(t) \leq 0 \Rightarrow u_n(t) \geq 0 \quad r_n(t) \geq 0 \quad u_n(t)r_n(t) = 0 \end{aligned} \tag{1.26}$$

Cette relation découle de deux points. Le premier est que si pour un temps t^0 , δ est positif ou nul, alors si pour $t \in \text{I}$ $\delta(t)$ est négatif ou nul alors la vitesse relative normale u_n à cet instant est positive. En effet si à un instant donné, deux corps adjacents ne s'interpénètrent pas et qu'à un instant suffisamment proche ils s'interpénètrent alors, pour satisfaire la condition d'interpénétrabilité, la vitesse relative au contact va devoir être positive pour faire en sorte que cette interpénétration deviennent nulle écartant ainsi les deux corps : cela se traduit par $u_n \geq 0$ (c.f. lemme de viabilité de Moreau [109]).

Loi de choc

Dans le cas de corps rigides, la *condition de Signorini* ne contient pas assez d'information pour modéliser le choc entre particules rigides. Les relations (4.12) et (1.26) ne décrivent pas totalement la physique de la collision (choc inélastique). Il nous faut donc compléter notre relation par une loi de choc. Par exemple la loi de restitution de Newton stipule que l'on peut relier les vitesses avant (-) et après (+) choc par la relation $u_n^+(t) = -e_n u_n^-(t)$. On peut citer d'autre loi de restitution comme la loi de poisson ou des lois énergétiques. Pour des corps à géométrie plus complexes que les corps circulaires ou sphériques, le coefficient de restitution normal e_n n'a pas alors plus vraiment de sens, la géométrie devant alors être pris en compte [92]. Un exemple souvent cité est celui du

bloc oscillant sur un plan horizontal qui transite par un statut de double contact pour lequel la loi de Newton ne permet pas de décrire le décollement de l'un des coins et reproduire ainsi le phénomène de basculement appelé "rocking" [71]. Notons aussi que pour des échantillons compacts impliquant des chocs multiples simultanés lors d'évolutions quasi-statiques ce coefficient n'a pas un grand rôle. Pour présenter une approche plus générale, nous utiliserons la notion de vitesses formelles au sens de Moreau ou vitesses pondérées [109] mettant en jeu une combinaison des vitesses avant et après choc,

$$\begin{cases} (1 + e_n)\bar{u}_n = u_n^+ + e_n u_n^- \\ (1 + e_t)\bar{u}_t = u_t^+ + e_t u_t^- \end{cases}, \quad (1.27)$$

e_t représentant le coefficient de restitution tangentiel, il est alors possible de traiter l'exemple de "rocking".

Frottement de Coulomb

Le frottement est traité en utilisant la loi de Coulomb classique stipulant que la norme de la composante tangentielle de l'impulsion de contact locale doit rester inférieure à un seuil, produit du coefficient de frottement μ par la composante normale de la dite impulsion : $\|r_t\| \leq \mu r_n$. Le coefficient de frottement entre particules, est a priori différent de la notion de frottement interne du milieu. Si la vitesse de glissement est non-nulle alors la force tangentielle est opposée à la vitesse de glissement et a une valeur égale à μr_n : $u_t \neq 0 \Rightarrow \mu r_n u_t / \|u_t\|$. De la même façon que pour le contact unilatéral, il est possible de formuler les deux relations précédentes en utilisant encore une fois le formalisme de l'analyse convexe

$$r_t = \text{proj}_{\mathcal{C}(\mu r_n)}(r_t - \rho u_t), \quad \rho > 0, \quad (1.28)$$

$\mathcal{C}(\mu r_n)$ étant la section du cône de Coulomb, le segment $[-\mu r_n, \mu r_n]$ dans une modélisation bidimensionnelle.

Grâce au formalisme introduit, nous avons maintenant tous les outils en main pour résoudre notre problème de contact local. Le couple d'équations (1.26) et (1.28) constitue la loi de contact-frottant de *Signorini-Coulomb*, appellation que nous serons amenés à utiliser par la suite.

1.5.3 Un solveur non régulier

La résolution locale est faite contact par contact par un algorithme de type Gauss-Seidel non-linéaire. Si on considère le contact α et qu'on suppose les réactions aux autres contacts fixées (par commodités l'indice de temps i sera omis). Le schéma itératif est alors défini de la façon suivante (itération $k + 1$) :

$$\begin{cases} \mathbf{u}_\alpha^{k+1} - \mathbb{W}_{\alpha\alpha} \mathbf{r}_\alpha^{k+1} = \mathbf{u}_{\alpha, \text{free}} + \sum_{\beta < \alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\beta^{k+1} + \sum_{\beta > \alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\beta^k \\ \text{Loi}[\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1}] = \text{vraie} \end{cases} \quad (1.29)$$

La résolution locale de ce problème consiste à trouver le couple $(\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1})$ satisfaisant le système de relations (1.29) où $\text{Loi}_\alpha[\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1}] = \text{vraie}$ exprime la loi de contact-frottant

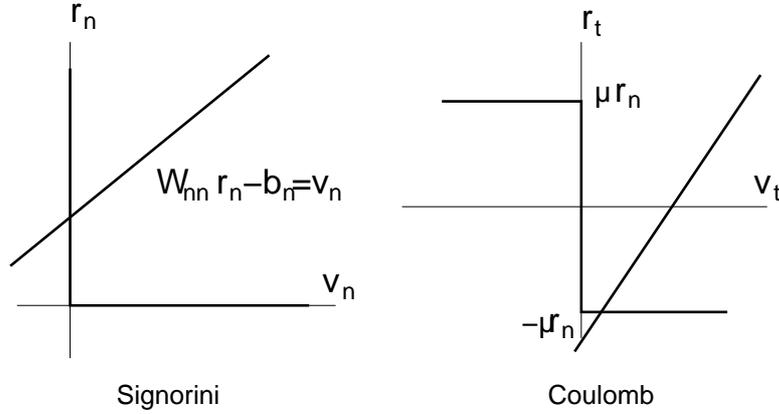


FIG. 1.6 – Graphes de Signorini et de Coulomb

(loi de *Signorini-Coulomb*) devant être satisfaite. Le membre de droite de la première équation de (1.29) sera noté $-\mathbf{b}_\alpha$. Pour une description bidimensionnelle, (1.29) peut être résolu par une intersection affine de graphe. Nous verrons que pour une description tridimensionnelle, objet du chapitre 5, nous utilisons un algorithme de type Newton généralisé (du à Alart et Curnier [6]). En dimension deux, la première équation de (1.29) à résoudre est la suivante

$$\begin{bmatrix} W_{nn} & W_{nt} \\ W_{tn} & W_{tt} \end{bmatrix} \begin{bmatrix} r_n \\ r_t \end{bmatrix} - \begin{bmatrix} u_n \\ u_t \end{bmatrix} = \begin{bmatrix} b_n \\ b_t \end{bmatrix} \quad (1.30)$$

Pour un problème bidimensionnel, la solution peut être obtenue de façon explicite en procédant à l'étude de cas présentée dans le tableau 1.5.3.

Cette résolution locale se fait contact par contact. Il s'avère que cette méthode est sensible à l'ordre de parcours des contacts. En effet une solution obtenue en traitant le contact α_1 avant le contact α_2 peut différer d'une solution obtenue dans l'ordre inverse. Ainsi une renumérotation peut influencer le nombre d'itérations et conduire à l'obtention de solutions différentes. Ce phénomène est dû au caractère singulier de l'opérateur de Delassus \mathbb{W} qui est semi-défini positif, sauf cas très particulier [37]. Notre système étant alors sous déterminé, l'algorithme peut nous faire prendre une direction plutôt qu'une autre dans l'espace des solutions admissibles. La figure 1.7 illustre ce phénomène. Nous avons réalisé une compression bi-axiale, à force imposée, sur un échantillon de 2304 disques avec et sans renumérotation (scramble) du parcours des contacts lors de la résolution. L'évolution du nombre d'itérations nécessaire à la convergence est en moyenne faiblement perturbée (figure 1.7). On peut cependant observer des écarts importants à certains instants. Ces écarts n'influent toutefois pas sur les propriétés macroscopiques de l'échantillon : la distribution des forces de contact dans l'échantillon (figure 1.8) est identique en fin de processus pour chaque simulation. Dans la zone des forces faibles ($F/F < 1$), les valeurs se situent autour d'une valeur constante dans la représentation semi-logarithmique (droite pointillée

Si $b_n \leq 0$	alors	$\mathbf{r} = \mathbf{0}$ statut : séparation
Sinon		
$b_n > 0$ $r_n > 0$		
On pose pour	$\epsilon = -1, 1$	$Df_{t_\epsilon} = W_{TN} + \epsilon\mu W_{TT}$ $Df_{n_\epsilon} = W_{NN} + \epsilon\mu W_{TN}$
Si $\epsilon(Df_{t_\epsilon}b_n - Df_{n_\epsilon}b_t) \leq 0$	alors	$r_t = \epsilon\mu r_n$, $r_n = b_n/Df_{n_\epsilon}$ statut : glissement ϵ
Sinon		$\mathbf{r} = \mathbb{W}^{-1}\mathbf{b}$ statut : adhérent

TAB. 1.2 – Résolution du problème local dans le cas bidimensionnel.

horizontale). Dans la zone des forces fortes, ($F / \langle F \rangle \gg 1$), les valeurs se situent autour d'une droite décroissante, avec des fluctuations pour les forces les plus fortes dû à une faible statistique. On retrouve ainsi le comportement mis en évidence par Radjai [125]. L'orientation des normales au contact est elle aussi identique en fin de processus (voir encart de la figure 1.8) dénotant dans ce processus une légère ovalisation dans la direction de la sollicitation.

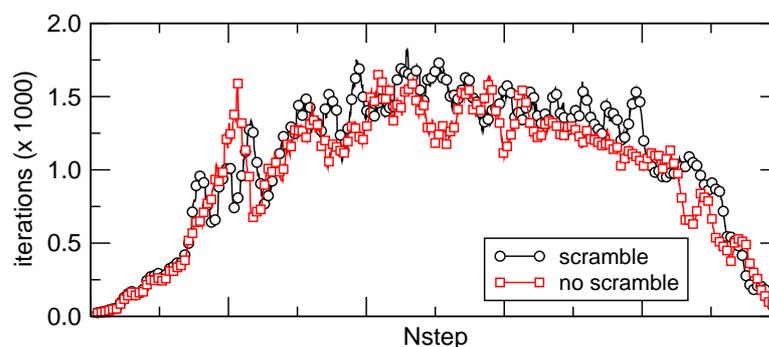


FIG. 1.7 – Evolution du nombre d'itérations lors d'un processus de dépôt sous pesanteur avec (scramble) et sans (no scramble) renumérotation des contacts.

Le cadre 1.3 présente l'algorithme synthétisant la méthode NSCD englobant les étapes présentées précédemment.

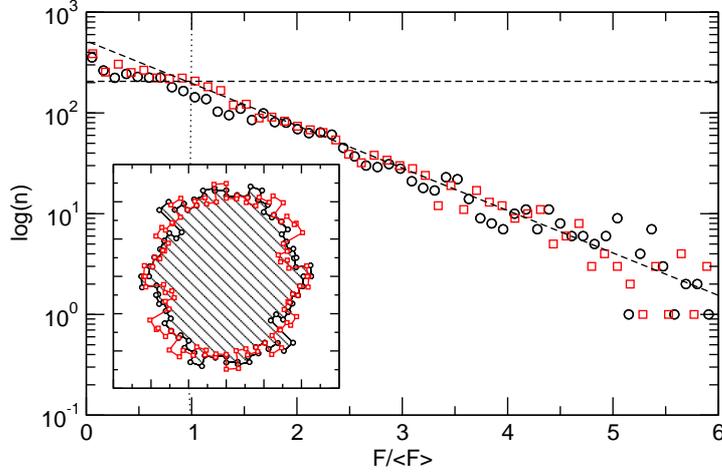


FIG. 1.8 – Distribution des forces de contact pour les simulations avec (scramble) et sans (no scramble) renumérotation des contacts dans une représentation semi-logarithmique. En encart orientation des normales au contact en fin de processus pour les deux méthodes.

$$\left[\begin{array}{l}
 i = i + 1 \text{ (pas de temps)} \\
 \text{Evaluation de } \mathbf{q}(i) \\
 \text{Détection des contacts} \\
 \text{Evaluation de } \dot{\mathbf{q}}(i)_{\text{libre}} \text{ i.e. de } \mathbf{u}_{\alpha, \text{libre}}(i) \quad \alpha = 1, n_c \\
 \left[\begin{array}{l}
 k = k + 1 \text{ (itération NLGS)} \\
 \left[\begin{array}{l}
 \alpha = \alpha + 1 \text{ (boucle des contacts)} \\
 \text{Evaluation de } \mathbf{b}_{\alpha} \text{ (membre de droite)} \\
 \text{Résolution du problème local, inconnues } (\mathbf{u}_{\alpha}, \mathbf{r}_{\alpha}) \text{ (via (1.29))}
 \end{array} \right] \\
 \text{Test de convergence (b)}
 \end{array} \right] (a) \\
 \text{Evaluation de } \dot{\mathbf{q}}(i + 1) \text{ (utilisant (1.19))}
 \end{array} \right.$$

TAB. 1.3 – Algorithme de l’approche NSCD en mettant en évidence le solveur non linéaire

1.5.4 Critère de convergence et qualité de solution.

Travailler avec des méthodes itératives nous oblige à faire un choix de critère de convergence, tâche qui n’est pas facile. La plupart du temps, ce test s’appuie sur les propriétés de convergence de l’algorithme permettant de borner la distance des itérés à la solution généralement inconnue. Hélas dans le cas des granulats, existence et unicité sont des propriétés se faisant désirer. Il nous faut alors faire un choix de critère, qui se doit d’être suffisamment précis pour que la solution en sortie ait un sens, mais qui ne doit pas être trop strict afin de garder des temps de calcul raisonnables. Dans les méthodes liées aux

éléments discrets certains critères d'erreur s'apparentent à des critères en loi de comportement, tel celui utilisé par De Saxé dans son approche par bi-potentiel [40]. Les critères peuvent s'apparenter également à des critères en stagnation, de distances en graphes ??, ... Ceux présentés ici ont été introduits par Jean [69, 70] et possèdent un aspect énergétique. A chaque itération du solveur Gauss-Seidel, le couple solution (\mathbf{r}, \mathbf{u}) obtenu ne satisfait pas les équations de la dynamique puisque, de par le principe itératif de l'algorithme de Gauss-Seidel, les couples locaux $(\mathbf{r}_\alpha, \mathbf{u}_\alpha)$ sont calculés avec des valeurs provisoires des réactions locales. La loi de *Signorini-Coulomb* est elle vérifiée. Pour déterminer l'écart à la solution nous allons effectuer, lors du test de convergence, une itération de Jacobi "à blanc" nous permettant de calculer une valeur temporaire à partir de termes portant le même indice d'itération. On définit alors le vecteur $\bar{\mathbf{r}}$ (resp. $\bar{\mathbf{u}}$) comme la demi-somme des itérations \mathbf{r}_{GS} (resp. \mathbf{u}_{GS}) et \mathbf{r}_{Jac} (resp. \mathbf{u}_{Jac}) obtenues respectivement avec Gauss-Seidel et Jacobi. La variation en vitesse relative est également calculée et notée $\Delta \mathbf{u}$ ($\Delta \mathbf{u} = \mathbf{u}_{Jac} - \mathbf{u}_{GS}$). On introduit ainsi plusieurs grandeurs,

$$\begin{aligned} S_{vr}^m &= \sum_{\alpha} \bar{\mathbf{r}}_{\alpha} \cdot \Delta \mathbf{u}_{\alpha} \\ S_{vv}^q &= \sum_{\alpha} \Delta \mathbf{u}_{\alpha} \cdot \Delta \mathbf{u}_{\alpha} \\ S_{vr}^q &= \sum_{\alpha} (\Delta \mathbf{u}_{\alpha} \cdot \Delta \mathbf{u}_{\alpha}) \cdot (\bar{\mathbf{r}}_{\alpha} \cdot \bar{\mathbf{r}}_{\alpha}) \end{aligned}$$

et

$$\begin{aligned} \mathcal{E}^m &= \sum_{\alpha} \bar{\mathbf{r}}_{\alpha} \cdot \mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha} \\ \mathcal{E}^q &= \frac{1}{N_{actif}} \sum_{\alpha} (\mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha}) \cdot (\mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha}) \end{aligned}$$

où \mathcal{E}^m et \mathcal{E}^q représentent des énergies de référence (N_{actif} est le nombre de contact transmettant une force non nulle), et on définit les trois critères suivant

$$\left\{ \begin{aligned} e_{vr}^m &= \frac{S_{vr}^m}{\mathcal{E}^m} \\ e_{vv}^q &= \sqrt{\frac{S_{vv}^q}{\mathcal{E}^q}} \\ e_{vr}^q &= \frac{\sqrt{N_{actif} \cdot S_{vr}^q}}{\mathcal{E}^m} \end{aligned} \right. \quad (1.31)$$

Dans les échantillons denses, une précaution supplémentaire est à prendre. En effet, les tests effectués étant des tests en moyenne sur le nombre de contacts, il se peut que certaines crises dynamiques locales ne soit pas perçues. Prenons l'exemple d'un échantillon où la globalité des contacts ne change pas d'un pas sur l'autre et où seulement une poignée de contacts changent violemment (figure 1.9). Dans ce cas, l'algorithme de Gauss-Seidel propageant lentement l'information, la crise locale peut se retrouver masquée par le mutisme de ses congénères et ainsi ne pas perturber le critère d'arrêt. L'algorithme itère peu et crée une accumulation de violations au sein de la cellule de crise. Afin de pallier ces accumulations, il devient nécessaire de faire itérer l'algorithme un minimum pour ne pas se retrouver en fin de simulation avec un échantillon non présentable. M. Jean propose un nombre minimum d'itération relié au nombre de contacts [26], efficace sur les systèmes denses. Sur les systèmes plus lâches, un tel critère est plus difficile à mettre en oeuvre, le

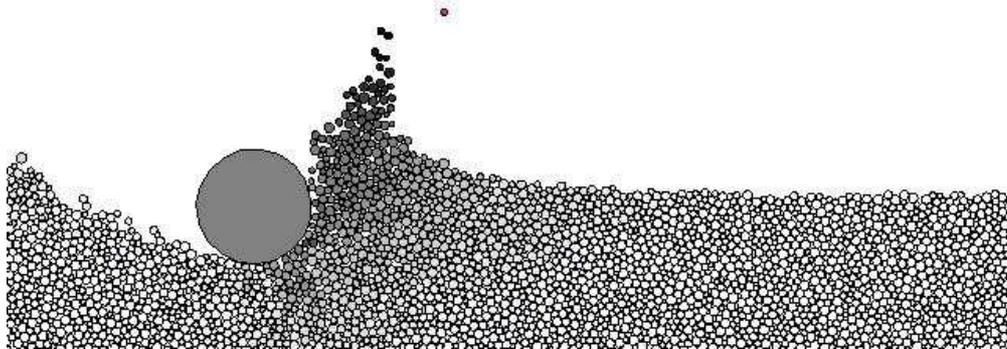


FIG. 1.9 – Simulation d’impact d’un corps rigides sur un lit meuble composé de 15 000 particules. Lors de l’impact, peu de contacts sont affectés mais les variations sont très grandes pouvant conduire à des violations importantes si des précautions supplémentaire concernant le critère de convergence ne sont pas prises.

nombre de contacts pouvant plus fortement évoluer. Il va de soi que ce choix appartient au modélisateur et rien qu’à lui, garant de sa simulation du temps t^0 au temps final. Notons que le test de convergence peut ne pas être effectué à chaque itération. On peut ainsi se fixer une valeur it_{check} séparant deux tests consécutif de convergence. De ce fait le nombre de calculs effectués est minimisé, permettant de ne pas pénaliser le temps de calcul.

1.6 Comparaisons numériques

Nous venons de mettre en évidence les points propres à chaque méthode. Toutefois il faut garder à l’esprit que celles-ci oeuvrent toutes dans un seul but : la modélisation des systèmes multi-contacts. Alors il est maintenant intéressant de voir quelles sont les différences au niveau des résultats numériques entre les trois méthodes. De telles comparaisons ont déjà été effectuées. Lanier et Radjai comparent les méthodes NSCD et MD [89] à des résultats expérimentaux, dans le but d’étudier les distributions de forces à l’intérieur d’échantillons constitués de particules circulaires. Les deux méthodes tendent vers les mêmes résultats. Nougier *et al.* [116] comparent lors de compressions bi-axiales les résultats obtenus via les méthodes NSCD et DEM. Les auteurs en arrivent aux mêmes conclusions poussant la comparaison à des simulations plus dynamiques. Les comparaisons entre ces différentes méthodes restent peu nombreuses dans la littérature ; on ne trouve pas de plate-formes ou codes de calcul possédant une modularité permettant de jongler entre deux implémentations, ceux-ci étant souvent dédiés à une seule approche (GEM, PFC, UDEC). Les comparaisons, lorsqu’elles sont effectuées, sont faites en terme de résultats mécaniques uniquement ; les comparaisons numériques apparaissant comme plus délicates. Ainsi après avoir introduit notre plate-forme logiciel, nous présenterons quelques résultats de simulations.

1.6.1 LMGC90 : plate-forme de développement

Pour nos recherches nous avons bénéficié du code de calcul *LMGC90* (Logiciel de Mécanique Gérant le Contact écrit en fortran 90) créé par M. Jean et re-développé récemment par F. Dubois [45]. Cette plate-forme logiciel, initialement conçue dans un esprit orienté objet, possède de nombreuses extensions, et permet assez facilement l'ajout de nouvelles fonctionnalités. Elle permet de traiter des problèmes de contacts aussi bien entre corps rigides qu'entre corps déformables. De nombreuses contributions dues à différents travaux de recherches ont contribué à son développement : outre les présents travaux citons ceux de G. Saussine [30, 139] liés au développement d'un code 3D pour corps polyédriques. A l'heure où ses quelques pages sont écrites les approches NSCD et MD sont intégrées dans le code pour des modélisations bi et tridimensionnelles. D'autres implémentations ont été réalisées, issues des chapitre 3 et 4 concernant les développements d'une version parallèle et d'un nouvel algorithme de résolution. La méthode GEM quant à elle n'est pas encore intégrée dans le code.

1.6.2 MD vs NSCD

Nous avons simulé la mise en pression d'un échantillon constitué de 2300 disques dont le rayon moyen est de 3 cm avec une variation de ± 1 cm (figure 1.10).

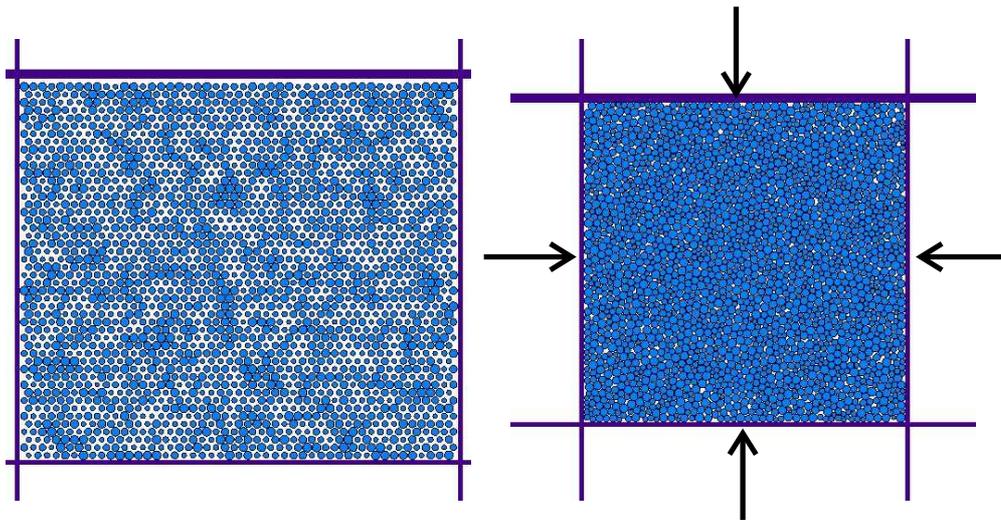


FIG. 1.10 – Echantillon au départ (gauche) puis après la mise en pression (droite)

Les particules sont initialement déposées sur une grille constituée de cellules triangulaires et compactées grâce à une pression exercée sur les quatre parois latérales croissante sur la première moitié de la simulation puis maintenue constante sur l'autre moitié. Le choix de cette simulation repose sur le fait qu'elle est nécessaire à la préparation d'un échantillon

et qu'elle peut conditionner l'évolution de son comportement. Les paramètres à caler pour les différentes simulations ne sont pas les mêmes.

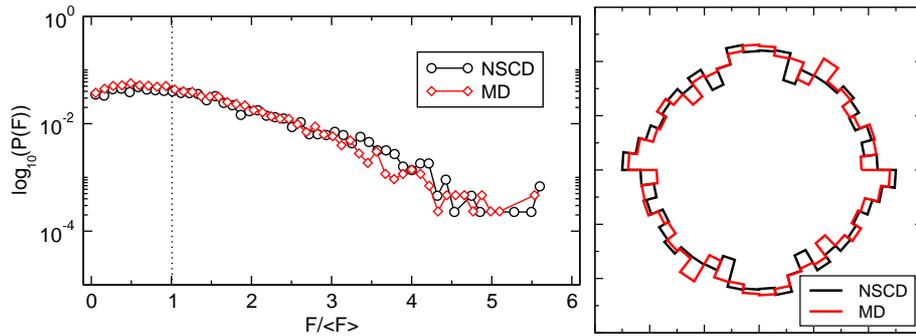


FIG. 1.11 – Distribution des forces de contact (gauche) et des normales au contact (droite) en fin de simulation.

Nous obtenons une bonne concordance entre les différentes quantités mesurées (nous renvoyons au chapitre suivant pour leur définition). Les distributions des orientations des normales au contact obtenues par les deux approches sont identiques (figure 1.11-droite) : on obtient une probabilité constante, presque indépendantes de l'orientation caractéristique des compressions de milieux polydispersés effectués sans gravité. Les distributions des forces de contact (figure 1.11-gauche) coïncident également. On retrouve des comportements distincts pour les deux sous réseaux (séparés par la ligne pointillée). On peut noter cependant quelques différences masquées par la représentation en \log_{10} . Pour les faibles forces on obtient une chute avec la méthode *MD* causée par la présence de forces négatives. On peut également observer quelques fluctuations pour les hautes valeurs de $F/\langle F \rangle$, mais ces fluctuations affectent les deux approches et sont liées à la faible statistique effectuée.

D'un point de vue numérique les simulations ne demandent pas les mêmes précautions, et les paramètres à ajuster sont différents. Par exemple le pas de temps utilisé avec *MD* est 100 fois inférieur à celui utilisé avec *NSCD* ($h_{MD} = 2 \cdot 10^{-5}$). Ainsi pour simuler le même procédé il faut 200 000 pas de temps avec *MD*, là où avec *NSCD* il n'en faut que 2000. Le coût d'une résolution *MD* (calcul des forces) est légèrement supérieur au coût d'une itération *NSCD* respectivement de $8 \cdot 10^{-3} s$ et de $6 \cdot 10^{-3} s$. Toutefois on passe, dans l'ensemble de la simulation, moins de temps dans la partie résolution avec *MD* (1530s) qu'avec *NSCD* (6570). Le facteur pénalisant en temps de calcul pour *MD* n'est pas le solveur mais la détection des contacts qui est effectuée 100 fois plus, ce qui fait au final une simulation moins longue avec *NSCD* qu'avec *MD*.

1.6.3 GEM vs NSCD

Le fait que *GEM* utilise des lois régularisantes apparaît être un point très important d'un point de vue numérique pouvant entraîner des différences dans les simulations. Pour illustrer ce phénomène, prenons un exemple mono-dimensionnel très simple : une balle élastique, coincée entre deux parois dotée, d'une vitesse initiale (voir figure 1.12).

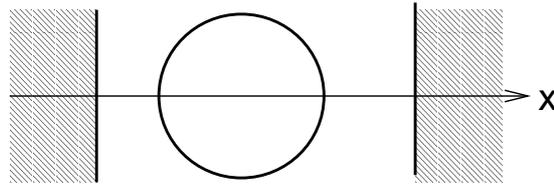
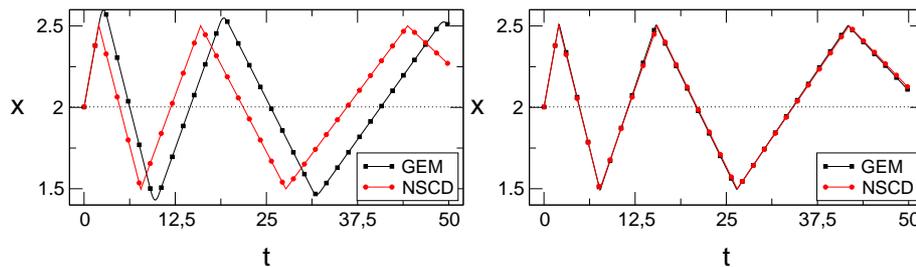


FIG. 1.12 – Exemple d'une balle coincée entre deux murs.

GEM traite donc les contacts entre la balle et les murs avec un modèle classique de ressort-amortisseur où la rigidité et la viscosité sont représentées respectivement par k_n et c . La méthode *NSCD* utilise une loi de choc avec restitution dont le coefficient est représenté par ρ_n . Pour de faibles valeurs de k_n , la pénétration est non négligeable, et il nous est alors impossible de superposer les courbes de déplacement obtenues avec les deux méthodes (voir figure 1.13 a)). Mais pour des rigidités plus grandes, nous aboutissons à des évolutions similaires (voir figure 1.13 b)). Pour une étude dynamique le coefficient

FIG. 1.13 – (left-a) $k_n = 2.10^4$ (right-b) $k_n = 2.10^6$

de restitution joue un rôle important. Pour *GEM*, la restitution dépend de k_n mais aussi fortement du coefficient de viscosité c , tandis que pour *NSCD* ce coefficient est caractérisé uniquement par ρ_n . La superposition des courbes (figure 1.13 b)) est obtenue pour $c = 2.10^4$ et $\rho_n = 0.715$ [131].

1.7 Un choix parmi d'autre

Au final ces trois méthodes apparaissent comme avoir de grandes différences. Nous pouvons toutefois les classer dans les deux classes précédemment citées : les méthodes "smooth" et les méthodes "non-smooth". Au Laboratoire de Mécanique et Génie Civil, cette seconde classe apparaît comme une méthode phare et fut l'outil numérique de nombreux prédécesseurs [47, 115, 124, 144, 147, 157]. Elle sera également la notre et ce pour les raisons suivantes :

- la méthode s'appuie sur le formalisme mathématique développé par Moreau et Jean, permettant une écriture rigoureuse des équations discrétisées.
- le contact est traité sans régularisation c'est à dire sans introduction de paramètres de rigidité locale de contact, difficilement ajustables sans références expérimentales.
- aucun terme d'amortissement n'est nécessaire pour stabiliser le schéma numérique, nous permettant de travailler avec des pas de temps raisonnables.

Quoi qu'il en soit ce choix reste propre à chaque modélisateur. Le point le plus important, valable dans chacune des situations, est qu'il faut garder à l'esprit que ces méthodes restent des modèles numériques et comme tous les modèles elles ont leurs avantages et leurs défauts. Il en est ainsi par exemple de la nécessité d'imposer un nombre minimal d'itérations, mentionnée en section 1.5.4 ; nous verrons que l'algorithme proposé au chapitre 4 n'impose plus une telle procédure.

Chapitre 2

Milieus granulaires : paramètres d'étude

En complément des observations expérimentales, les simulations numériques apparaissent comme un formidable outil pour accéder à des informations locales difficiles voire impossibles à isoler expérimentalement comme la carte complète des forces de contacts. Il paraît difficile d'analyser les caractéristiques locales de chaque grain et donc travailler avec des valeurs moyennes est primordial. Les informations obtenues localement sont soumises à un traitement statistique, nous permettant d'écrire des grandeurs macroscopiques à l'échelle du milieu ou d'une partie de celui-ci. On peut donc caractériser notre milieu en fonction de certains paramètres représentatifs, paramètres nous permettant suivant les études de comparer expériences et simulations. On trouve dans la littérature de nombreux travaux couplant ces deux types d'observations [88], nous en proposerons une dans le chapitre III liée aux écoulements granulaires. Pour notre part, outils statistiques et grandeurs moyennes nous permettront ainsi de valider nos algorithmes sur des processus avec de multiples trajectoires solutions pour chaque grain mais au comportement collectif similaire et relativement indépendant des paramètres numériques (pas de temps, critère de convergence, ...)

2.1 Compacité et nombre de coordination

La *compacité*, noté ν , permet de définir si un échantillon est dense ou lâche. Défini comme le rapport du volume occupé par les particules (V_p) sur le volume total (V_t), elle est préférée à l'indice des vides e_V couramment utilisé en mécanique des sols. Lors des essais conventionnels de la mécanique de sols [154] tels que les essais bi-axiaux ou de cisaillement simple, on peut caractériser la variation de compacité (ou variation de volume) au cours du temps, la reliant ainsi à la déformation de l'échantillon. On peut ainsi parler de dilatance ou contractance suivant que ν diminue ou augmente.

Le *nombre de coordination* d'une particule, ou *coordinnence*, est quant à lui le nombre de particules voisines qui transmettent des efforts de contacts. Le nombre de coordination

moyen d'un échantillon est alors défini par

$$z = \frac{1}{n_b} \sum_{i=1}^{n_b} z_i, \quad (2.1)$$

nous permettant de connaître la connectivité dans notre échantillon. Dans un milieu faiblement polydisperse, la moyenne sera proche de 4 en 2D et 6 en 3D.

Ces paramètres nous informent ainsi sur la densité de notre échantillon. Nous serons amené à calculer ces grandeurs lors des différents processus et ainsi analyser leur variation. Notons que c'est deux paramètres caractérisant tout deux la densité de l'échantillon ne sont pas corrélés au sens où l'on ne peut pas exprimer l'un en fonction de l'autre. La figure 2.1 trace l'évolution de z en fonction de la compacité. Les différents points de la courbe ont été tracés à partir de simulations de compactage dans une boîte de taille fixe. Chaque simulation met en jeu un nombre différents de particules tandis que les dimensions de la boîte restent inchangées.

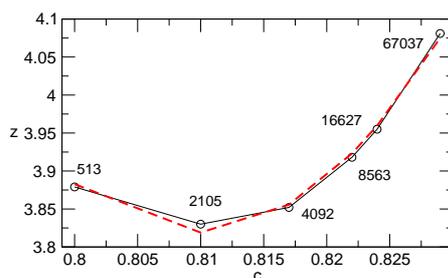


FIG. 2.1 – La compacité vue comme une fonction de z : les différents points sont obtenus pour un nombre croissant de particules. Le nombre de particules est reporté sur le graphe à coté du point correspondant.

Il est difficile de relier les deux paramètres à partir des points calculés. Une régression quadratique est effectuée sur les points de la figure 2.1 (pointillés), mais des points obtenus par d'autres simulations ne valident pas forcément cette approximation.

2.2 Tenseur de contrainte discret

On trouve de nombreuses définitions du tenseur de contraintes pour un milieu granulaire en équilibre ou quasi-équilibre [138]. Toutes ces définitions ont pour objectif d'approcher la définition du tenseur de Cauchy pour les milieux continus en utilisant différentes considérations théoriques (approche milieux continus, principe des puissances virtuelles). Une des premières définitions est celle de Drescher [44] qui définit la contrainte moyenne sur un échantillon de volume V soumis à n_e forces extérieures \mathbf{r}_α appliquées sur la frontière

du domaine aux points \mathbf{x}_α ($\alpha = 1, n_e$) par

$$\sigma_{ij} = \frac{1}{V} \sum_{\alpha=1}^{n_e} x_\alpha^i r_\alpha^j. \quad (2.2)$$

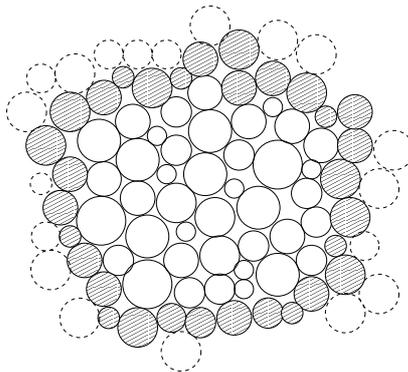


FIG. 2.2 – Portion de granulat. Les efforts de contacts transmis par les particules grisées sur l'intérieur du domaine qu'elles définissent correspondent aux forces extérieures appliquées sur ce même domaine.

Cette définition est similaire, dans le cas statique, à celle proposée par Moreau [106] qui introduit la notion de "moment interne". La définition (2.2) peut être appliquée à chaque particule de l'échantillon, les forces extérieures étant les forces de contacts appliquées à chaque particule. En utilisant le principe d'additivité on remonte au tenseur de contrainte moyen de l'échantillon. On peut ainsi écrire,

$$\sigma_{ij} = \frac{1}{V} \sum_{k=1}^{n_b} \sum_{\alpha=1}^{\alpha_k} x_{k,\alpha_k}^i r_{\alpha_k}^j, \quad (2.3)$$

α_k étant le nombre de contacts de la particule k , x_{k,α_k} le vecteur reliant le centre de la particule k au point de contact α_k . On aboutit ainsi à la définition donnée par Weber [152], Christoffersen [33] et Rothenburg [136] écrivant directement σ comme le produit tensoriel des forces de contact et des vecteurs inter-centres \mathbf{d} ,

$$\sigma_{ij} = \frac{1}{V} \sum_{\alpha=1}^{n_c} d_\alpha^i r_\alpha^j. \quad (2.4)$$

Toutefois les définitions (2.2), (2.3) et (2.4) ne sont équivalentes (et ont un sens!) que pour des évolutions quasi-statiques. Pour prendre en compte les efforts dynamiques, il faut étendre ces définitions. Avec l'introduction du tenseur des moments internes, Moreau [106] prend en compte les termes dynamiques pour donner une définition plus générale. Par la suite nous utilisons deux grandeurs rattachées au tenseur de contrainte : p et q représentant respectivement la moyenne des contraintes principales et la partie déviatrice (différence des contraintes principales divisée par deux).

2.3 Anisotropie et tenseur de fabrique

L'anisotropie, caractérisant une variation dans la statistique des orientations des normales au contact, est fortement corrélée à la micro-structure du milieu. Elle apparaît donc comme un paramètre interne d'une grande importance. La statistique des orientations des normales au contact, notée usuellement $p(\mathbf{n})$, est définie grâce à une moyenne sur tout les contacts du milieu considéré. La moyenne peut aussi être effectuée sur le nombre de particules et dans ce cas l'intégrale de $p(\mathbf{n})$ sur tout le milieu est égale au nombre de coordination moyen z ,

$$\int_0^{2\pi} p(\theta) d\theta = z. \quad (2.5)$$

Satake [117] propose une description de $p(\mathbf{n})$ via des tenseurs de fabrique d'ordre k , défini par le produit tensoriel de k coordonnées des normales au contact. Les travaux de Trodec [147] basé sur la texture locale des milieux granulaires permettent d'avoir une idée globale sur l'importance de l'anisotropie. Dans le cas d'évolution quasi-statique, il apparaît que le tenseur de fabrique d'ordre deux défini par

$$\mathbb{F} = \langle \mathbf{n} \otimes \mathbf{n} \rangle \quad (2.6)$$

est suffisant pour obtenir une première approximation de $p(\theta)$ donné par

$$p(\theta) = \frac{z}{2\pi} (1 + a \cos(2(\theta - \theta_p))), \quad (2.7)$$

où θ_p est la direction privilégiée des contacts (valeur propre principale de \mathbb{F}) et a l'anisotropie du matériau pouvant s'exprimer à partir de \mathbb{F} par la relation

$$a = 2 \frac{Dev(\mathbb{F})}{Tr(\mathbb{F})}. \quad (2.8)$$

Notons que le nombre de coordination moyen z est égal à $Tr(\mathbb{F})$. La statistique des orientations des normales au contact dépend alors de a , z et θ_p . On trouve dans [147, 148] une étude de ces différents paramètres ainsi que leurs prise en compte dans des modèles de plasticité.

2.4 Réseaux forts et faibles

La transmissions des efforts à l'intérieur d'un milieu granulaire dense est assuré par le réseau formé par les particules en contact. Malgré le caractère quasi statique de certain types de sollicitations ce réseau évolue avec la déformation du milieu tout en restant dense. Il n'est pas isotrope comme le souligne de nombreux travaux [126] et peu même être dissocié en deux sous-réseau : un faible et un fort caractérisant respectivement l'ensemble des forces de contact inférieures et supérieures à la force de contact moyenne $\langle F \rangle$. Il est possible de représenter la distribution des forces de contact normalisées dans des échantillons denses

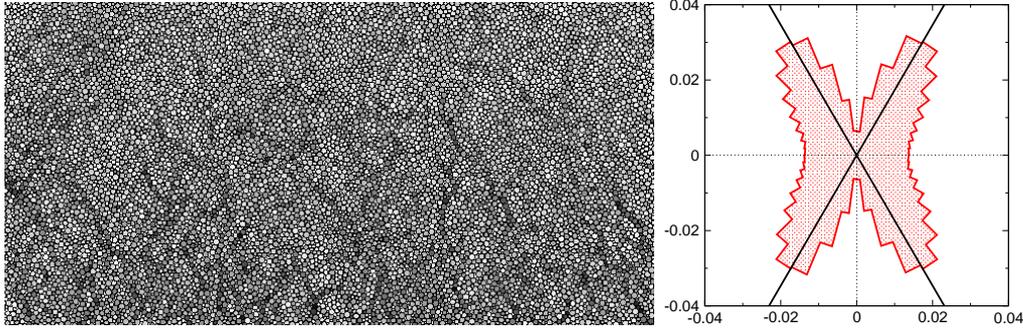


FIG. 2.3 – A droite, zoom sur un échantillon déposé sous pesanteur ($n_b = 67\,100$). A gauche représentation de la distribution des orientations des normales au contact (représentation polaire). Les axes $\pm 60^\circ$ sont mis en valeur correspondant aux orientations privilégiés lors d'un dépôt sous pesanteur.

soumis à différents types de sollicitation (cisaillement simple, essai biaxial) (c.f. figure 2.4). On peut alors vérifier que la distribution des forces respecte la loi suivante

$$P(F) = \begin{cases} P(1)F^\alpha & \text{si } F / \langle F \rangle < 1 \\ P(1)e^{\beta(1-F)} & \text{si } F / \langle F \rangle > 1 \end{cases} \quad (2.9)$$

On retrouve dans des études aussi bien numériques [112, 125] qu'expérimentales [118] des validations de ce modèle. La figure 2.4 représente les différentes distributions de force sur un échantillon de 9 216 polygones lors d'une compression bi-axiale pour 2.5, 5, 7.5 et 10%. On peut constater que $P(F)$ ne dépend pas du pourcentage de déformation et que $P(F)$ satisfait (2.9) pour $\alpha \approx -0.08$, $\beta \approx 1.24$ et $P(1) \approx 0.3$.

La force moyenne sépare les deux réseaux qui possèdent des propriétés différentes. Dans un milieu granulaire le réseau de contact est très hétérogène. Cette hétérogénéité se retrouve sur le comportement de ces deux sous-réseaux. Le réseau fort dont le squelette s'étend sur tout l'échantillon porte l'essentiel des efforts. D'un point de vue numérique, les statuts des contacts sont essentiellement adhérents. En revanche le réseau faible, plus important que le réseau fort [125], se retrouve séparé en réseaux locaux, prisonniers du réseau fort. Les statuts sont essentiellement glissants. Ainsi si l'on détermine le tenseur de fabrique et le tenseur de contrainte des deux sous réseaux, on va pouvoir caractériser les différents comportements. On indexera par la suite les différentes grandeurs macroscopiques par s pour le réseau fort (*strong network*) et w pour le réseau faible (*weak network*). Ainsi si l'on regarde notre exemple de compression bi-axiale, la figure 2.5 illustre l'évolution avec la déformation du rapport q/p (à gauche sur la figure) et l'anisotropie des deux sous-réseaux en fin de simulation (à droite sur la figure). Le rapport q/p du réseau faible est pratiquement constant lors de la simulation et reste à un niveau faible tandis que celui du réseau fort augmente avec la déformation jusqu'à atteindre un plateau dont le seuil est

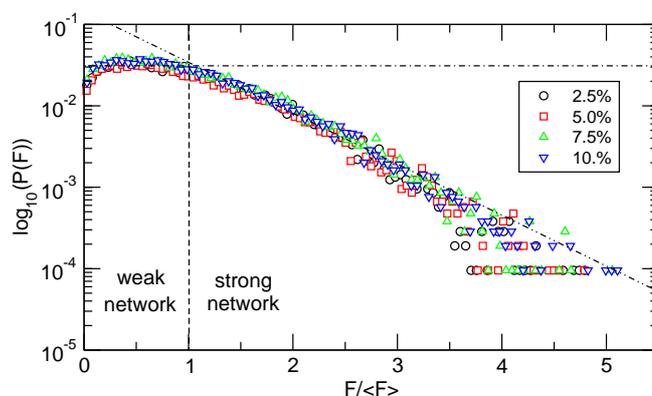


FIG. 2.4 – Distribution des forces de contact dans un échantillon de 9 216 polygones soumis à un essai de compression bi-axial à différent degré de déformation. La sollicitation n’influence pas la répartition la répartition des forces fortes et faibles. $P(F)$ est représenté en \log_{10} .

beaucoup plus important que les valeurs du réseau précédent.

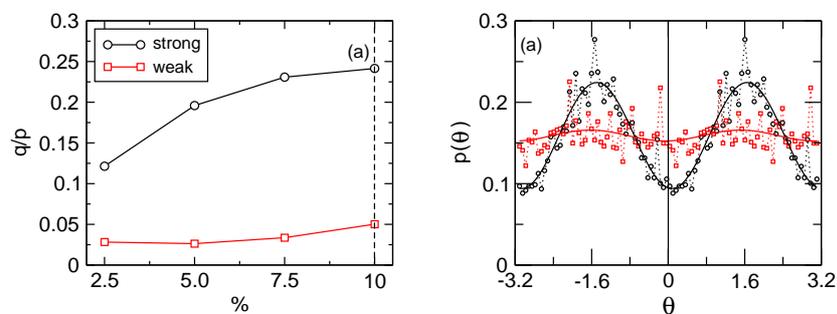


FIG. 2.5 – Evolution de q/p en fonction de la déformation (gauche) et distribution des orientations des normales au contact en fin simulation pour les deux sous-réseaux.

Nous avons procédé à deux *fit* pour $p_s(\theta)$ et $p_w(\theta)$ nous permettant de déterminer des valeurs pour z , a et θ_p respectivement de 2, 0.4 et 94° pour le réseau fort et 2, 0.04 et 86° . On retrouve ainsi la direction principale de sollicitation pour les deux sous-réseaux, une forte anisotropie dans le réseau fort et une faible anisotropie dans le réseau faible. On retrouve ainsi les caractéristiques classiques du réseau faible le rapprochant d’un milieu fluide.

2.5 Conclusion

D'autres grandeurs peuvent être utilisées, comme par exemple la longueur de corrélations ou le tenseurs de taux de déformations. Les différentes grandeurs macroscopiques introduites ici seront utilisées lors des différents tests numériques que nous serons amenés à effectuer dans les prochains chapitres. Ainsi en plus de valider les tests effectués, elles nous permettront d'exhiber également d'étudier les différents échantillons. Dans les deux chapitres qui suivent, portés sur l'optimisation du solveur non régulier, elles auront essentiellement un rôle de validation. Dans le chapitre lié au écoulements granulaires en tambour tournant, elles nous serviront en tant que paramètre d'étude.

Deuxième partie

Optimisation numérique

Chapitre 3

Implémentations et Multithreading

L'optimisation numérique des méthodes et logiciels dédiés à la simulation des milieux granulaires peut et doit prendre plusieurs voies parce que les besoins en réduction du temps de calcul sont considérables et d'origines diverses selon les applications. La simulation numérique discrète de tels milieux est en plein développement et concerne de plus en plus de domaines de recherche. Dans la plupart des cas elle a pour objectif de mieux comprendre les mécanismes sans pour l'instant prétendre à simuler la réalité dans sa complexité et notamment dans la représentation exhaustive de tous les grains d'un milieu réel. Ainsi, comme l'illustre M. Jean [26], 1 cm^3 de sable de plage fin ($0,1\text{ mm}$) contient environ 10^6 grains. Il est donc hors de question de simuler le comportement des dunes par un modèle numérique "exact". On introduit alors la notion d'échantillon, qui se veut représentatif. Pour certaines applications cet échantillon peut comporter plusieurs centaines de milliers de particules ; la simulation numérique des "boîtes à sable", qui sont elles-mêmes des simulations analogiques de plaques tectoniques [35, 86], exige un très grand nombre de grains si l'on souhaite accéder à la topologie des plis mis en évidence expérimentalement. Mais le nombre de grains n'est pas le seul enjeu ; certaines applications peuvent nécessiter un nombre de grains raisonnable mais en contre partie exiger une durée de simulation très importante parce que la dynamique impose des pas de temps très petits ou parce que les phénomènes étudiés ne se manifestent qu'après un temps long sans que l'on puisse faire l'impasse sur la simulation de l'ensemble du processus. Il en est ainsi des phénomènes de tassement dans les ballasts ferroviaires intervenant après un très grand nombre de cycles de chargement (passages de trains) [30].

Les méthodes d'éléments finis ont bénéficié depuis des décennies d'apports importants en optimisation numérique, parfois issus des résultats de mathématiques appliquées, notamment en algèbre linéaire (pour les solveurs) ou en recherche opérationnelle (pour les maillages), parfois développés spécifiquement comme les préconditionneurs (le plus spécifique étant EBE, Element By Element [66]), les méthodes de décomposition de domaines, ... Les méthodes d'éléments discrets peuvent parfois utiliser ces améliorations, mais le plus souvent ces dernières ne sont pas directement transférables ; on peut s'en ins-

pirer mais ce n'est pas toujours aussi efficace. La méthode des boîtes de Jean et Breikopf [22] s'inspire de la décomposition de domaine pour les milieux continus, mais au-delà des performances pas toujours au rendez-vous, une telle approche concerne des échantillons non soumis à des flux importants de grains, à moins de redéfinir très souvent les dites boîtes.

Face à la diversité des applications et des méthodes il convient donc d'identifier d'abord les étapes où l'on consomme le plus de temps calcul ; c'est l'objet de la première section. Ainsi on sait où porter les efforts d'optimisation et pour quelles applications on pourra attendre des gains conséquents. Ceci est d'autant plus important que les algorithmes itératifs mis en œuvre ne bénéficient pas pour l'essentiel de démonstration de convergence et a fortiori d'estimation de leur vitesse de convergence. D'une manière pragmatique, c'est-à-dire sans chercher à bouleverser ni les méthodes ni les structures mises en œuvre dans LMGC90, la suite du chapitre s'attache à améliorer les performances de l'algorithme de type Gauss Seidel non linéaire, solveur de base de l'approche Non Smooth Contact Dynamics, en testant différentes implémentations jouant sur le stockage et en procédant à une première parallélisation du solveur via une répartition des tâches sur plusieurs processus "voyant la même mémoire" (mémoire partagée) ; on parle alors de *multithreading*.

3.1 Analyse du temps de calcul

Avant d'attaquer toute optimisation, notre première tâche a été d'identifier les parties consommatrices en temps de calcul (comme la détection des contacts, la résolution ou encore l'actualisation des particules via le schéma d'intégration temporel). La part relative de la consommation du temps de calcul dépend fortement de la géométrie et des propriétés intrinsèques de l'échantillon, nous permettant ainsi d'exhiber une relation entre le temps *CPU* des différentes parties du code et les propriétés mécaniques de l'échantillon. Rappelons qu'un milieu granulaire se comporte selon les sollicitations comme un gaz (brassage et vol libre), un liquide (avalanche, écoulement, tambour tournant) ou un solide (compactage, tests de cisaillement et biaxial, sandbox) [67]. Trois exemples typiques ont été choisis pour illustrer l'influence de la dynamique sur le calcul : un compactage à surface libre, un écoulement en tambour tournant et un brassage de particules. Chaque échantillon est polydisperse et constitué du même nombre de particules, à savoir 1 000. Une loi de choc élastique est associée au problème de brassage tandis qu'une loi de choc inélastique est associée au deux autres simulations [26].

La figure 3.2 montre le réseau des contacts pour chaque situation. Ce réseau n'existe pas dans le cas du brassage, situation où le milieu est en agitation permanente. Chaque particule se déplace en vol libre entre deux impacts et les contacts lors des collisions sont essentiellement binaires. Par contre le réseau des contacts est beaucoup plus conséquent dans les deux autres situations mettant en jeu des échantillons denses. Le "squelette" du réseau des contacts est maintenu d'un pas de temps à l'autre, seul des réarrangements

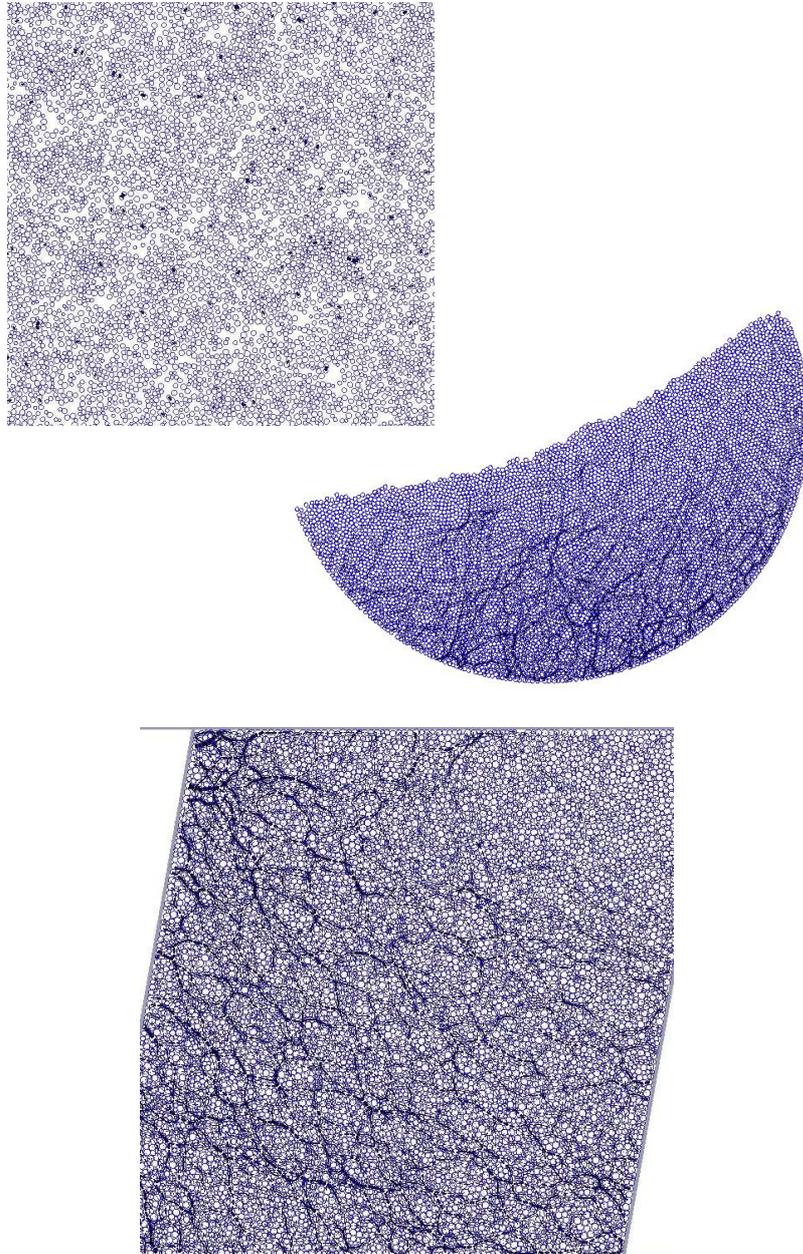


FIG. 3.1 – Trois exemples caractéristiques : en haut un brassage de particules, au milieu une simulation en tambour tournant et en bas un essai de cisaillement simple.

locaux sont effectués, excepté lors de crises dynamiques. Afin de déterminer le comportement de la méthode NSCD, pour chaque situation nous avons comptabilisé le temps *CPU*

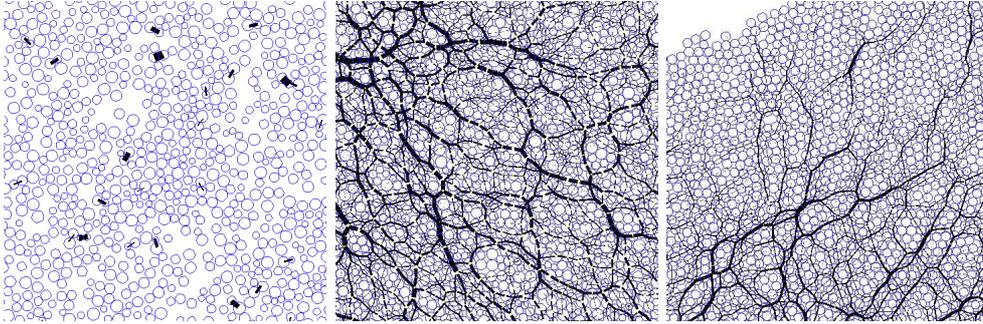


FIG. 3.2 – Zoom sur les trois exemples caractéristiques et visualisation du réseau des contacts : ils sont essentiellement binaires pour le brassage et s’organisent sous forme de réseau pour les deux autres échantillons, la simulation en tambour tournant gardant une phase liquide où les contacts restent binaires.

Problème	solveur	convergence	détection
brassage	18.6%	2.9%	47.44%
compactage	84.68%	2.43%	5.82%
tambour	85.68%	2,52%	1.89%

TAB. 3.1 – Répartition du temps *CPU* pris par les différentes sous-routines (%).

dans les différentes parties du code : la *détection* des contacts et l’étape de *résolution* locale (détermination des impulsions et vitesses libres). L’étape de *résolution* est décomposée en deux parties : la résolution elle-même (tableau 1.3, première colonne) et la partie test de *convergence* (tableau 1.3, deuxième colonne). Le pourcentage de temps écoulé donné dans le tableau 3.1 vient renforcer l’idée précédente : le solveur non linéaire consomme la majeure partie du temps *CPU* pour les deux échantillons denses tandis que la détection s’avère être la plus coûteuse en temps dans le cas du brassage. Ceci est également le cas des approches explicites telles que MD ou DEM. Pour de telle approche, la détermination des forces de contacts n’est pas coûteuse, pouvant être assimilée au calcul d’une seule itération. Ainsi le nombre de détection de contacts effectuées est égal au nombre de calcul des forces de contacts locales, contrairement à notre approche où le nombre de détections devient faible devant le nombre total d’itérations, plus particulièrement dans les simulations de systèmes denses. L’utilisation d’approches discrètes explicites nécessite en priorité une optimisation des algorithmes de détection des contacts (parallèles ou séquentiels). L’intérêt de l’approche NSCD réside dans la détermination rigoureuse des efforts de contacts (formulation en mesures différentielles et lois de contact et frottement non régularisées) nous permettant de disposer d’incrément de temps plus grand que les méthodes explicites. Ceci lui confère un fort degré de généralité pour résoudre une large gamme de problèmes, no-

tamment à contacts simultanés. La simulation des systèmes denses aux comportements hybrides (solide / liquide) constitue donc un challenge pour la mécanique granulaire où l'évolution des configurations exige une détermination la plus précise possible des efforts intergranulaires la pilotant.

3.2 Implémentation du solveur Gauss-Seidel non linéaire

L'approche NSCD utilise un algorithme de type Gauss-Seidel non linéaire (NLGS pour *Non Linear Gauss-Seidel*) par blocs pour résoudre le problème local [70] où les lois d'interactions locales sont le contact unilatéral et le frottement de Coulomb. Le principal avantage de cette méthode est sa convergence régulière rarement mise en défaut, mais qui peut être relativement lente. Nous verrons plus tard qu'elle présente a fortiori un handicap pour la parallélisation puisqu'elle est séquentielle dans sa formulation. Mais avant cela il convient de se pencher sur l'implémentation de l'algorithme lui-même et notamment la gestion de la matrice \mathbb{W} dont la partie hors bloc diagonale n'intervient que dans le second membre du problème non linéaire (et même non régulier) local du chapitre précédent,

$$\begin{cases} \mathbf{u}_\alpha^{k+1} - \mathbb{W}_{\alpha\alpha}\mathbf{r}_\alpha^{k+1} = \mathbf{u}_{\alpha,free} + \sum_{\beta<\alpha} \mathbb{W}_{\alpha\beta}\mathbf{r}_\beta^{k+1} + \sum_{\beta>\alpha} \mathbb{W}_{\alpha\beta}\mathbf{r}_\beta^k \\ r_n = \text{proj}_{\mathbb{R}^+}(r_n - \rho_n\delta), \quad \rho_n > 0 \\ r_t = \text{proj}_{\mathcal{C}(\mu r_n)}(r_t - \rho_t u_t), \quad \rho_t > 0 \end{cases} \quad (3.1)$$

On voit là qu'il n'est pas forcément nécessaire de stocker la matrice entière ; les matrices $\mathbf{W}_{\alpha\beta}$ non nulles suffisent ; on peut même s'en passer. Notons ici une différence avec les matrices à gérer dans la méthode des éléments finis qui sont obtenues par assemblage de matrices élémentaires : \mathbb{W} est la concaténation des matrices $\mathbb{W}_{\alpha\beta}$ et non leur assemblage. Le seul usage des matrices $\mathbf{W}_{\alpha\beta}$ s'apparente aux approches *matrix free* [60] pour les éléments finis sans être tout à fait la même chose. Ici donc on peut même se passer de calculer les matrices $\mathbf{W}_{\alpha\beta}$ ($\beta \neq \alpha$) ; c'est le but de la première technique ELG. L'autre procédure, SDL, calcule, et stocke, les matrices $\mathbf{W}_{\alpha\beta}$. La structure du code LMGC90, utilisant des modules propres à FORTRAN 90, amène à les dénommer en fonction des niveaux qu'elles mettent en jeu : le niveau global est celui des corps où sont gérés les configurations et les torseurs des efforts exercés sur chaque solide, le niveau local est celui des contacts où les grandeurs sont exprimées dans un repère local. Les deux paragraphes qui suivent sont dédiés à ces deux implémentations précisant leurs avantages et inconvénients : la méthode **E**change entre les niveaux **L**ocal et **G**lobal (ELG) et la méthode **S**tockage des **D**onnées au niveau **L**ocal (SDL).

3.2.1 Méthode ELG (sans assemblage)

La première technique d'implémentation consiste à ne construire que les matrices blocs $\mathbb{W}_{\alpha\alpha}$ sur la diagonale de l'opérateur de Delassus : les matrices $\mathbb{W}_{\alpha\beta}$ ne sont pas calculées. Le calcul des matrices $\mathbb{W}_{\alpha\alpha}$ ne se fait qu'une seule fois avant le processus itératif. Ainsi le calcul du second membre \mathbf{b}_α de l'équation (3.1), ne s'effectue pas par produits matrice-vecteur au niveau local mais par des échanges entre les niveaux local (contacts) et global

(grains) via les opérateurs $\mathbb{H}_\alpha, \mathbb{H}_\alpha^T$. Ainsi à chaque itération et pour chaque contact, nous sommes amenés avant la résolution locale à calculer un terme auxiliaire, noté \mathbf{u}_α^{aux} , prenant en compte les torseurs des efforts appliqués aux particules impliquées dans le contact α (ici indicées par j et l),

$$\text{Pour } \alpha = jl \quad \mathbf{u}_\alpha^{aux} = \mathbb{H}_\alpha^T[(\mathbb{M}_j)^{-1}\mathbf{R}_j^k - (\mathbb{M}_l)^{-1}\mathbf{R}_l^k] \quad (3.2)$$

Une fois ce terme calculé nous sommes capables de déterminer \mathbf{b}_α , de résoudre notre problème local et d'obtenir ainsi le couple $(\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1})$. Les torseurs des particules sont ensuite actualisés via l'opérateur \mathbb{H}^α ,

$$\begin{bmatrix} \mathbf{R}_j \\ \mathbf{R}_l \end{bmatrix}^{k+1} = \begin{bmatrix} \mathbf{R}_j \\ \mathbf{R}_l \end{bmatrix}^k + \mathbb{H}_\alpha(\mathbf{r}_\alpha^{k+1} - \mathbf{r}_\alpha^k) \quad (3.3)$$

et on passe ensuite au traitement du contact suivant. L'algorithme de résolution via la méthode ELG est détaillé dans le tableau 3.2.

<p>Nouveau pas de temps Evaluation de $\dot{\mathbf{q}}^{libre}$ ainsi que $\mathbf{u}_\alpha^{libre}$ ($\alpha = 1, nc$) (o) Calcul des matrices bloc diagonales $\mathbb{W}_{\alpha\alpha}$</p> <div style="border-left: 1px solid black; border-right: 1px solid black; padding: 0 10px;"> <p>$k = k + 1$ (iteration NLGS)</p> <div style="border-left: 1px solid black; border-right: 1px solid black; padding: 0 10px;"> <p>$\alpha = \alpha + 1$ (indice de contact)</p> <p>(i) Calcul de \mathbf{u}_α^{aux} en identifiant les deux corps en contact : $\alpha = jl$ et $\mathbf{u}_\alpha^{aux} = \mathbb{H}_\alpha^T[(\mathbb{M}_j)^{-1}\mathbf{R}_j^k - (\mathbb{M}_l)^{-1}\mathbf{R}_l^k]$</p> <p>(ii) Evaluation de \mathbf{b}_α (membre de droite) $\mathbf{b}_\alpha = -\mathbf{u}_\alpha^{libre} - \mathbf{u}_\alpha^{aux} + \mathbb{W}_{\alpha\alpha}\mathbf{r}_\alpha^k$</p> <p>(iii) Résolution du problème local (3.1), les inconnues étant $(\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1})$.</p> <p>(iv) Actualisation de \mathbf{R}_j et de \mathbf{R}_l $\begin{bmatrix} \mathbf{R}_j \\ \mathbf{R}_l \end{bmatrix}^{k+1} = \begin{bmatrix} \mathbf{R}_j \\ \mathbf{R}_l \end{bmatrix}^k + \mathbb{H}_\alpha(\mathbf{r}_\alpha^{k+1} - \mathbf{r}_\alpha^k)$</p> </div> <p>Test de convergence pour $k = 0 \dots k_{max}$ Evaluation de $\dot{\mathbf{q}}^{i+1}$ grâce à (1.19).</p> </div>

TAB. 3.2 – Algorithme de la méthode ELG : les échanges local-global sont effectués lors des étapes (i) et (iv).

<p>nouveau pas de temps Evaluation de $\dot{\mathbf{q}}^{libre}$ ainsi que $\mathbf{u}_\alpha^{libre}$ ($\alpha = 1, nc$) (o) Calcul des matrices blocs $\mathbb{W}_{\alpha\beta}$</p> <div style="border-left: 1px solid black; border-right: 1px solid black; padding: 0 10px;"> <p>$k = k + 1$ (iteration NLGS)</p> <div style="border-left: 1px solid black; border-right: 1px solid black; padding: 0 10px;"> <p>$\alpha = \alpha + 1$ (indice de contact)</p> <p>(i) Evaluation de \mathbf{b}_α (membre de droite) $\mathbf{b}_\alpha = -\mathbf{u}_\alpha^{libre} - \sum_{\beta < \alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\alpha^{k+1} - \sum_{\beta > \alpha} \mathbb{W}_{\alpha\beta} \mathbf{r}_\alpha^k$</p> <p>(ii) Résolution du problème local (3.1), les inconnues étant $(\mathbf{u}_\alpha^{k+1}, \mathbf{r}_\alpha^{k+1})$.</p> </div> <p>Test de convergence pour $k = 0 \dots k_{max}$</p> </div> <p>Evaluation de $\dot{\mathbf{q}}^{i+1}$ grâce à (1.19).</p>

TAB. 3.3 – Algorithme de la méthode SDL : l'étape (o) prend une part de temps plus importante que dans la méthode précédente, mais reste en proportion faible devant la partie solveur elle même.

3.2.2 Méthode SDL (avec assemblage)

La seconde technique d'implémentation consiste à éviter le dialogue entre les deux niveaux (contacts-particules) lors de la résolution du problème 1.29. Pour cela il faut avoir localement toute la structure de la matrice. Ainsi en plus des matrices blocs $\mathbb{W}_{\alpha\alpha}$, les matrices $\mathbb{W}_{\alpha\beta}$ sont elles aussi calculées avant de commencer les itérations de Gauss-Seidel. Notons que les matrices $\mathbb{W}_{\alpha\beta}$ nulles ne sont ni calculées ni stockées : nous réduisons ainsi considérablement la taille mémoire, nous rapprochant de l'espace pris par un stockage Morse. Le calcul du second membre \mathbf{b}_α se fait pour chaque contact par un simple produit matrice-vecteur et une fois le traitement d'un contact effectué lors du processus itératif, aucune information n'est à transmettre au niveau des particules. L'algorithme de résolution via la méthode SDL est explicité dans le tableau 3.3.

3.2.3 Que choisir ?

Les techniques ELG et SDL diffèrent en trois points : les étapes (o), (i) et (iv) (c.f. tableaux 3.2 et 3.3) mis en évidence sur la figure 3.3. L'étape préparatoire (o) de la méthode SDL est plus riche, donc plus coûteuse en temps CPU. Un rapide calcul nous permet de prévoir que, pour un échantillon dense ($z \simeq 4$), un contact possède 7 contacts voisins, soit 7 matrices de plus à calculer, et donc que la préparation serait 8 fois plus coûteuse ; ce rapport pouvant être diminué en bénéficiant de la symétrie de la matrice \mathbb{W} . Lors des itérations Gauss-Seidel, les étapes (i) et (iv) sont elles aussi différentes, la méthode SDL s'affranchissant de cette dernière étape (i.e. l'étape (iv)).

Une étude comparative a été réalisée entre les deux techniques d'implémentation afin d'évaluer le surcoût des différentes étapes exposées jusqu'à présent. Le paramètre guidant

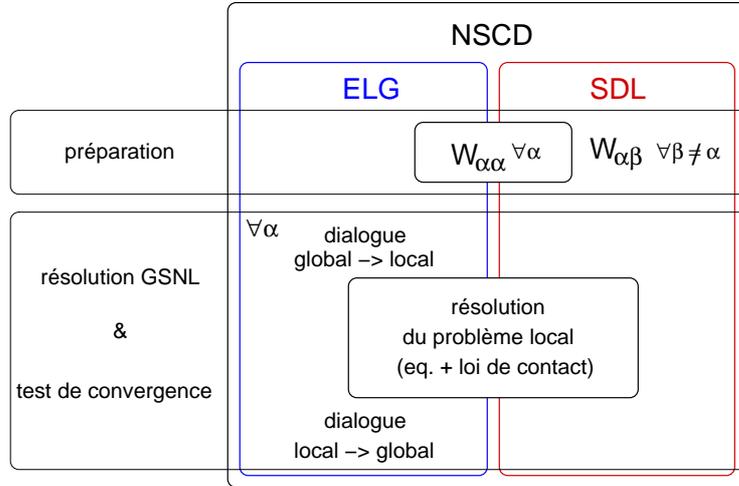


FIG. 3.3 – Mise en évidence des principales différences entre les deux structures de NSCD : la version ELG favorise une minimisation de l’espace mémoire et du temps de préparation, tandis que la version SDL favorise un gain de temps lors du calcul des itérés au détriment de l’espace mémoire.

nos simulations est le nombre de particules. Nous avons simulé des processus de dépôt sous pesanteur sur quelques pas de temps pour différents diamètres de billes (la taille de l’échantillon reste fixe - $1m \times 3m$). Le nombre de contacts dans l’ensemble de nos simulations est de 1 070, 4 100, 8 200, 17 100, 33 500, 70 400 et 137 040. Les deux techniques d’implémentation sont testées sur chaque échantillon. La figure 3.4 représente l’évolution du rapport de différentes grandeurs en fonction du \log_{10} de n_c : Ts correspond au rapport du temps d’une itération ELG et du temps d’une itération SDL, Tt correspond au temps total de la méthode ELG sur celui de la méthode SDL et $\langle it \rangle$ le nombre d’itérations ELG moyen sur le nombre d’itérations SDL moyen (encadré droit de la figure 3.4). Les simulations ont été réalisées sur deux types de calculateurs (SUN et SGI) afin d’évaluer l’influence du calculateur.

Pour les deux types d’architectures, la méthode SDL reste plus performante que la méthode ELG sur l’ensemble des simulations. Toutefois on peut noter une différences de valeur pour Ts suivant que l’on soit sous SUN ou SGI, les valeurs obtenues sous SGI sont supérieures à celles obtenues sous SUN. Pour les simulations effectuées sous SUN, pour de petits échantillons ($n_c < 4\,000$), la méthode SDL apparaît comme étant 1.8 fois plus rapide que la méthode ELG. Ce rapport chute pour des échantillons plus conséquents ($n_c > 100\,000$) pour atteindre une valeur de 1.2 : la méthode ELG reste encore plus performante. Sous SGI, le rapport Ts a moins de fluctuations et reste proche de 1.6 avec un pic proche de 2, pour des valeurs de n_c voisines de 10 000. Si l’on observe le rapport du nombre d’itérations moyen des méthodes ELG et SDL, on s’aperçoit qu’il est relativement constant : il peut subir toutefois des crises, cas de l’échantillon à 17 100 contacts sous SUN où le solveur a

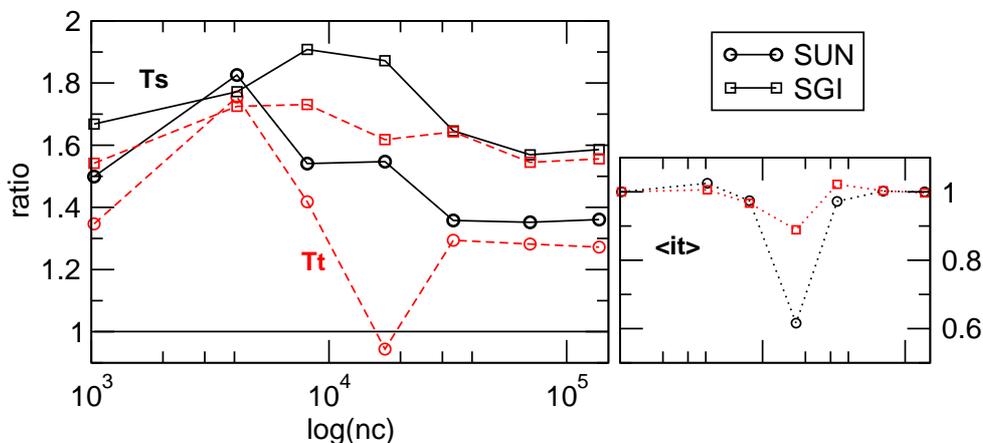


FIG. 3.4 – Rapport du temps des simulations ELG et SDL en fonction du \log_{10} du nombre de contacts : T_s est le ratio du temps relatif passé dans le solveur, T_t est le ratio du temps total. En encart, rapport du nombre moyen d'itérations obtenu avec les des deux techniques d'implémentations.

du mal à converger, ce qui se répercute ensuite sur le temps total de la simulation (graphe $ratio(\log(nc))$ de la figure 3.4).

Sur la gamme d'étude, la méthode SDL apparaît donc comme étant la plus efficace. Toutefois l'évolution du ratio nous laisse supposer que pour des échantillons encore plus conséquents (10^5 , 10^6 particules, ...) les deux méthodes pourraient devenir équivalentes, voire la méthode ELG pourrait devenir plus performante que la méthode SDL. Notons que ce constat est bien entendu tributaire de la machine utilisée puisque le facteur pénalisant pour la méthode SDL est la place mémoire. Pour des calculateurs possédant une capacité mémoire importante, on peut s'attendre à ce que le ratio observé sur la figure 3.4, évolue en faveur de la méthode SDL pour de plus grands échantillons ; le phénomène de *swap*, "écriture sur disques des données" étant d'autant plus réduit que la mémoire disponible est importante. Il faut aussi noter qu'une préparation SDL est plus coûteuse qu'une préparation ELG. Ainsi à temps équivalent dans la partie solveur, il devient plus intéressant d'utiliser l'implémentation ELG.

Les échanges entre les niveaux effectués dans la version ELG entraînent un surplus d'opérations par rapport à la version SDL. Ceux-ci engendrent donc plus d'arrondis, ce qui peut conduire à de très légères variations sur la solution d'un pas de temps sur l'autre. Ces variations ici sans conséquence peuvent s'accumuler au cours d'un processus et ainsi faire diverger les solutions obtenues pour les deux implémentations surtout au voisinage de crises. Celles-ci ne sont que très rarement effacées ; leur apparition peut-être simplement décalée dans le temps (c.f. figure 3.5). Cependant suivant l'évolution du processus, une crise peut apparaître en fin de simulation pour une méthode et pas pour l'autre, perturbant ainsi le nombre d'itérations (cas de la valeur singulière dans l'encart de la figure 3.4).

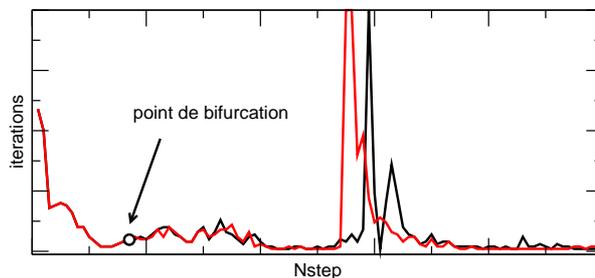


FIG. 3.5 – Séparation des courbes de convergence des implémentations SDL et ELG due à l’accumulation d’erreurs numériques et d’arrondis.

3.3 Multithreading

3.3.1 Bref tour d’horizon

Le calcul parallèle connaît un essor depuis les années 70 – 80 où les premières architectures parallèles voient le jour. Historiquement, les premières machines parallèles étaient des réseaux d’ordinateurs, proches des clusters de PC que l’on rencontre maintenant. Face aux demandes de grandes puissances de calcul, le parallélisme s’avère être un formidable moyen de faire chuter le temps *CPU* et ainsi traiter des problèmes de plus en plus importants et de plus en plus réalistes et qui nous semblait encore inaccessibles il y a quelques années. Actuellement on peut trouver de nombreux codes de mécanique des fluides ou de calcul de structures capable d’exploiter plusieurs centaines de processeurs permettant de donner des résultats très précis. Les machines parallèles peuvent être composées de quelques processeurs (moins d’une dizaine) jusqu’à plusieurs centaines voire milliers (on parle alors de parallélisme massif). Actuellement la technologie parallèle possède de gigantesques calculateurs : le *Earth Simulator* situé à Tokyo (Japon) [146] (depuis 2002 il occupe la première place au niveau mondial des super-calculateurs avec ces $40 Tflops \cdot 10^{12}$ opérations par seconde), l’*ASCI Q* du LANL (Los Alamos National Laboratory) (USA) ou encore les machines du laboratoire SANDIA situé à Albuquerque (USA) [153] (voir <http://www.top500.org>). Parmi les nombreux domaines de recherches utilisant cette technologie nous pouvons citer la chimie moléculaire, la bio-chimie, la mécanique des fluides, l’aérospatiale, l’aéronautique ou encore la météorologie, qui souvent sont les plus demandeur en temps *CPU*.

Les calculateurs appartiennent à deux grandes classes d’architectures : les architectures ”SMP” (**S**ymmetric **M**ulti **P**rocessor) (appelée encore *UMA* (**U**niform **M**emory **A**ccess)) et les architectures *NUMA* (**N**on **U**niform **M**emory **A**ccess). Pour les architectures *UMA*, le temps d’accès à la mémoire est uniforme pour chaque processeur, il n’y a pas de fragmentation de celle-ci (découpage). Pour les architectures *NUMA*, le temps d’accès n’est plus

uniforme (d'où le "N") et la mémoire est fragmentée (distribuée). Sur ces deux types d'architectures peuvent alors se développer différents modèles de programmation parallèle : les modèles à mémoire partagée (*UMA* seulement) et les modèles à mémoire distribuée (*UMA & NUMA*).

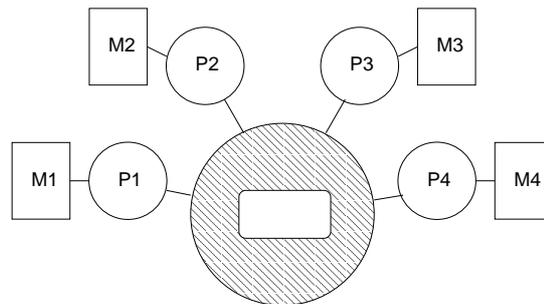


FIG. 3.6 – Modèles à mémoire partagée

Les modèles à mémoire distribuée consistent à distribuer la mémoire globale sur chaque processeur. Les codes utilisant ce genre de modèle utilisent le langage *MPI* (**M**essage **P**assing **I**nterface). Chaque processeur effectue une partie des calculs localement sans s'occuper des calculs que font ses voisins. C'est alors à l'utilisateur de gérer les échanges d'informations entre les différents processeurs lorsque ceux-ci sont nécessaires. Ceci implique souvent une restructuration assez importante de certaines parties du code. Langage largement utilisé en calcul parallèle, *MPI* apparaît comme le langage le plus performant en vue d'une utilisation massive de processeurs. Toutefois l'interconnexion doit être la plus réduite possible pour optimiser le temps de calcul.

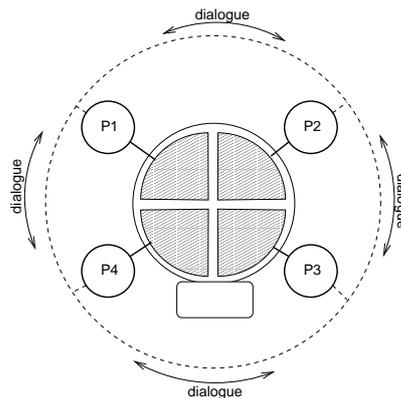


FIG. 3.7 – Modèles à mémoire distribuée

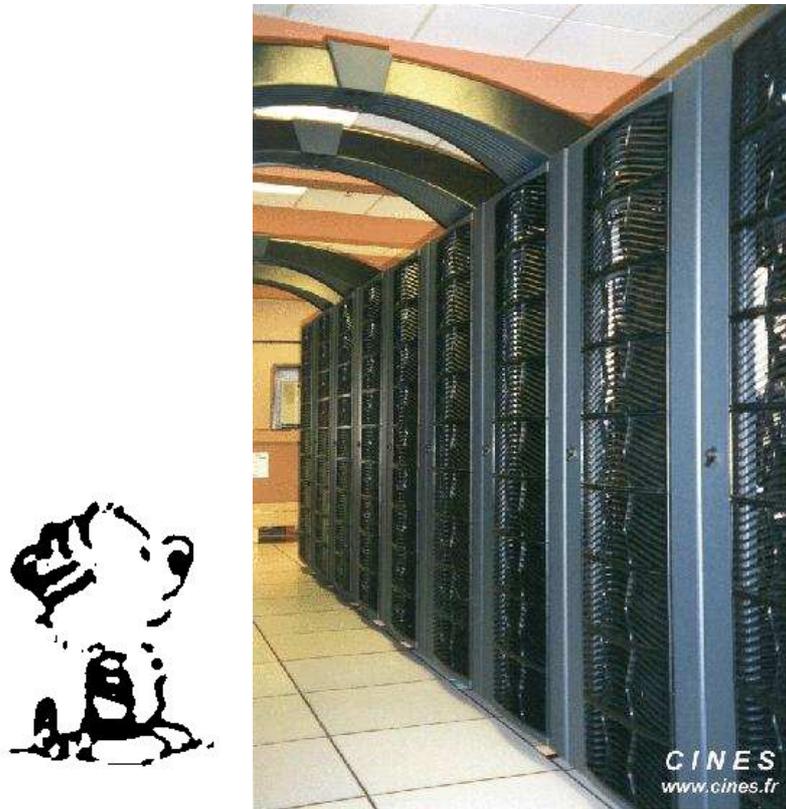


FIG. 3.8 – SGI O3800 (CINES - FRANCE), 768 processeur R14000/500 MHz

Le principe des modèles à mémoire partagée, moins répandus que leurs confrères, est que tous les processeurs du calculateur voient la mémoire globale, moyennant des caches locaux propres à chaque processeur. Ainsi aucun échange de mémoire n'est nécessaire entre processeurs puisque à tout instant chacun d'eux a accès à la mémoire globale. Les processeurs travaillant tous sur la même mémoire il faut rester vigilant quant au possible conflit d'accès mémoire entre différents processeurs : l'actualisation d'une donnée par deux processeurs simultanés fait que l'une des informations est perdue. Le langage utilisé par les codes de calcul basés sur ce principe est *OpenMP*. *OpenMP* est un ensemble de directives reconnues par les différents compilateurs qui s'intègrent dans le code sans en changer (ou faiblement) la structure. L'utilisation est ensuite transparente, l'utilisateur n'ayant à gérer uniquement le nombre de processeur à sa disposition. Il peut aussi définir des variables privées à chaque processeur, souvent des variables temporaires, pour effectuer des calculs locaux. Toutefois cette simplicité d'utilisation fait que les codes utilisant *OpenMP* ne peuvent pas prétendre à faire efficacement du calcul intensif avec plus d'une centaine de processeurs même avec une implémentation parallèle rigoureuse.

3.3.2 Mais pourquoi OpenMP ?

L'approche NSCD utilise classiquement un solveur de type Gauss-Seidel non linéaire qui possède une structure typiquement séquentielle, et donc ne favorise pas l'implémentation d'une version parallèle. Cependant une première technique développée par Bretkoff et Jean [22] consiste à découper le milieu en "boites" sur chacune desquelles travaille un processeur. Des échanges d'informations sont à gérer entre les boites afin d'actualiser les efforts transmis d'une boite sur ses voisines. Cette approche s'apparente à une méthode de décomposition de domaine [93], largement répandue en calcul des structures dans le contexte d'une modélisation par éléments finis. Cependant c'est sur des problèmes linéaires (ou linéarisés dans un processus de résolution non linéaire) que les techniques de décomposition de domaine ont montré leur efficacité maximale. Or ici la non linéarité est au cœur du solveur granulaire. Les tests effectués par Bretkoff et Jean montrent la difficulté de transférer une approche issue du calcul des structures dans le domaine des milieux granulaires : les échanges entre les boites et les difficultés à équilibrer les charges de calcul entre les processeurs viennent perturber le solveur et réduire l'efficacité de la parallélisation. De plus dans une telle approche il convient de procéder au découpage, de définir les interfaces ainsi que d'actualiser régulièrement les sous domaines et les interfaces au cours d'un calcul avec toutes les difficultés techniques attenantes. Nous avons donc opter pour une approche plus simple afin de mettre en œuvre une version parallèle de l'algorithme *NLGS*. Plus précisément, on considère la matrice \mathbb{W} de notre problème. Si les longueurs de corrélation des contacts restent raisonnables, autrement dit si la largeur de bande de la matrice est petite devant le nombre de contacts (et surtout devant le rapport $\frac{n}{P}$, où P désigne le nombre de processeurs), alors lors de l'itération en cours, en répartissant les contacts sur les différents processeurs, le calcul des impulsions locales affectées au processeur P_i ($i \in \{1, \dots, P\}$) ne sera pas perturbé par les calculs des impulsions locales effectués par les processeurs P_j ($j \neq i$). Pour rester au plus précis d'un algorithme de type Gauss-Seidel (séquentiel), les impulsions locales doivent être actualisées dès quelles ont été calculées. Un modèle à mémoire partagée est donc essentiel. Nous avons donc opté pour l'utilisation des directives *OpenMP* à l'intérieur du code de calcul pour écrire une première version parallèle de l'approche *NSCD* utilisant un solveur *NLGS*. Le principe s'avère être le même que celui effectué sur le solveur Gauss-Seidel linéaire [129], que nous allons rappeler en quelques mots et illustrer par le tableau 3.4.

Le schéma de parallélisation consiste à découper la boucle des contacts en P portions de tailles égales, P étant le nombre de processeurs (procédure de multithreading). Chaque processeur va ainsi effectuer les calculs des impulsions locales des contacts qui lui sont rattachés. Une fois cette impulsion calculée, elle est alors automatiquement visible par tous les autres processeurs nous affranchissant ainsi d'un échange d'information entre processeurs. Ce découpage s'opère très simplement grâce aux directives *OpenMP* qui viennent s'ajouter au début et en fin de boucle des contacts. La seule difficulté consiste à définir correctement les variables qui sont partagées par tous les processeurs ou propres (privées) à chacun d'eux. Notons que ce schéma conduit à une lecture différente de la boucle des contacts, lecture aléatoire et non reproductible (figure 3.9) ; un contrôle de l'ordre de lecture des contacts viendrait pénaliser le temps CPU.

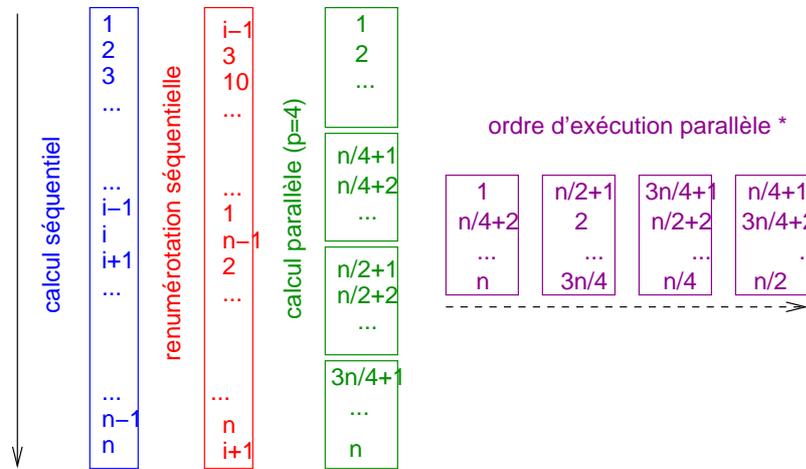


FIG. 3.9 – L’application du multithreading conduit à un parcours différent de l’ordre des contacts. Cette lecture n’est pas reproductible et complètement aléatoire.

Cette simple mise en oeuvre soulève toutefois trois questions. La première est de savoir ce qu’il advient lorsque la largeur de bande est supérieure à $\frac{n_c}{p}$, puisque dans ce cas un processeur peut être en train de calculer l’impulsion du contact α_i alors que l’impulsion du contact α_{i-1} est toujours inconnue. Une étude préliminaire a montré une faible perturbation de l’algorithme linéaire [129] et on peut donc espérer le même comportement pour le cas non-linéaire. La seconde est lié au problème lui-même. N’ayant pas à chaque instant t une solution unique, la permutation de l’ordre des contacts peut nous conduire à l’obtention d’une autre solution appartenant à l’espace des solutions admissibles. Comparer alors deux solutions apparaît comme délicat. Toutefois on peut s’attendre à ce que les propriétés macroscopiques du matériau (ou leur évolution) soient respectées lors d’une simulation. Enfin la dernière concerne l’implémentation de la méthode (SDL ou ELG). En effet si la méthode SDL, ne travaillant qu’au niveau local, n’est pas perturbée par le multithreading, il n’en est pas de même pour la méthode ELG. Les échanges entre les niveaux local et global effectués après résolution peuvent entraîner un phénomène dit de *”race condition”* lié au fait que deux processeurs vont aller écrire simultanément dans la même zone mémoire. Ce conflit intervient lors de l’actualisation des torseurs des efforts appliqués à chaque particule. Une particule intervient le plus souvent sur plusieurs contacts. Si deux contacts impliquant une même particule, gérés par deux processeurs différents doivent actualiser simultanément le torseur des efforts sur la particule commune aux deux contacts, l’une des deux valeurs ne sera pas prise en compte. La probabilité que le phénomène de *”race condition”* se produise reste toutefois faible mais ne peut être négligées.

Dans l’étude numérique qui suit nous évaluons, via une comparaison des temps de calculs et du nombre d’itérations, l’influence du multithreading sur l’efficacité de l’algorithme

```

Nouveau pas de temps :  $t = t + \Delta t$ 
[ Mouvement prédictif
[ Détection des contacts
  [  $k = k + 1$  (iteration NLGS)
    !$OMP PARALLEL PRIVATE(...) SHARED(...) ...
    !$OMP DO ...
      [  $\alpha = \alpha + 1$  (indice de contact)
        Evaluation de  $\mathbf{b}_\alpha$  (membre de droite)
        Résolution du problème local (3.1),
        Actualisation si nécessaire
      !$OMP END DO
    !$OMP END PARALLEL
  [ Test de convergence pour  $k = 0 \dots k_{max}$ 
[ Correction du mouvement

```

TAB. 3.4 – Insertion des directives *OpenMP* lors de la parallélisation de l’algorithme NLGS. A l’intérieur de la zone parallèle il est nécessaire de définir les variables propres à chaque processeur (**PRIVATE(...)**) et celles communes à l’ensemble des processeurs (**SHARED(...)**).

(perturbation ou non du nombre d’itérations), sur le mode d’implémentation, mais aussi sur la qualité de la solution en regardant l’évolution des différentes grandeurs macroscopiques caractérisant le milieu.

3.3.3 Analyse des tests

Méthodologie

Nous discutons dans cette partie des résultats obtenus sur différentes simulations : des compactages à surface libre et des compressions bi-axiales. Les compactages à surface libre ont été réalisés avec des échantillons de 1 016 et 10 251 disques déposés sous pesanteur dans une boîte ; la composition est de 95% de disques ayant un rayon moyen de $0.01m$ et de 5% égal à $0.02m$. Pour chaque disque, la densité volumique est de $580kg.m^{-3}$. Après le dépôt, les murs latéraux sont mis en mouvement dont la vitesse est décrite par l’équation suivante :

$$|v_x| = \frac{1}{60} \left(1 - \cos\left(\frac{\pi \cdot t}{30}\right) \right).$$

Ce processus est réalisé sur 10 000 pas de temps ($h = 6.10^{-2}s$). Le coefficient de frottement disque/disque est égal à 0.3 tandis que celui avec les parois est égal à 0.5. Les compressions bi-axiales comportent 9 216 disques polydispersés dont le rayon moyen est $0.03 \pm 0.01 m$. Après une compression uniforme, on anime les parois horizontale d’une vitesse uniforme

déformant ainsi notre échantillon : la hauteur de l'échantillon est diminué de 10%. Le processus s'étend également sur 10 000 pas de temps ($h = 3.10^{-3}s$).

Analyse de performance

Une étude préliminaire a été réalisée, utilisant l'implémentation ELG, sur les compactages à surface libre tendant à identifier l'influence de la largeur de bande. On utilisera les notations suivantes : *DAF* pour un compactage avec frottement, *DSF* pour un compactage sans frottement et *DOSF* pour un compactage sans frottement et en modélisant les parois par des collections de disques. Ce dernier type de simulations nous permet ainsi d'avoir une plus faible largeur de bande qu'avec la simulation *DSF*.

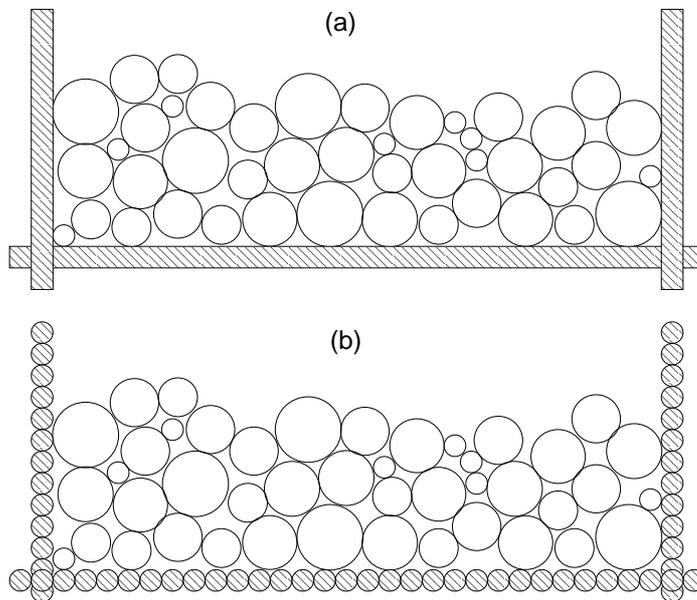


FIG. 3.10 – Echantillons utilisés lors des simulations *DAF*, *DSF* (a) et *DOSF* (b). L'utilisation de collections de particules pour la modélisation des parois assure une faible connexité entre les contacts.

On notera T_s le temps CPU passé dans le solveur, $\langle it \rangle$ le nombre moyen d'itérations, Pa la *performance algorithmique* correspondant au rapport du nombre d'itérations moyen du calcul séquentiel sur le nombre moyen d'itérations du calcul parallèle (i.e. $Pa = \langle it \rangle_{seq.} / \langle it \rangle_{par.}$). Les simulations ont été réalisées sur un ordinateur *SunFire 880* utilisant six processeurs *UltraSparc III 750 MHz/1 Go*. Les temps simulations sont donnés dans le tableau 3.5.

Le tableau 3.5 montre que le solveur *NLGS* est faiblement perturbé par la parallélisation. Le nombre d'itérations supplémentaires ne dépasse pas les 13% (dans le cas de la simula-

Processeur(s)	1		2		4		6	
	Ts	$\langle it \rangle$	Ts	Pa	Ts	Pa	Ts	Pa
DAF	21 531	335	10 535	1.018	5 425	1.009	3 874	1.027
DSF	28 722	382	14 174	0.962	7 782	0.888	5 780	0.876
DOSF	37 000	441	18 064	0.997	9 483	1.006	6 945	0.988

TAB. 3.5 – Analyse de performance pour des simulations réalisées avec 1,2,4 et 6 processeurs.

tion DSF avec 6 processeurs). On peut mettre même moins d'itérations (DAF) quand les parois sont modélisées par une couche de billes ; le rapport Pa reste très proche de 1. On peut relier cette faible perturbation à la petite largeur de bande, puisque la même simulation utilisant un seul corps rigides pour modéliser les parois conduit à une plus grande perturbation de Pa . La méthode ELG pouvant entraîner des accès simultanés d'écriture mémoire, les simulations de type DOSF permettent de réduire considérablement ces probabilités et ainsi préserver une stabilité du nombre d'itérations. Toutefois l'utilisation de parois composées de corps rigides entraîne une augmentation du nombre de contacts et donc, comme le montre par ailleurs le tableau 3.5, une augmentation du temps de calcul. D'un point de vue modélisation mécanique elles conduisent aussi à introduire une certaine rugosité des parois. Ainsi pour la suite, nous garderons le premier type de modélisation quitte à accepter une légère perturbation du solveur Gauss-Seidel.

Pour apprécier l'efficacité du solveur parallèle, nous avons évalué en plus de la performance algorithmique, la performance du calculateur. Pour cela nous utilisons un *speed-up relatif* S_P pour P processeurs défini par :

$$S_P = Pa_P \frac{T_{seq}}{T_{SP}}, \quad (3.4)$$

où T_{SP} représente le temps de calcul passé dans le solveur pour P processeurs ($T_{seq} = T_1$). Pa_P correspond à la performance algorithmique pour P processeurs. Si $S_P = P$, la parallélisation est pleinement efficace. Pour compléter notre étude sur la parallélisation du solveur *NLGS*, nous avons également réalisé des simulations sur *SGI Origin 3800* utilisant des processeurs *R14000/500 MHz*. Les tests parallèles ont été réalisés avec les deux techniques d'implémentation (SDL et ELG). La figure 3.11 montre l'évolution du speed-up S_P en fonction de P pour les problèmes tests de compactage à surface libre : à droite les résultats obtenus sur *SUN* et à droite ceux obtenus sur *SGI*.

Le premier résultat encourageant à l'observation des figures de gauche et de droite, est la non dépendance de la méthode de parallélisation par rapport à l'architecture machine malgré le fait que l'efficacité des directives *OpenMP* soit fortement liée à l'implémentation machine des compilateurs. Cette implémentation "dure" ne vient pas perturber nos résultats, pouvant apprécier les mêmes comportement pour les différents types de simulations. Les courbes de la figure 3.11 illustrant le Speed-Up relatif montrent que l'implémentation ELG possède un meilleur comportement en parallèle que l'implémentation SDL, la courbe

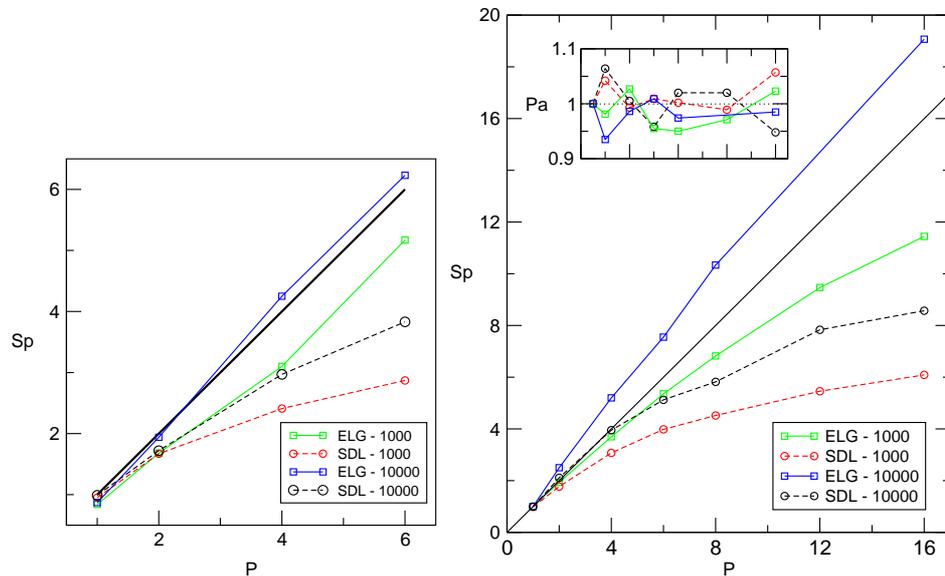


FIG. 3.11 – Speed-up obtenu sur *SUN* (gauche) et sur *SGI* (droite) avec les techniques d’implémentation pour deux tailles d’échantillons.

se rapprochant de la droite de comportement optimal ($S_P = P$) pour ELG et s’en écartant pour SDL. De plus pour les deux techniques d’implémentations, le speed-up est plus important lors de simulations importantes (10 000 disques), ce qui nous encourage pour des échantillons conséquents à augmenter le nombre de processeurs : ce n’est pas le cas pour les petits échantillons pour lesquels le calcul parallèle ne s’avère pas intéressant.

Quelque soit l’architecture, nous pouvons également observer une évolution super-linéaire pour l’implémentation ELG pour 10 000 disques. Ce phénomène est le cumul de plusieurs points particuliers nécessitant un rappel sur le calcul de S_P . Le temps de référence est le temps pour une simulation réalisée avec un processeur avec un code compilé pour une utilisation parallèle. Lors d’un calcul parallèle, le programme bénéficie de plus de mémoire à sa disposition (mémoire séquentielle $\times P$), ainsi pour un calcul dépassant la limite disponible, la gestion de la mémoire peut venir perturber les performances. Ainsi lors du calcul séquentiel pour un échantillon de taille conséquente, cette limite mémoire peut-être atteinte est ainsi perturber le temps de référence, ce qui conduit à ce phénomène de comportement super-linéaire. En prenant pour temps de référence le temps de la simulation avec deux processeurs, ce phénomène disparaît. L’implémentation ELG devant effectuer des calculs stockés à différents niveaux du code (contrairement à l’implémentation SDL), le temps d’accès mémoire vient s’ajouter également au phénomène précédent.

La figure 3.13, qui vient en complément de la figure 3.4, compare les deux techniques d’implémentation lors d’une exécution parallèle sur un échantillon de 10 000 particules.

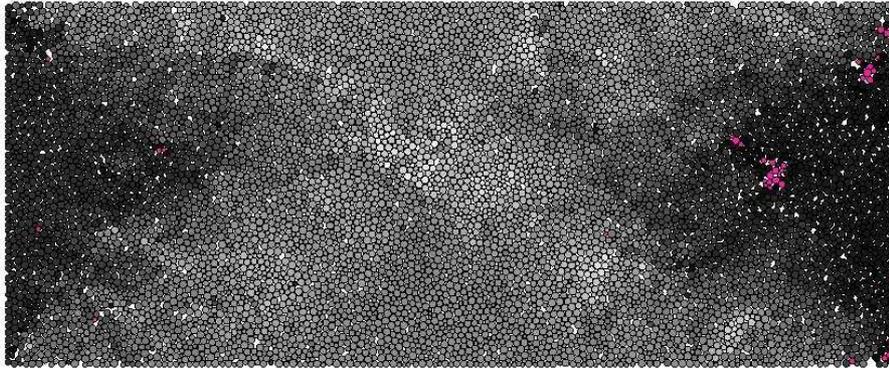


FIG. 3.12 – Incrément de déplacement entre deux instants lors d'un essai de compression biaxiale. Le code de couleur varie du blanc (faibles variations) au noir (fortes variations). On observe les bandes de cisaillement localisées à gauche et à droite de l'échantillon.

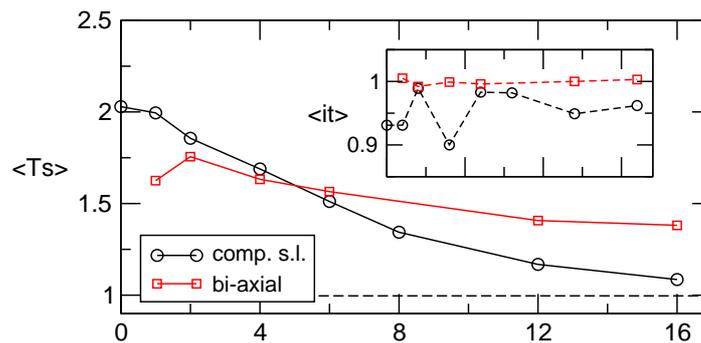


FIG. 3.13 – Rapport du temps passé dans le solveur pour une utilisation parallèle des deux techniques d'implémentation (ELG et SDL). En encart, rapport du nombre moyen d'itération en fonction de l'utilisation du nombre de processeurs.

On retrouve ainsi le comportement observé la figure 3.4 pour une exécution séquentielle, à savoir un temps plus raisonnable avec l'implémentation SDL. Ce gain en temps pour le compactage à surface libre est voisin de 2 pour une exécution séquentielle et chute avec l'augmentation du nombre de processeurs pour atteindre un rapport proche de 1.1 avec 16 processeurs. Pour l'essai bi-axial, la décroissance est moins prononcée partant d'une valeur séquentielle de 1.7 pour atteindre une valeur de 1.4 pour 16 processeurs. On peut observer également grâce à l'encart de la figure 3.13, que la performance algorithmique est constante pour l'essai bi-axial (quasiment égal à 1) et très faiblement perturbée pour le compactage, n'engendrant ainsi pas de grandes différences entre les deux techniques d'implémentation au niveau du nombre moyen d'itérations. A la vue de l'évolution de Ts plusieurs conclu-

sions semble se poser : l'augmentation du nombre de processeur va en faveur de la méthode ELG, mais sur l'ensemble des simulations effectuées, la méthode SDL reste plus efficace. Pour des calculs parallèles plus massifs ($P > 20$, par exemple), en prenant en compte la phase de préparation, la méthode ELG peut devenir plus intéressante.

3.3.4 Influence sur la solution

Il nous reste à voir de quelle façon les solutions diffèrent puisque les solutions obtenues avec un nombre différent de processeurs ne sont pas toujours, voire jamais identiques, en terme de distributions des grains et de profil d'échantillons. L'évolution d'un milieu granulaire est un procédé erratique possédant plusieurs chemins possibles pour une sollicitation donnée : pour chaque pas de temps, la distribution des forces de contact n'est pas unique. Comme nous l'avons signalé précédemment, la façon dont est parcourue la liste des contacts va déterminer une solution parmi l'ensemble des solutions admissibles. Cependant, nous devons trouver des critères de comparaisons pour nos différentes solutions afin de voir si le calcul parallèle préserve les propriétés de nos échantillons. C'est justement sur ce point précis que nous avons basé nos comparaisons, les milieux granulaires faisant souvent appel à des grandeurs statistiques telle que l'anisotropie et la distribution des orientations des normales aux contacts. Donc en plus de comparaisons numériques (évolution des itérations), nous avons comparé l'évolution de l'anisotropie dans le cas de compactages à surface libre et nous avons tracé l'évolution du rapport q/p pour des compressions bi-axiales.

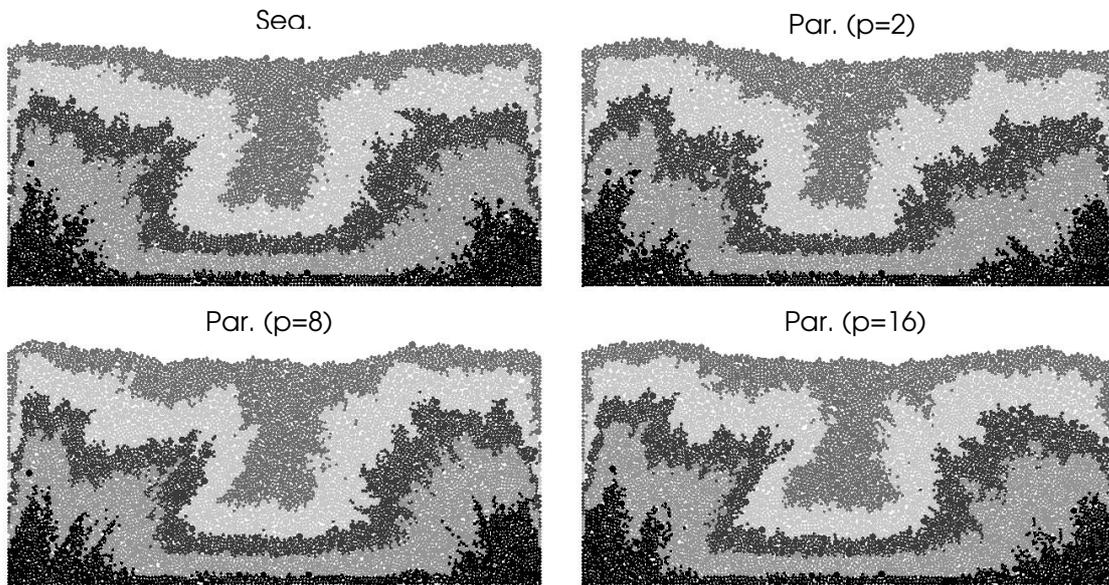


FIG. 3.14 – Profil de différentes simulations (seq., 2, 8 et 16 proc.) en fin de processus : les différences soulignent l'influence de la re-numérotation sur la méthode.

La figure 3.14 souligne le fait que les solutions obtenues en fin de simulation sont différentes. En observant le profil de la surface libre on note des différences. La répartition des grosses particules n'est pas identique dans les différentes simulations. La figure 3.15 montre quant à elle l'évolution du nombre d'itération moyenné sur 100 intervalles de temps. Cette évolution est similaire pour chaque simulation, et donc quelque soit le nombre de processeurs utilisés, les échantillons semblent passer par les mêmes crises. Sur la même figure, on peut aussi trouver des comparaisons des distributions des orientations de contacts ($\theta \rightarrow p(\theta)$: probabilité de l'orientation θ) pour les différentes simulations. Deux instants ont été choisis correspondant au premier et second tiers du processus. Pour les deux instants, les courbes se superposent avec de très légères variations. Ainsi l'anisotropie du milieu, variable utilisée pour caractériser un milieu granulaire ne dépend pas de la simulation.

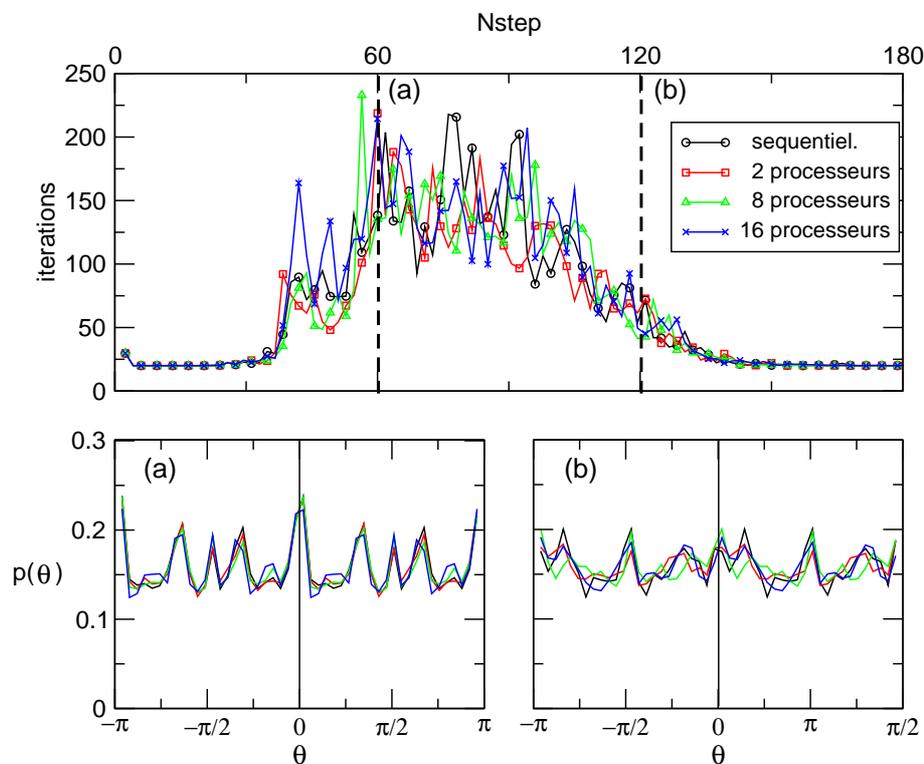


FIG. 3.15 – Evolution des itérations au cours du processus pour différentes simulations (seq. , 2, 8 et 16 proc.). Les approximations de la fonction $p(\theta)$, présent en deux instant de la simulation, concordent pour chaque simulation.

Pour les tests parallèles effectués sur des compressions bi-axiales, nous avons décidé d'illustrer par d'autres quantités les résultats. Ainsi, ayant un milieu confiné soumis à une évolution quasi statique, nous avons, après avoir calculé les tenseurs de contrainte σ , σ^s et σ^w correspondant respectivement au tenseur global, du réseau fort et du réseau faible,

calculé le déviateur q et la pression p pour chacun d'eux, et représenté l'évolution du rapport q/p pour les différentes simulations et tenseurs (global sur la figure 3.16 et réseau fort et faible sur la figure 3.17).

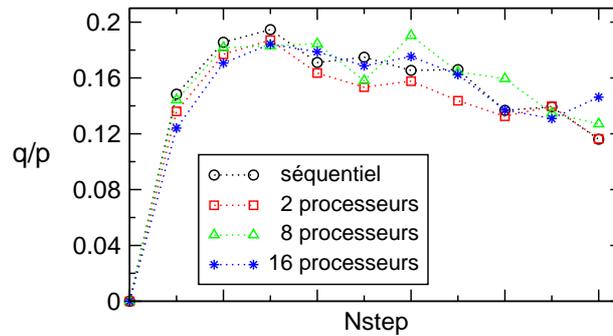


FIG. 3.16 – Evolution du rapport q/p du tenseur de contrainte global.

L'évolution de q/p , identique pour chaque simulation, correspond au résultat classique des compressions bi-axiales de systèmes compacts. Une fois encore le calcul parallèle ne vient pas interférer avec l'évolution des grandeurs mécaniques. Sur la figure 3.17 nous avons dissocié les deux réseaux de contact et représenté l'évolution de la même grandeur pour chacun d'eux. Là encore les mêmes conclusions en sont tirées.

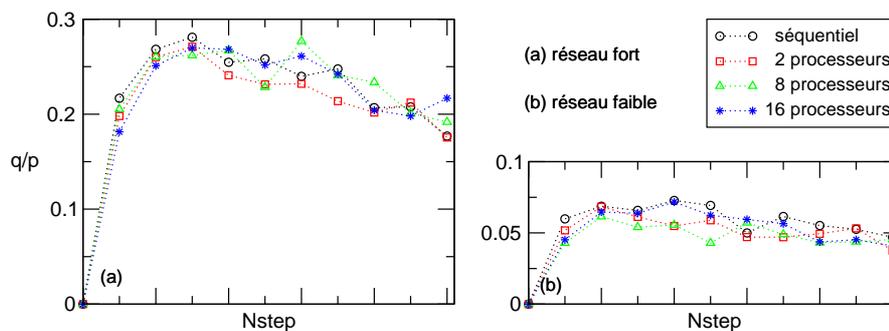


FIG. 3.17 – Evolution du rapport q/p des tenseurs de contrainte des réseaux fort et faible

On peut souligner le caractère fluide du réseau faible puisque le rapport q/p reste faible, contrairement au réseau fort, "squelette" de l'échantillon qui assure l'essentiel de la tenue mécanique.

Conclusion

Pour l'évolution de milieux granulaires denses, la méthode parallèle donne des résultats concordant avec ceux obtenus en séquentiel tout en donnant des résultats en temps *CPU*. L'évolution du speed-up nous laisse envisager de meilleurs résultats pour des échantillons plus conséquents.

3.4 vers un code 100% parallèle ?

Dans le cas de systèmes granulaires denses, la méthode de parallélisation donne des résultats macroscopiques concordants en comparaison avec les simulations séquentielles, et ce pour des temps de calcul plus raisonnables. Toutefois pour des simulations fortement dynamiques, la méthode n'est pas forcément la plus appropriée. Le solveur pouvant rapidement représenter une faible part du temps de calcul, sa seule parallélisation ne suffira pas à diminuer le temps de calcul. Ainsi dans une optique d'obtenir un code 100% parallèle, s'attaquer à la parallélisation des autres parties du code (détection des contacts, actualisation des grandeurs cinématiques,...) devient alors essentielle : une partie d'un code prenant moins de 10% du temps *CPU* lors d'une exécution séquentielle peut vite atteindre plus de 60% du temps suivant le nombre de processeurs utilisés (figure 3.18). A ce titre nous pouvons bénéficier de nombreuses bibliothèques parallèles performantes (ParaMETIS, ScaLAPACK [16]) et ainsi que de méthodes performantes [24, 31].

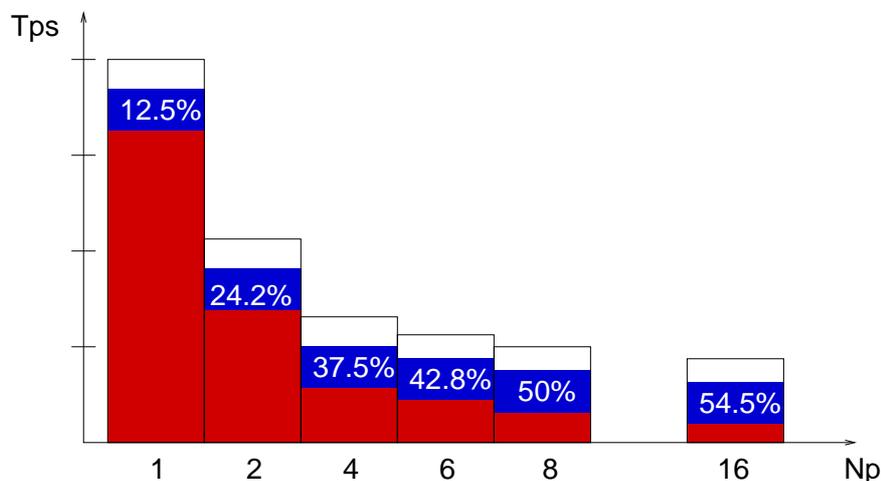


FIG. 3.18 – Augmentation du pourcentage de temps *CPU* de la partie séquentielle lors d'un calcul parallèle.

Pour clore cette partie faisons deux remarques. La première remarque est que pour une étude numérique des milieux granulaires, certains utilisateurs préféreront utiliser les calculateurs parallèles pour lancer plusieurs programmes simultanés plutôt que de lancer un

seul calcul sur plusieurs processeurs différents. Le besoin de faire des études statistiques leur demande un certain nombre de simulations : le "nombre" de calculs devient plus important que le "temps" de calcul. Cette catégorie d'utilisateurs préférera posséder des algorithmes de résolution performants en séquentiel mais pas forcément parallélisables. D'autres préféreront, au contraire, obtenir un seul résultat en peu de temps pour pouvoir analyser les données obtenues et régler des paramètres en fonction, et dans ce cas, le fait d'utiliser plusieurs centaines de processeurs pour un même calcul leur apparaîtra comme un grand avantage. La deuxième remarque est que les directives *OpenMP* sont peu adaptées pour le calcul intensif [153] et deviennent vite inefficaces lors de calculs parallèles massifs même si la méthode de parallélisation est très rigoureuse. Le langage *MPI* apparaît alors comme un moyen de mettre en oeuvre un code entièrement parallèle dédié au calcul intensif. Il demande toutefois une gestion des échanges de données qui peut être non triviale.

Afin de pouvoir répondre à ces remarques nous proposons dans le chapitre suivant de présenter une méthode de type Gradient Conjugué plus efficace que Gauss-Seidel pour des problèmes linéaires et pouvant bénéficier de versions parallèles [158] basées sur le partage des tâches.

Chapitre 4

Algorithmes de type gradient

Les algorithmes de type Gauss Seidel ne sont pas a priori les plus adaptés à notre ambition qui est de traiter le plus efficacement possible des problèmes à très grand nombre de corps et de contacts. Ils ne sont pas rapides et sont, en calcul des structures, utilisés comme auxiliaires dans des stratégies plus performantes, notamment comme "lisseurs" dans les méthodes multigrilles [7]. Parmi les méthodes les plus efficaces dans la résolution de systèmes linéaires de grande taille, on trouve les algorithmes basés sur le gradient conjugué ; ils sont souvent dotés de préconditionneurs qui en accélèrent la convergence [13, 14] ; ils peuvent être associés à des stratégies plus sophistiquées comme les méthodes de décomposition de domaine [93]. Ils présentent en outre deux intérêts importants pour nous. Basés sur la formulation d'un problème d'optimisation quadratique, ils peuvent être étendus à des problèmes non linéaires et en particulier avec contraintes comme celles issues du contact. Leur structure utilisant pour l'essentiel des produits scalaires et matrices-vecteurs les rend également intrinsèquement parallèles si les préconditionneurs le sont eux-mêmes.

Concernant la mécanique numérique du contact, ils ont été utilisés, dans les années 80, pour résoudre des problèmes de contact sans frottement [41, 98] dans le contexte de la modélisation par élément finis. Une extension au contact frottant est proposée dans [128], mais est restreinte aux problèmes de calculs de structures utilisant le champ de déplacement comme inconnue principale ; une méthode de point fixe, se traduisant par une boucle supplémentaire, est utilisée pour satisfaire la loi de Coulomb, et la loi de frottement est régularisée pour ne pas introduire de termes non régulier dans l'énergie à minimiser. Nous proposons dans ce qui suit un algorithme constitué d'une unique boucle d'itérations, très proche dans son implémentation de la méthode du Gradient Conjugué classique, complété par des adaptations locales (niveau des contacts) consistant en une correction de l'itéré et une projection des gradients. Ces adaptations sont justifiées, dans la section 4.1 par une étude des formulations des problèmes d'optimisation ou de quasi-optimisation associés au contact, au frottement et au contact frottant, dans la section 4.2 par une revue des méthodes de gradients et de leurs améliorations (projection, conjugaison) et dans la section 4.3 par une analyse du processus de diagonalisation d'une stratégie de point fixe. Implémentation, préconditionneurs et critères d'erreur sont discutés dans la section

4.4. Malgré cette approche qui se veut rigoureuse, aucune démonstration mathématique de convergence ne peut être exhibée et des tests numériques réalisés sur de nombreuses simulations de systèmes granulaires réunis dans la section ?? viennent confirmer les bons comportements attendus d'une telle stratégie.

4.1 (Quasi) optimisation en mécanique du contact

L'utilisation de techniques d'optimisation nécessite d'écrire le problème de contact comme un problème d'optimisation. L'attention est alors portée sur des problèmes discrets obtenus à partir de techniques d'approximation (tel que la méthode des éléments finis) ou obtenus directement par la modélisation de systèmes mécaniques possédant un nombre fini de degrés de liberté. Dans [5], on peut trouver un parallèle entre problèmes de structures et de granulats montrant que le choix d'une stratégie de résolution des équations non linéaires dépend du problème même si le formalisme est similaire dans les deux problèmes. Quelques éléments de cette étude comparative sont fournis dans la partie qui suit. On se limite à des conditions de contact unilatéral sans frottement.

4.1.1 Problèmes types ; *Structures* et *Granulat*

Dans [5], nous présentons deux problèmes typiques, appelé problème *Structure* et *Granulat*. Le problème *Structure* consiste à déterminer l'équilibre d'un corps élastique discrétisé en éléments finis en contact affleurant avec une fondation rigide. L'entier n dénote le nombre total de degrés de liberté du corps élastique, $m = n_c$ le nombre de noeuds potentiellement en contact. Les équations d'équilibre forment un système linéaire (dans une première approximation) dans lequel on retrouve la matrice de rigidité \mathbb{K} , les déplacements généralisés \mathbf{x} , les réactions normales au contact \mathbf{r} aux noeuds de l'aire en contact, les forces extérieures \mathbf{f} . Ceci est la première ligne des relations suivantes à satisfaire,

$$\begin{cases} \mathbb{K}\mathbf{x} - \mathbb{H}\mathbf{r} = \mathbf{f} \\ \mathbb{H}^T\mathbf{x} \geq \mathbf{0}, \mathbf{r} \geq \mathbf{0}, \mathbf{r}.\mathbb{H}^T\mathbf{x} = 0 \end{cases} \quad (4.1)$$

La matrice \mathbb{H} passe du repère local au repère global pour tous les noeuds potentiellement en contact. Les conditions de contacts exprimées à travers les deux inégalités et une condition de complémentarité relient les réactions de contacts et les composantes normales du déplacement sur la surface potentielle de contact. En mécanique des structures, il convient souvent de considérer d'autres non linéarités issues du corps (les forces internes $\mathbb{K}\mathbf{x}$ deviennent des fonctions non linéaires du déplacement $\mathbb{K}(\mathbf{x})$) ou de la géométrie du contact lorsque de fortes courbures doivent être prises en compte : $\mathcal{G}(\mathbf{x})$ remplace $\mathbb{H}^T\mathbf{x}$ et $\nabla(\mathcal{G})^T\mathbf{r}$ $\mathbb{H}\mathbf{r}$. De telles non linéarités additionnelles conduisent à utiliser des méthodes de type Newton, étendues aux équations non différentiables [4, 32, 119], ou combinées avec des méthodes de contact spécifiques : Gauss-Seidel [70, 74, 128], Lagrangien Augmenté [29, 90].

La situation est assez différente pour le problème *Granulat*. Suivant l'approche de Moreau [107], l'objet principal du calcul est la vitesse relative au contact et une méthode "time

stepping” est utilisée comme intégrateur temporel. Nous renvoyons à la première partie déjà exposée ainsi qu’à [23, 108] pour une discussion sur les différents intégrateurs temporels. Pour une comparaison avec le cas précédent nous considérons le problème sur un seul pas de temps ; nous devons prédire la distribution des vitesses dans une assemblée de corps rigides à la fin du pas de temps. Le contact opère maintenant comme une loi d’interaction entre particules. Comme nous l’avons montré, les équations de la dynamique peuvent être facilement écrites aux points de contact utilisant les variables dans le repère local : la vitesse relative \mathbf{u} au contact et l’impulsion \mathbf{r} . Le système à résoudre est donc constitué des équations de la dynamique réduites ou condensées aux points de contact,

$$\begin{cases} \mathbb{W}\mathbf{r} - \mathbf{u} = \mathbf{b} \\ \mathbf{r} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0}, \mathbf{r} \cdot \mathbf{u} = 0 \end{cases} \quad (4.2)$$

Le membre de droite de la première ligne dans (4.2) prend en compte les forces extérieures et les vitesses relatives de la fin du pas de temps précédent. Dans les granulats denses, le nombre de contacts n_c est plus grand que le nombre de corps n_b ; la matrice de masse \mathbb{M} appartient à $\mathbb{R}^{n \times n}$ avec $n = 3n_b$ et $6n_b$ suivant la dimension du problème considéré (2D ou 3D). La matrice de masse étant facilement inversible, car diagonale, on peut exprimer l’opérateur de Delassus par $\mathbb{W} = \mathbb{H}^T \mathbb{M}^{-1} \mathbb{H}$. Toutefois la matrice \mathbb{W} peut être singulière et donc conduire à une indétermination des impulsions. L’opérateur de passage $\mathbb{H}(\mathbf{q})$, qui dépend généralement du paramètre de configuration $\mathbf{q} \in \mathbb{R}^n$, est maintenue constant sur le pas de temps.

Le problème *Structure* (4.1) peut être reformulé comme un problème d’optimisation où la solution \mathbf{x} est l’argument du minimum de la fonction quadratique $\phi(\cdot)$ sur l’ensemble défini par les contraintes $\mathbb{H}^T \mathbf{x} \geq \mathbf{0}$,

$$\mathbf{x} = \underset{\mathbb{H}^T \tilde{\mathbf{x}} \geq \mathbf{0}}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{x}} \cdot \mathbb{K} \tilde{\mathbf{x}} - \mathbf{f} \cdot \tilde{\mathbf{x}}. \quad (4.3)$$

Généralement la matrice \mathbb{K} est définie positive et la solution est unique ; ce qui explique l’égalité. Le problème *Granulat* (4.2) est aussi reformulé comme un problème d’optimisation en considérant par contre les impulsions de contact \mathbf{r} comme variables primales,

$$\mathbf{r} \in \underset{\tilde{\mathbf{r}} \geq \mathbf{0}}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (4.4)$$

Dans les problèmes granulaires la matrice \mathbb{W} est souvent seulement semi-définie positive entraînant une indétermination des impulsions ; nous devons ainsi déterminer une solution parmi plusieurs solution admissibles. Les techniques d’optimisation sont recommandés dans ce cas tant que la fonction reste bornée inférieurement, condition que l’on peut aisément vérifier .

4.1.2 Conditions de frottement

Tout d’abord la description du contact est complétée en décomposant la matrice \mathbb{H} en ses parties normale et tangentielle : $\mathbb{H} = [\mathbb{H}_n, \mathbb{H}_t]$. Les lois de frottement relient la

vitesse relative tangentielle à la composante tangentielle de l'impulsion de contact. La loi de Tresca postule un seuil de frottement non négatif, noté s_α pour chaque contact α , au delà duquel le glissement est proscrit (statut adhérent). Dans une forme incrémentale le problème de contact frottant s'exprime encore comme un problème d'optimisation sous contrainte avec cependant une fonction coût non différentiable,

$$\mathbf{x} = \underset{\mathbb{H}_n^T \tilde{\mathbf{x}} \geq \mathbf{0}}{\operatorname{argmin}} \phi(\tilde{\mathbf{x}}) + \sum_{\alpha=1}^{n_c} s_\alpha \|\mathbb{H}_{\alpha,t}^T(\tilde{\mathbf{x}} - \bar{\mathbf{x}})\|, \quad (4.5)$$

où $\bar{\mathbf{x}}$ dénote le déplacement au pas de temps précédent et $\mathbb{H}_{\alpha,t}^T$ est la contribution tangentielle du contact α de \mathbb{H}^T ; $\mathbb{H}_{\alpha,t}^T(\tilde{\mathbf{x}} - \bar{\mathbf{x}})$ correspond alors à un incrément local de glissement. Pour le problème *Granulat* une loi de Tresca fournit une formulation simple,

$$\mathbf{r} \in \underset{\tilde{\mathbf{r}} \in \mathcal{C}(s)}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (4.6)$$

où uniquement l'ensemble contraint est modifié en comparaison du problème 4.4; cet ensemble est le produit cartésien de demi-cylindres infinis dans \mathbb{R}^2 ou \mathbb{R}^3 ayant pour section $\mathcal{B}(0, s_\alpha)$, le disque de rayon s_α ,

$$\mathcal{C}(s) = \prod_{\alpha=1}^{n_c} \mathbb{R}^+ \times \mathcal{B}(0, s_\alpha) \quad (4.7)$$

Cependant la loi de Coulomb, loi de frottement plus usuelle lie le seuil de frottement à la composante normale de la force de contact locale (ou impulsion) via un coefficient de frottement noté μ . Le problème couplé de contact frottant n'est alors plus un problème d'optimisation. Formellement il est toujours possible de formuler un problème de "*quasi*"-optimisation (en référence aux plus classiques inégalités quasi-variationnelles) pour qui l'ensemble des contraintes dépend des composantes normales de la solution comme paramètre; le problème *Granulat* avec contact frottant s'exprime alors,

$$\mathbf{r} \in \underset{\tilde{\mathbf{r}} \in \mathcal{C}(\mu r_n)}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (4.8)$$

La non unicité de la solution est due à la singularité de la matrice \mathbb{W} mentionnée précédemment, mais également à la non associativité de la loi de glissement conformément aux nombreuses études réalisées sur cet aspect [3, 8, 36, 46, 113]. La normalité du gradient de la fonction coût à l'ensemble conique des contraintes n'est pas satisfaite comme le montre la figure 4.1; le gradient \mathbf{g} de la fonction coût ($\mathbf{g} = \mathbb{W} \mathbf{r} - \mathbf{b}$) est orthogonale au demi-cylindre passant par \mathbf{r} .

Dans la suite nous considérons un problème quadratique *standard* sous contraintes utilisant les notations du problème *Granulat*,

$$\mathbf{r} \in \underset{\tilde{\mathbf{r}} \in \mathcal{C}}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}. \quad (4.9)$$

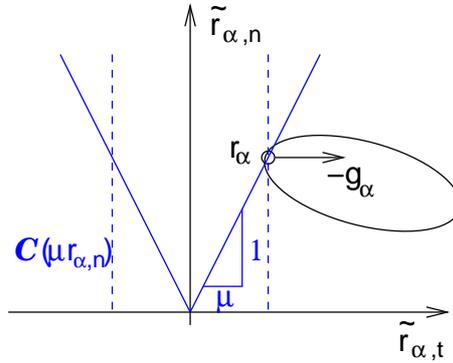


FIG. 4.1 – Le cône de Coulomb et la loi de glissement

Il est intéressant de préciser le caractère géométrique de l'ensemble des contraintes \mathcal{C} selon la nature du problème. Si on considère seulement le contact unilatéral, l'ensemble \mathcal{C} est le cône positif ou l'orthant positif de \mathbf{R}^{n_c} . Si l'on considère seulement le frottement (le contact est alors assumé maintenu) l'ensemble convexe est un hyper-rectangle produit cartésien de segments $s_\alpha[-\mu, \mu]$. Pour un problème de contact et frottement de Tresca dans une modélisation bi-dimensionnelle l'ensemble \mathcal{C} est le produit cartésien de bandes semi-infinies dans \mathbb{R}^2 . Pour ces trois cas l'ensemble \mathcal{C} est un ensemble convexe polyédrique. Pour un problème de contact et frottement de Tresca l'ensemble \mathcal{C} est le produit cartésien de cylindres semi-infinis dans \mathbb{R}^3 mais il n'est plus polyédrique. Pour un problème de contact et frottement de Coulomb, le problème de Tresca en tant que problème de contact frottant peut être vu comme un problème intermédiaire dans une stratégie numérique.

4.2 Les méthodes de gradient (projeté)

La méthode de gradient la plus simple pour résoudre (4.9) est la méthode du gradient avec projection définissant le schéma itératif par,

$$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}} [\mathbf{r}^k + \gamma(\mathbf{b} - \mathbb{W}\mathbf{r}^k)], \quad \forall \gamma > 0 \quad (4.10)$$

Le gradient est noté $\mathbf{g}^k = \mathbb{W}\mathbf{r}^k - \mathbf{b}$ et la direction de descente est $\mathbf{u}^k = -\mathbf{g}^k$. La projection sur l'ensemble admissible concerne uniquement l'itéré mais pas le gradient. La condition de convergence est lié au paramètre positif γ , qui doit être inférieur à $\frac{2}{\lambda_{\max}(\mathbb{W})}$, où $\lambda_{\max}(\mathbb{W})$ est la plus grande valeur propre de \mathbb{W} , pour que la convergence soit assurée. La convergence est lente spécialement si la matrice \mathbb{W} est mal conditionnée (c.f. figure 4.2).

4.2.1 Les méthodes de Rosen généralisées

La première catégorie d'amélioration de l'algorithme du gradient consiste en la méthode du gradient projeté. Pour une meilleure compréhension, considérons une seule contrainte

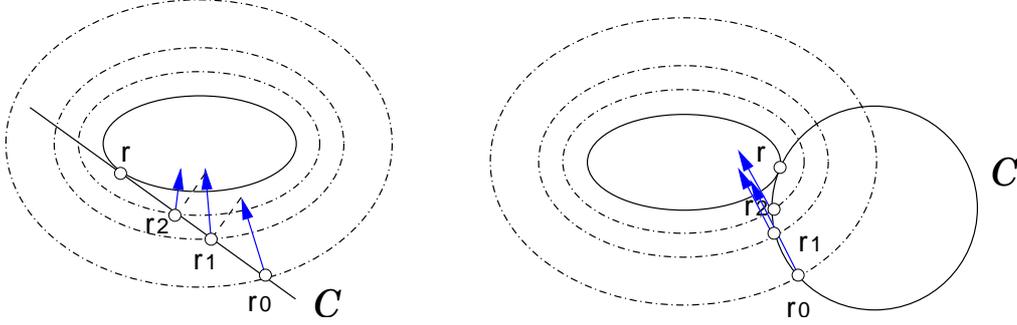


FIG. 4.2 – Méthode du gradient avec projection pour des contraintes linéaires et non linéaires ; les lignes de niveau de la fonction coût sont représentées par chaque itéré.

linéaire donnée par $\mathbf{d} \cdot \mathbf{r} \geq 0$, où \mathbf{d} est un vecteur unitaire. Nous pouvons prouver que le vecteur $\mathbf{p}^k := \text{proj}_C(\mathbf{r}^k - \gamma \mathbf{g}^k) - \mathbf{r}^k$ est encore une direction de descente ($\mathbf{p}^k \cdot \mathbf{g}^k < 0$) et, si $(\mathbf{r}^k - \gamma \mathbf{g}^k) \cdot \mathbf{d} < 0$, cette nouvelle direction est la projection du gradient sur l'hyperplan frontière du demi-espace de l'ensemble des contraintes : $\mathbf{p}^k = \gamma[-\mathbf{g}^k + (\mathbf{d} \cdot \mathbf{g}^k)\mathbf{d}]$; la contrainte est alors appelée active. La recherche de la solution peut donc être restreinte au sous-espace des contraintes actives. Suivant les idées développées donc par Rosen pour des contraintes linéaires [134] et non linéaires [135], Mehrez [99] nomme méthode de Rosen généralisée le schéma utilisant les deux paramètres α and γ ,

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \alpha[\text{proj}_C(\mathbf{r}^k - \gamma \mathbf{g}^k) - \mathbf{r}^k], \quad \forall \gamma > 0, \quad \forall \alpha > 0. \quad (4.11)$$

La convergence est assurée sous certaines conditions sur ces paramètres [99] ; deux cas particuliers peuvent être mentionnés,

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

Ces conditions ont la fâcheuse conséquence de ralentir la convergence (tout en la garantissant) ; on peut donc en pratique s'en passer. Cependant le paramètre α doit rester dans l'intervalle $]0, 1[$ assurant ainsi que \mathbf{r}^{k+1} reste dans l'ensemble des contraintes. Cette restriction peut être combinée à un pas optimal variable α^k pour accélérer la convergence. Mais une telle stratégie peut conduire à une solution non optimale comme prouvé par Wolfe avec un simple contre-exemple [19]. Il apparaît alors plus sûr de sortir de l'espace admissible et puis d'y projeter l'itéré menant ainsi au schéma suivant,

$$\mathbf{r}^{k+1} = \text{proj}_C\{\mathbf{r}^k + \alpha^k[\text{proj}_C(\mathbf{r}^k - \gamma^k \mathbf{g}^k) - \mathbf{r}^k]\}, \quad \forall \gamma^k > 0, \quad \forall \alpha^k > 0. \quad (4.12)$$

Ce schéma est assez général mais il n'utilise pas le caractère polyédrique de l'ensemble convexe disponible en 2D - et il ne permet pas d'introduire le concept de gradient conjugué. D'autre part le choix optimal du paramètre γ est une question ouverte. La figure 4.3 montre

que pour une valeur assez petite de γ et un ensemble polyédrique, la direction de descente \mathbf{p}^k ne dépend pas de γ : le gradient \mathbf{g}^k , ou plus précisément la direction de descente $\mathbf{u}^k = -\mathbf{g}^k$, est projeté sur le cône tangent de \mathcal{C} en \mathbf{r}^k . On note le cône tangent $T_{\mathcal{C}}$. Cette remarque conduit à postuler une version locale de la méthode de Rosen généralisée ou méthode de gradient projeté, combinant les projections précédentes et le choix optimal du paramètre α^k , détaillée dans le tableau 4.1.

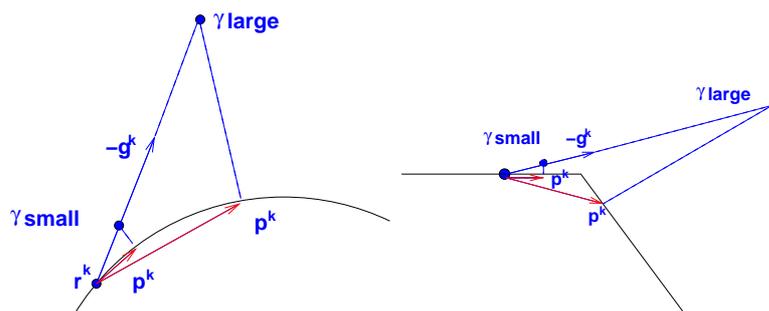


FIG. 4.3 – Influence de la norme de γ sur la direction de descente pour des ensembles de contraintes réguliers ou polyédriques.

$\begin{array}{l} \mathbf{r}^0, \mathbf{u}^0 \text{ et } \mathbf{p}^0 \text{ donné} \\ \left[\begin{array}{l} k \rightarrow k + 1 \\ \mathbf{r}^{k+\frac{1}{2}} = \mathbf{r}^k + \alpha^{k+1} \mathbf{p}^k \quad \text{avec } \alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k} \\ \mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}}) \\ \mathbf{u}^{k+1} = \mathbf{b} - \mathbb{W} \mathbf{r}^{k+1} \\ \mathbf{p}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1})) \end{array} \right. \end{array}$

TAB. 4.1 – Algorithme de Gradient Projeté (PG)

4.2.2 Un algorithme de gradient projeté conjugué

Pour améliorer la convergence de l'algorithme précédent, nous introduisons une approche de type gradient conjugué en conjuguant la direction précédente \mathbf{p}^{k-1} avec le gradient courant \mathbf{u}^k après projection sur le cône tangent, et définissant ainsi un algorithme de Gradient Projeté Conjugué (CPG) (c.f. table 4.2).

Le procédé de conjugaison est a priori pleinement efficace si les itérés restent, pendant plusieurs itérations, dans le même ensemble de contraintes actives, c'est à dire sur la

$ \begin{array}{l} \mathbf{r}^0, \mathbf{u}^0 \text{ et } \mathbf{p}^0 \text{ donné} \\ k \rightarrow k + 1 \\ \mathbf{r}^{k+\frac{1}{2}} = \mathbf{r}^k + \alpha^{k+1} \mathbf{p}^k \quad \text{avec } \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; T_{\mathcal{C}}(\mathbf{r}^{k+1})) \\ \mathbf{p}^{k+1} = \mathbf{w}^{k+1} + \beta^{k+1} \mathbf{z}^{k+1} \quad \text{avec } \beta^{k+1} = \frac{\mathbf{w}^{k+1} \cdot \mathbb{W} \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k} \end{array} $

TAB. 4.2 – Algorithme de Gradient Projeté Conjugué (CPG)

même facette ou la même arête de l'ensemble des contraintes. Un cas intéressant correspond à la situation où l'ensemble des contraintes actives est celui auquel appartient la solution. Dans le problème de contact cet ensemble définit le statut de contact de la solution, concaténation des statuts locaux de contact (contact / pas contact). Après que cet ensemble soit trouvé, l'algorithme de Gauss-Seidel est lent pour atteindre la solution; nous pouvons espérer dans ce cas un meilleur comportement de l'algorithme de gradient conjugué; les tests numériques doivent alors confirmer ou infirmer cette prédiction. Une stratégie similaire fût proposée par May [98] et Dilintas *et al.* [41] pour les problèmes de contact sans frottement, mais notre formulation est plus générale et synthétique. Ainsi Dilintas *et al.* [41] fixe l'ensemble des contraintes actives, applique ensuite l'algorithme du gradient projeté conjugué sur cet ensemble jusqu'à convergence avant d'actualiser les contraintes et recommencer le processus. Il compte ainsi bénéficier de la conjugaison des gradients dans le sous-espace vectoriel des contraintes actives, mais son schéma contient alors deux niveaux de boucles d'itérations dont il faut gérer les critères d'arrêt. Le schéma du gradient conjugué projeté dans le tableau 4.2 ne cherche pas à fixer les contraintes actives sur plusieurs itérations; il n'est même pas certain que les gradients projetés, \mathbf{w}^k et \mathbf{z}^k , soient dans un ensemble de contraintes actives, par exemple si les projections sur le même cône tangent ne s'effectuent sur une même facette (contraintes actives) du cône. Une tentative de stabilisation provisoire des contraintes actives, et donc la possibilité de bénéficier temporairement du processus de conjugaison, consiste à projeter \mathbf{p}^{k-1} , non sur le cône tangent à \mathcal{C} en \mathbf{r}^k , mais sur la facette du cône tangent auquel appartient le gradient projeté courant \mathbf{w}^k , notée $A(\mathbf{w}^k)$ (c.f. figure 4.4). Ainsi le gradient conjugué suivant \mathbf{p}^k est bien défini dans un ensemble de contraintes actives, celui défini par \mathbf{w}^k :

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

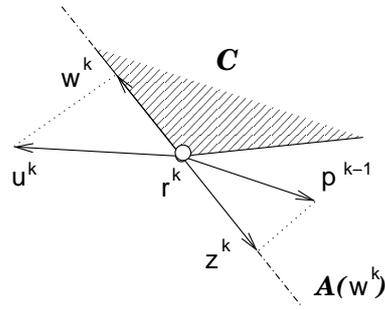


FIG. 4.4 – Illustration du sous espace des contraintes actives $A(\mathbf{w}^k)$.

4.2.3 Application au contact unilatéral, au frottement.

Pour le problème standard (4.9), l'algorithme précédent a l'avantage de travailler avec un ensemble de contraintes admissibles possédant une forme simple. Pour le contact unilatéral contact ou le frottement bi-dimensionnel, l'ensemble des contraintes est le produit cartésien d'intervalles de \mathbb{R} . Par conséquent les projections, de l'itéré et du gradient, sont faciles à implémenter et ceci composante par composante, comme précisé ici.

Contact Unilatéral : $\mathcal{C} = \mathbb{R}_+^{n_c}$

$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$	$\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$
pour $\alpha = 1, n_c$ $r_{\alpha}^{k+1} = \max(r_{\alpha}^{k+\frac{1}{2}}, 0)$	pour $\alpha = 1, n_c$ if $r_{\alpha}^{k+1} = 0$ then $w_{\alpha}^{k+1} = \max(u_{\alpha}^{k+1}, 0)$ else $w_{\alpha}^{k+1} = u_{\alpha}^{k+1}$

Frottement 2D : $\mathcal{C} = \prod_{\alpha=1}^{n_c} [-s_{\alpha}, s_{\alpha}]$

$\mathbf{r}^{k+1} = \text{proj}_{\mathcal{C}}(\mathbf{r}^{k+\frac{1}{2}})$	$\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}; T_{\mathcal{C}}(\mathbf{r}^{k+1}))$
pour $\alpha = 1, n_c$ Si $r_{\alpha}^{k+\frac{1}{2}} > s_{\alpha}$ alors $r_{\alpha}^{k+1} = s_{\alpha}$ si $r_{\alpha}^{k+\frac{1}{2}} < -s_{\alpha}$ alors $r_{\alpha}^{k+1} = -s_{\alpha}$ sinon $r_{\alpha}^{k+1} = r_{\alpha}^{k+\frac{1}{2}}$	pour $\alpha = 1, n_c$ si $r_{\alpha}^{k+1} = s_{\alpha}$ then $w_{\alpha}^{k+1} = \min(u_{\alpha}^{k+1}, 0)$ si $r_{\alpha}^{k+1} = -s_{\alpha}$ alors $w_{\alpha}^{k+1} = \max(u_{\alpha}^{k+1}, 0)$ sinon $w_{\alpha}^{k+1} = u_{\alpha}^{k+1}$

4.3 Un algorithme spécifique pour le contact frottant

Le couplage du contact unilatéral et du frottement de Coulomb est exprimé par (4.8) comme un problème de quasi-optimisation. Une approche classique consiste à mettre en oeuvre un algorithme de point fixe sur le seuil de frottement.

4.3.1 Algorithme de point fixe

Le problème (4.8) est approché par une succession de problèmes d'optimisation (4.6) à seuil fixé s . L'algorithme résultant est résumé dans le schéma du tableau 4.3.

Initialisation du seuil \mathbf{s}	
$\mathbf{s}^0 = (s_\alpha^0, \alpha = 1, n_c)$ avec $s_\alpha^0 \geq 0$	
$(\mathbf{r}^0 = \mathbf{0})$	
$l = l + 1$	
$s_\alpha^l = \mu r_{\alpha,n}^{l-1}, \alpha = 1, n_c$	(point fixe)
$\mathbf{r}^l = \underset{\tilde{\mathbf{r}} \in \mathcal{C}^l}{\operatorname{argmin}} \frac{1}{2} \tilde{\mathbf{r}} \cdot \mathbb{W} \tilde{\mathbf{r}} - \mathbf{b} \cdot \tilde{\mathbf{r}}$	(problème de contact frottant de type Tresca)

TAB. 4.3 – Algorithme de point fixe

Pour bien comprendre la version diagonalisée qui va suivre, il est utile de détailler cet algorithme dans le tableau 4.4. En particulier on insiste sur trois étapes qui sont bien séparées dans ce schéma : *l'actualisation du seuil* de frottement au début de la boucle de point fixe, suivie de *l'initialisation de l'algorithme* du gradient projeté conjugué par la valeur obtenue à convergence à l'étape précédente, et enfin, à l'intérieur de la boucle du gradient, *la projection de l'itéré* sur les contraintes.

4.3.2 Version diagonalisée

Le nouvel algorithme consiste à supprimer la boucle indexé par l suivant un procédé de diagonalisation. Les adaptations à effectuer suivent deux principes. Tout d'abord la projection des gradients est toujours réalisée sur le cône tangent au produit cartésien de demi cylindres infinis approchant le produit cartésien de cônes convexes. Ainsi, à la convergence, la normalité du gradient à l'ensemble approché des contraintes reste satisfaite conformément à la caractérisation de la solution (c.f. figure 4.1). Ensuite la projection de l'itéré est remplacé par une correction. Cette correction doit concentrer dans une seule opération les trois étapes du schéma précédent soulignées dans le tableau 4.4 : l'actualisation du seuil, l'initialisation de l'itéré et la projection elle-même. Ceci nous conduit à un algorithme possédant une simple boucle et présenté dans le tableau 4.5.

4.3. UN ALGORITHME SPÉCIFIQUE POUR LE CONTACT FROTTANT

Initialisation du seuil : $\mathbf{s}^0 = (s_\alpha^0, \alpha = 1, n_c)$ avec $s_\alpha^0 \geq 0$	
$(\mathbf{r}^0 = \mathbf{0})$	
$l = l + 1$	
$s_\alpha^l = \mu r_{\alpha, n}^{l-1}, \alpha = 1, n_c$	(actualisation du seuil)
$\mathbf{r}^{l,0} = \mathbf{r}^{l-1, conv}$	(initialisation de l'itéré)
$\mathbf{u}^0 = \mathbf{b} - \mathbb{W}\mathbf{r}^{l,0}, \mathbf{p}^0 = \mathbf{u}^0$	
$k = k + 1$	
$\mathbf{r}^{l, k+\frac{1}{2}} = \mathbf{r}^{l, k} + \alpha^{k+1} \mathbf{p}^k$ avec $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	
$\mathbf{r}^{l, k+1} = proj_{C^l}(\mathbf{r}^{l, k+\frac{1}{2}})$ (projection de l'itéré)	
$\mathbf{u}^{k+1} = \mathbf{b} - \mathbb{W}\mathbf{r}^{l, k+1}$	
Si convergence alors $\mathbf{r}^l = \mathbf{r}^{l, k+1}$ et on quitte la boucle.	
$\mathbf{w}^{k+1} = proj(\mathbf{u}^{k+1}; T_{C^l}(\mathbf{r}^{l, k+1}))$	
$\mathbf{z}^{k+1} = proj(\mathbf{p}^k; A(\mathbf{w}^{k+1}))$	
$\mathbf{p}^{k+1} = \mathbf{w}^{k+1} + \beta^{k+1} \mathbf{z}^{k+1}$ with $\beta^{k+1} = -\frac{\mathbf{w}^{k+1} \cdot \mathbb{W}\mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$	

TAB. 4.4 – Algorithme détaillé de point fixe

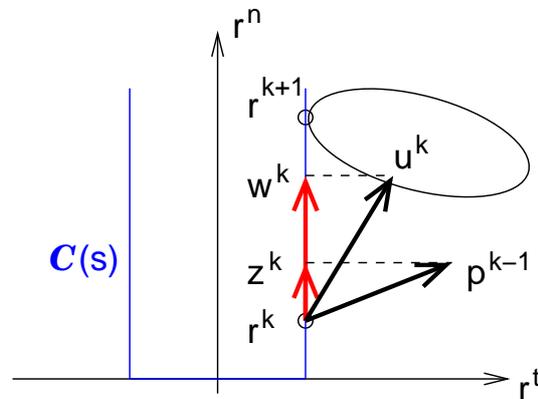


FIG. 4.5 – Gradient Projeté pour un problème de type "Tresca" (l'indice α est omis)

Les modifications sont détaillées contact par contact (localement) et restreinte à une modélisation bi-dimensionnelle, pour laquelle la stratégie de gradient projeté a un sens conformément aux remarques des sections 4.2.1 et 4.2.2.

Initialisation de $\mathbf{u}^0 = \mathbf{b} - \mathbb{W}\mathbf{r}^0$, $\mathbf{w}^0 = \mathbf{u}^0$, $\mathbf{p}^0 = \mathbf{u}^0$ $k = k + 1$ $\mathbf{r}^{k+\frac{1}{2}} = \mathbf{r}^k + \alpha^{k+1}\mathbf{p}^k$ avec $\alpha^{k+1} = \frac{\mathbf{u}^k \cdot \mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$ Correction de l'itéré : $\mathbf{r}^{k+\frac{1}{2}} \rightarrow \mathbf{r}^{k+1}$ $\mathbf{u}^{k+1} = \mathbf{b} - \mathbb{W}\mathbf{r}^{k+1}$ Critère de convergence : si il est vrai alors $\mathbf{r}^{\text{CONV}} = \mathbf{r}^{k+1}$ Projection des gradients : $\mathbf{w}^{k+1} = \text{proj}(\mathbf{u}^{k+1}, T_{\mathcal{C}^{k+1}}(\mathbf{r}^{k+1}))$ $\mathbf{z}^{k+1} = \text{proj}(\mathbf{p}^k; A(\mathbf{w}^{k+1}))$ $\mathbf{p}^{k+1} = \mathbf{w}^{k+1} + \beta^{k+1}\mathbf{z}^{k+1}$ avec $\beta^{k+1} = -\frac{\mathbf{w}^{k+1} \cdot \mathbb{W}\mathbf{p}^k}{\mathbf{p}^k \cdot \mathbb{W}\mathbf{p}^k}$

TAB. 4.5 – Algorithme de Gradient Projeté Conjugué (CPG), version contact frottant

Correction de l'itéré

Tout d'abord l'actualisation du seuil diffère de l'algorithme à double boucle puisque $r_{\alpha, n}^{k+\frac{1}{2}}$ n'est pas nécessairement positif ou nul,

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

Ensuite les étapes d'initialisation et de projection doivent se fondre en une seule procédure puisqu'une seule itération du gradient est à effectuer avant une nouvelle actualisation du seuil. Bien évidemment si $\mathbf{r}_{\alpha}^{k+\frac{1}{2}}$ se trouve à l'extérieur des contraintes $\mathcal{C}_{\alpha}^{k+1}$, il y est alors projeté. S'il se trouve à l'intérieur, une application stricte de la procédure de projection amènerait à confirmer $\mathbf{r}_{\alpha}^{k+\frac{1}{2}}$ en $\mathbf{r}_{\alpha}^{k+1}$, et à considérer pour la suite un statut *adhérent*, c'est-à-dire à ne pas activer une contrainte *glissement*. Ainsi, si la solution a un statut *glissant*, l'itéré peut osciller entre les statuts *glissant* et *adhérent* à cause de la forme conique des contraintes et ne jamais activer la conjugaison des gradients dans le "bon" ensemble des contraintes actives. Il est donc souhaitable dans certains cas où $\mathbf{r}_{\alpha}^{k+\frac{1}{2}}$ se trouve à l'intérieur de $\mathcal{C}_{\alpha}^{k+1}$, de ne pas l'y conserver mais de le projeter sur la frontière. Plus précisément si la direction de descente locale courante \mathbf{p}_{α}^k ne possède pas de composante tangentielle, le statut du contact local, qui était donc précédemment "glissant", est maintenu même si l'itéré est à l'intérieur du cône de frottement ainsi qu'à l'intérieur du cylindre qui l'approche $\mathcal{C}_{\alpha}^{k+1}$; le cas (b) de la figure 4.6 illustre cette situation. Le procédé de correction détaillé dans le tableau 4.6 utilise la notion de statut de contact (local) qui est nécessaire pour comprendre la projection qui suit à savoir celle des gradients. Les statuts fournissent

4.3. UN ALGORITHME SPÉCIFIQUE POUR LE CONTACT FROTTANT

si $r_{\alpha,n}^{k+\frac{1}{2}} < 0$	alors $\mathbf{r}_\alpha^{k+1} = \mathbf{0}$	$(statut_\alpha^{k+1} = \text{séparation})$
sinon	$r_{\alpha,n}^{k+1} = r_{\alpha,n}^{k+\frac{1}{2}}$	$\epsilon = sign(r_{\alpha,t}^{k+\frac{1}{2}})$
	Si $ r_{\alpha,t}^{k+\frac{1}{2}} > s_\alpha^{k+1}$	alors $r_{\alpha,t}^{k+1} = \epsilon s_\alpha^{k+1}$ $(statut_\alpha^{k+1} = \text{glissement } \epsilon)$
	sinon	
	si $p_{\alpha,t}^k = 0$	then $r_{\alpha,t}^{k+1} = \epsilon s_\alpha^{k+1}$ $(statut_\alpha^{k+1} = \text{glissement } \epsilon)$
	sinon	$r_{\alpha,t}^{k+1} = r_{\alpha,t}^{k+\frac{1}{2}}$ $(statut_\alpha^{k+1} = \text{adhérent})$

TAB. 4.6 – Correction de l'itéré

l'information pertinente sur l'itéré en lequel le cône tangent T_C doit être évalué.

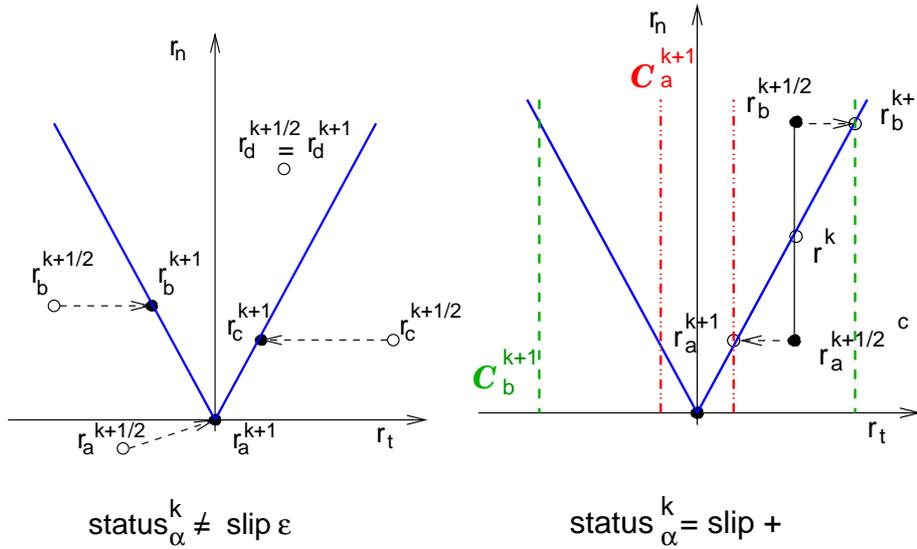


FIG. 4.6 – Correction de l'itéré pour diverses situations en fonction du statut précédent.

Projection des gradients

Comme l'approximation de l'ensemble convexe est bien définie, la projection du gradient sur le cône tangent est une simple combinaison des procédures détaillées pour le contact unilatéral et le frottement bi-dimensionnel. On retrouve une illustration sur la figure 4.7 et un résumé synthétique dans le tableau 4.7,

si $statut_{\alpha}^k = \text{séparation}$	alors	si $u_{\alpha,n}^k \leq 0$ alors $\mathbf{w}_{\alpha}^k = \mathbf{0}$ sinon $w_{\alpha,n}^k = u_{\alpha,n}^k$ and $w_{\alpha,t}^k = 0$
si $statut_{\alpha}^k = \text{adhérent}$	alors	$\mathbf{w}_{\alpha}^k = \mathbf{u}_{\alpha}^k$
si $statut_{\alpha}^k = \text{glissement}^+$	alors	$w_{\alpha,n}^k = u_{\alpha,n}^k$ and $w_{\alpha,t}^k = \min(0, u_{\alpha,t}^k)$
si $statut_{\alpha}^k = \text{glissement}^-$	alors	$w_{\alpha,n}^k = u_{\alpha,n}^k$ and $w_{\alpha,t}^k = \max(0, u_{\alpha,t}^k)$

TAB. 4.7 – Projection des gradients

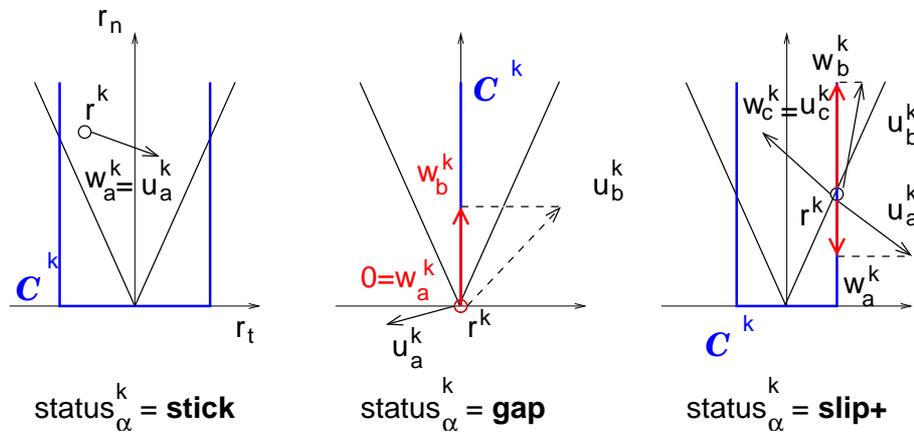


FIG. 4.7 – Projection du gradient pour diverses situations (en fonction du statut précédent)

L'extension de cette approche à une modélisation tridimensionnelle n'est pas immédiate ; le cône tangent "n'approche pas suffisamment bien" localement un ensemble non polyédrique ; la notion de contrainte active n'est plus vraiment disponible, facettes du cône tangent et frontière des contraintes ne coïncidant plus. Le chapitre 5 suivant est en particulier consacré aux différentes extensions tridimensionnelles possibles à la stratégie présentée ici.

4.4 Preconditionneur et critères de convergence

Les algorithmes présentés précédemment peuvent être appliqués aux deux types de problèmes de contact mentionnés en section 4.1.1. Cependant ils sont surtout adaptés au problème de type *Granulat*, pour lesquels les variables locales exprimées dans leur repère local sont privilégiées ; les contraintes prennent alors des caractéristiques agréables, orthant positif ou produit cartésien de domaines simples. L'application au problème Structure sans frottement requiert au minimum un passage dans le repère local en éliminant l'opérateur \mathbb{H} . Il existe des algorithmes d'optimisation pour des contraintes d'inégalité générales [63],

mais il conviendrait de les connaître dans le détail pour les adapter ensuite au problème de quasi-optimisation ; cette démarche ne nous a pas semblé nécessaire, nos préoccupations concernant exclusivement les milieux granulaires. Pour le problème *Structure* avec frottement, les algorithmes ne sont pas applicables tels quels ; il convient au préalable de reformuler le problème 4.5, exprimé en déplacement, sous la forme 4.8 privilégiant les efforts de contact (encore faut-il inverser \mathbb{K}). Par conséquent les méthodes de gradient projeté conjugué sont testées sur des problèmes de type *Granulat*. Il reste cependant des choix à effectuer avant de lancer les tests numériques ; ils concernent l'implémentation qui tient compte de la structure du code LMGC90, les préconditionneurs indispensables aux algorithmes de gradient conjugué et les critères de convergence afin de pouvoir comparer les méthodes.

4.4.1 Quelques remarques sur l'implémentation

Dans l'algorithme CPG deux produits matrice-vecteur sont effectués ($\mathbb{W}\mathbf{p}^k, \mathbb{W}\mathbf{r}^{k+1}$) au lieu d'un seul, $\mathbb{W}\mathbf{p}^k$, pour la méthode du gradient conjugué appliquée au problème quadratique non contraint. Une alternative consiste à effectuer le calcul de $\mathbb{W}\mathbf{r}^{k+1}$ seulement si au moins une projection a été activée dans l'étape de correction du pas précédent ($\mathbf{r}^{k+\frac{1}{2}} \rightarrow \mathbf{r}^{k+1}$) ; le gradient \mathbf{u}^{k+1} est ainsi actualisé par une formule classique utilisant le produit $\mathbb{W}\mathbf{p}^k$. Cette stratégie peut être efficace pour les problèmes de contact sans frottement [41] et de contact frottant avec une loi de Tresca, mais ce n'est pas probant pour les problèmes de contact avec du frottement de Coulomb car la correction de $\mathbf{r}^{k+\frac{1}{2}}$ est activée dès qu'est prédit un statut glissant pour un quelconque contact α . De plus dans les problèmes granulaire le nombre de contacts est assez grand pour induire un nombre non négligeable de changement de statuts.

Toutes les instructions du schéma du CPG peuvent être traitées en parallèle, car elles utilisent des produits scalaires et matrice-vecteur parce que les projections ou corrections sont effectuées contact par contact. Aussi un préconditionneur doit être le plus simple possible pour éviter notamment d'introduire une séquentialisation du code et donc d'handicaper un traitement sur ordinateur à architecture parallèle.

Le stockage de la matrice \mathbb{W} dépend du type de problème. Pour les matériaux granulaires, cette matrice n'est pas assemblée comme dans les matrices pour élément finis. Dans le logiciel *LMGC 90* [45] utilise essentiellement un solveur de type Gauss-Seidel Non Linéaire par blocs (méthode NLGS), la matrice \mathbb{W} n'est pas formée initialement ; deux stratégies ont été présentées dans le chapitre précédent stockant ou non l'ensemble de la matrice \mathbb{W} , ou plus exactement les petites matrices $\mathbb{W}_{\alpha\beta}$. La version SDL, où sont calculées et stockées en début de pas les matrices $\mathbb{W}_{\alpha\beta}$, sera prise comme référence pour comparer les algorithmes CPG et NLGS, car cette étape est aussi présente dans la méthode CPG.

4.4.2 Préconditionneurs de base

Que veut dire un préconditionneur pour solveur pour problème quadratique d'optimisation sous contraintes ? Le préconditionnement de l'algorithme du gradient conjugué appliqué à un problème quadratique non contraint consiste simplement à multiplier le

gradient courant \mathbf{u}^{k+1} par une matrice de préconditionnement \mathbb{P} . Cette simple instruction supplémentaire permet de remplacer le problème initial mal conditionné par un autre possédant un meilleur conditionnement, consistant à minimiser $\frac{1}{2}\bar{\mathbf{s}}.\bar{\mathbb{W}}\bar{\mathbf{s}} - \bar{\mathbf{b}}.\bar{\mathbf{s}}$ avec le changement de variables suivant,

$$\mathbf{s} = \mathbb{P}^{\frac{1}{2}}\bar{\mathbf{s}}, \quad \mathbf{b} = \mathbb{P}^{-\frac{1}{2}}\bar{\mathbf{b}} \quad \text{et} \quad \bar{\mathbb{W}} = \mathbb{P}^{\frac{1}{2}}\mathbb{W}\mathbb{P}^{\frac{1}{2}}$$

Dans le cas d'un problème d'optimisation quadratique sous contraintes, il convient d'identifier les étapes de l'algorithme CPG appliqué au problème préconditionné par rapport aux variables initiales, c'est-à-dire celles du problème non préconditionné. Le problème préconditionné s'écrit,

$$\text{Inf}_{\bar{\mathbf{s}} \in \bar{\mathcal{C}}} \frac{1}{2}\bar{\mathbf{s}}.\bar{\mathbb{W}}\bar{\mathbf{s}} - \bar{\mathbf{b}}.\bar{\mathbf{s}}, \quad (4.13)$$

avec de plus $\bar{\mathcal{C}} = \mathbb{P}^{-\frac{1}{2}}\mathcal{C}$. Dans le schéma de l'algorithme CPG appliqué à (4.13), les différentes projections des nouvelles variables, sur le convexe modifié comme sur son cône tangent, doivent être définies afin qu'elles coïncident avec les projections euclidiennes des variables initiales sur les convexes et cônes tangents initiaux. Autrement dit, comment définir $proj(\cdot; \bar{\mathcal{C}})$ de telle sorte que $\bar{\mathbf{a}} = proj(\bar{\mathbf{b}}; \bar{\mathcal{C}})$ corresponde à $\mathbf{a} = proj(\mathbf{b}; \mathcal{C})$? On peut aisément démontrer qu'il suffit d'adopter la norme associée à \mathbb{P} , $\|\mathbf{x}\|_{\mathbb{P}}^2 = \mathbf{x}.\mathbb{P}\mathbf{x}$,

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

La projection sur le cône tangent est définie de la même façon. Alors l'algorithme CPG exprimé par rapport aux variables initiales ne comportent, comme pour la méthode du gradient conjugué appliqué à un système linéaire, qu'une étape supplémentaire : le produit matrice-vecteur $\mathbb{P}\mathbf{u}^{k+1}$, et ce avant la projection des gradients. Ce nouvel algorithme est alors dénommé de Gradient Projeté Conjugué Préconditionné (Preconditionned Conjugate Projected Gradient, PCPG). Les préconditionneurs ont suscité un nombre important de travaux pour accélérer la convergence dans la résolution des systèmes linéaires de grande taille. Il en existe une grande variété ; les plus efficaces (Incomplete LU factorization et ses variantes ILUT, MILU) sont souvent complexes à mettre en œuvre ou spécifiques aux méthodes par éléments finis (Element By Element). Ils sont d'autant plus efficaces qu'ils approchent au mieux la matrice inverse de \mathbb{W} . L'efficacité sur le solveur est un compromis entre cette distance à \mathbb{W}^{-1} et le coût du calcul du préconditionneur. Pour les problèmes contraints il n'est pas sûr qu'il soit nécessaire de prendre un préconditionneur très sophistiqué car la matrice n'est qu'un élément du système, le convexe des contraintes en est un autre tout aussi important. On a donc opté dans un premier temps pour les préconditionneurs les plus simples à mettre en œuvre. Nous sommes confortés dans ce choix par la nécessité également de conserver le caractère parallèle du solveur global, avec son préconditionneur.

Préconditionneurs diagonaux

Si la matrice \mathbb{W} est décomposée en ses parties diagonale et non diagonale, $\mathbb{W} = \mathbb{D} - \mathbb{E}$, le préconditionneur est l'inverse de la partie diagonale, $\mathbb{P} = \mathbb{D}^{-1}$. Cette procédure n'est

pas coûteuse car le produit matrice-vecteur est très simple. Cette première stratégie peut être souvent suffisante pour améliorer la convergence, spécialement quand un échantillon granulaire polydisperse est considéré : l'échantillon contient ainsi de petites et de grosses particules entraînant des différences notables entre les termes (diagonaux) de la matrice de masse et donc de la matrice \mathbb{W} . Une variante consiste à considérer un découpage par blocs de la matrice \mathbb{W} , où chaque bloc diagonal correspond à un contact. Pour les problèmes granulaires cette distinction n'est que significative pour les grains polygonaux ou polyédraux car les matrices blocs diagonales de \mathbb{W} sont diagonales pour les grains sphériques.

Préconditionneurs polynomiaux

Parmi les preconditionneurs plus sophistiqués, les preconditionneurs polynomiaux semblent de bons candidats car leur préparation n'est pas tellement coûteuse. La matrice \mathbb{P} conserve la même structure que \mathbb{W} ; par exemple le preconditionneur de niveau un peut être construit lorsque \mathbb{W} elle-même est construite, $\mathbb{P} = \mathbb{D}^{-1} + \mathbb{D}^{-1}\mathbb{E}\mathbb{D}^{-1}$. Mais ce preconditionneur nécessite un troisième produit matrice-vecteur à chaque itération; cela peut être très coûteux comparé au gain que l'on peut espérer en termes d'itérations. Certains tests concernant de petits exemples de disques montrent que la valeur du conditionnement est seulement divisé par deux lors du passage du preconditionneur diagonal (niveau zéro) au preconditionneur de niveau un.

4.4.3 Critères de convergence pour systèmes non réguliers

La question de la précision et le choix du critère de convergence n'est pas une tâche facile suivant le problème considéré et encore plus dans le contexte des collections denses de corps rigides en contact. La solution exacte est habituellement inconnue et la seule information qui peut être utilisée est une certaine distance jusqu'à elle. Les bornes supérieures et inférieures d'une telle distance peuvent être obtenues en utilisant des propriétés mathématiques de convergence de l'algorithme, si l'existence d'une unique solution a été prouvée et si la convergence de l'algorithme a été établie. A part quelques situations particulières, ces circonstances ne sont jamais réalisées. Dans le cas granulaire la non unicité n'est pas une exception, mais plutôt la règle, qui peut être causée par deux phénomènes. Le premier est de nature géométrique. Le réseau des forces entre grains dans un granulat dense en l'absence de frottement peut être vu comme un réseau de treillis composé de barres liant les centres des particules. Un tel système est modélisé dans la matrice \mathbb{W} , il peut être hyperstatique et la matrice \mathbb{W} s'avère singulière (semi-définie positive). Afin d'effectuer le calcul du coefficient de conjugaison β^{k+1} , on vérifie si le terme $\mathbf{p}^k \cdot \mathbb{W} \mathbf{p}^k$ est nul; auquel cas l'algorithme est stoppé. Si ce terme est nul sans que le soit, une direction dans le noyau de \mathbb{W} est mise en évidence. Nous avons cependant constaté que cette précaution, bien que théoriquement nécessaire, n'est en pratique jamais activée. Le second point est lié au frottement. Les lois de frottement induisent aussi une multiplicité des solutions des impulsions de contact. La solution (numérique) dépend de l'histoire de chargement et numériquement du choix de l'initialisation de l'algorithme; les forces de contact précédentes sont naturellement considérées [110].

Dans les simulations des matériaux granulaires utilisant l'approche NSCD utilisant un algorithme NLGS, M. Jean introduit un critère en énergie appliqué à une correction de Jacobi de l'itéré courant [26]. Pour obtenir un test équivalent pour le solveur CPG, nécessaire pour les comparaisons futures, nous considérons les variations de vitesses relative $\Delta \mathbf{u}$ issues de deux impulsions successives et reliée à la direction de descente par

$$\Delta \mathbf{u} = \mathbf{u}^{k+1} - \mathbf{u}^k = \mathbb{W}(\mathbf{r}^{k+1} - \mathbf{r}^k) = \mathbb{W}\Delta \mathbf{r}$$

Nous utilisons les deux énergies de référence introduit aux chapitre 1, section 1.5.4,

$$\begin{aligned} \mathcal{E}^m &= \sum_{\alpha} \bar{\mathbf{r}}_{\alpha} \cdot \mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha} \\ \mathcal{E}^q &= \frac{1}{N_{actif}} \sum_{\alpha} (\mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha}) \cdot (\mathbf{W}_{\alpha\alpha} \bar{\mathbf{r}}_{\alpha}) \end{aligned}$$

et trois critères de convergence qui doivent alors être satisfait,

$$\left\{ \begin{aligned} \tilde{e}_{vr}^m &= \frac{1}{\mathcal{E}^m} \sum_{\alpha} \mathbf{r}_{\alpha}^{k+1} \cdot \mathbb{W}_{\alpha\alpha} \Delta \mathbf{r}_{\alpha} \\ \tilde{e}_{vv}^q &= \frac{1}{\mathcal{E}^q} \sqrt{\sum_{\alpha} (\mathbb{W}_{\alpha\alpha} \Delta \mathbf{r}_{\alpha}) (\mathbb{W}_{\alpha\alpha} \Delta \mathbf{r}_{\alpha})} \\ \tilde{e}_{vr}^q &= \frac{1}{\mathcal{E}^m} \sqrt{N_{actif} \sum_{\alpha} (\mathbf{r}_{\alpha}^{k+1} \cdot \mathbb{W}_{\alpha\alpha} \Delta \mathbf{r}_{\alpha}) (\mathbf{r}_{\alpha}^{k+1} \cdot \mathbb{W}_{\alpha\alpha} \Delta \mathbf{r}_{\alpha})} \end{aligned} \right. ,$$

où \tilde{e}_{vr}^m est une valeur moyenne sur tous les contacts et e^q deux versions quadratiques plus drastique dans de nombreux cas, N_{actif} décompte les contacts actifs (statut *adhérent* ou *glissant*).

4.5 Tests numériques en mécanique des milieux granulaires

Pour réaliser une étude complète des algorithmes de type gradient appliqués à la simulation des milieux granulaires, nous évaluons d'abord leur comportement sur un seul pas de temps. Par la suite des processus complet seront considérés pour prendre en compte la variété des problèmes liés aux milieux granulaires [18, 82, 126, 127] et pour fournir un comportement numérique moyen, un seul pas de temps n'étant pas représentatif au vu des fluctuations à convergence sur un processus. Dans cette partie l'algorithme CPG est essentiellement comparé à la méthode NLGS généralement associé à l'approche NSCD [70, 108]. Toutes les simulations sont réalisées avec le logiciel LMGC90 [45] dédié aux problèmes multicontacts, très utile dans la simulation de milieux granulaire.

4.5.1 Sur un seul pas de temps

Nous isolons un pas de temps à la fin d'un processus de dépôt de particules sous pesanteur dans une boîte ouverte. Ceci nous amène ensuite à déterminer la distribution des forces de contact d'un état d'équilibre dans un granulat compacté. Comme le nombre de contacts est un paramètre important de l'étude, plusieurs échantillons numériques ont été créés : la taille de la boîte reste fixe ($1m \times 1m$), seul le nombre de particules (circulaires) et leur taille changent. Ainsi la capacité de la boîte va de 1 000 à 33 000 disques.

Méthode de descente

Pour un problème sans frottement nous vérifions que les différents algorithmes se comportent bien comme des méthodes de descentes de l'énergie à minimiser. La figure 4.8 donne l'évolution de la fonction $E^k = \frac{1}{2} \mathbf{r}^k \cdot \mathbb{W} \mathbf{r}^k - \mathbf{r}^k \cdot \mathbf{b}$ au cours des itérations pour les méthodes suivantes : Gauss-Seidel Non Linéaire (NLGS), Gradient Projeté Conjugué (CPG), Gradient Projeté Conjugué Préconditionné (PCPG). Le meilleur comportement des méthodes de type gradients n'est pas surprenante mais doit être confirmé pour les problèmes de contact avec frottement avec les adaptations proposées précédemment.

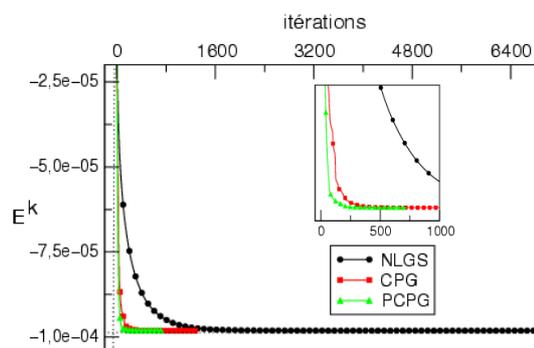


FIG. 4.8 – Décroissance de l'énergie E^k au cours des itérations ($n_c = 28000$).

Convergence

Le cas du contact avec frottement de Coulomb requiert les critères de convergence introduits dans la partie 4.4.3 pour mener à bien les différentes comparaisons. Pour des raisons de simplicité le critère e_m est omis ; le critère e_Q est généralement plus difficile à satisfaire et ainsi plus significatif. Un quatrième algorithme est considéré : le Gradient Projeté sans conjugaison (PG). L'échantillon utilisé pour la figure 4.9 comprend 15 000 contacts. La convergence est atteinte pour une valeur $e_Q < \epsilon = 1.6610^{-6}$. Le procédé de conjugaison se révèle nécessaire pour un bon comportement des méthodes de type gradient en comparaison de l'algorithme de Gauss-Seidel. Le solveur NLGS possède une convergence régulière tandis que les méthodes de type gradient convergent plus vite mais avec de fortes perturbations (en partie dues aux projections et aux corrections). Dans cet exemple le préconditionneur accélère de façon significative la convergence.

Deux paramètres sont significatifs pour apprécier les apports des méthodes de gradient : la taille du système caractérisé par le nombre de contact n_c et la précision requise ϵ . Nous présentons sur la figure 4.10 l'évolution du gain en itérations des méthodes CPG et PCPG ($\delta = (\text{nombre d'itérations NLGS}) / (\text{nombre d'itérations (P)CPG})$) en fonction du nombre de contacts et de trois précisions requises. Le gain est au moins égal à 2.98 ($n_c = 15\,249$

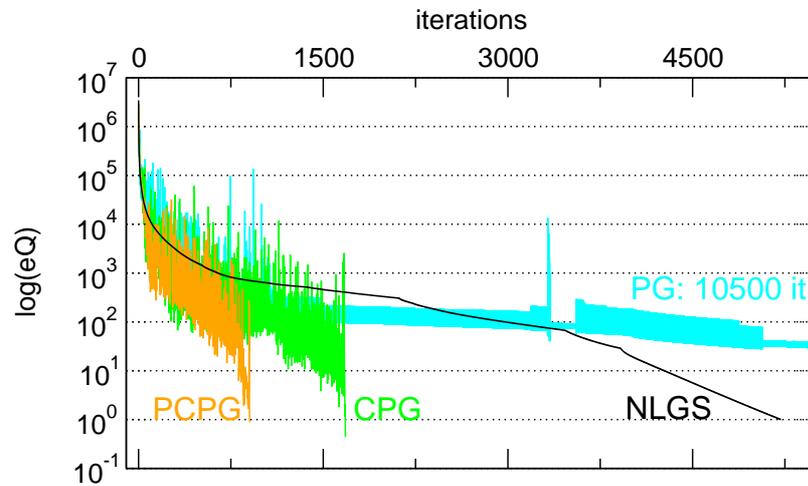


FIG. 4.9 – Evolution de la convergence des quatre algorithmes ($n_c = 15000$).

et $\epsilon = 1.66 \cdot 10^{-5}$) et peut atteindre 9.3 pour CPG ($n_c = 15249$ et $\epsilon = 1.66 \cdot 10^{-4}$) et même 9.47 pour PCPG ($n_c = 28014$ et $\epsilon = 1.66 \cdot 10^{-6}$).

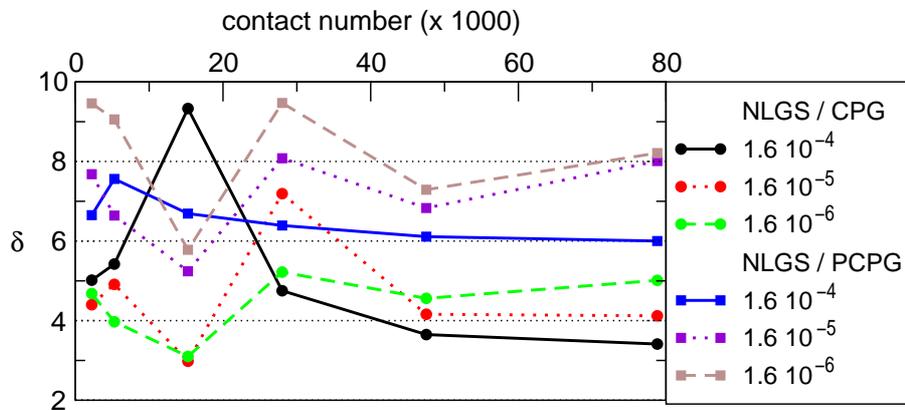


FIG. 4.10 – Gain en itérations, δ , du CPG et du PCPG par rapport à NLGS.

Au vu du comportement erratique de la convergence souligné par la figure 4.9, il n'est pas simple de conclure en deçà de 40 000 contacts. Après cette valeur le gain semble se stabiliser autour de 4 pour l'algorithme CPG et 7 pour l'algorithme préconditionné. Finalement le gain est d'autant plus important que la précision demandée est grande comme le prouve la figure 4.11 où deux tailles d'échantillons sont extraites de la figure précédente. Mais une telle étude nécessite d'être confirmée sur plusieurs processus complets d'évolution.

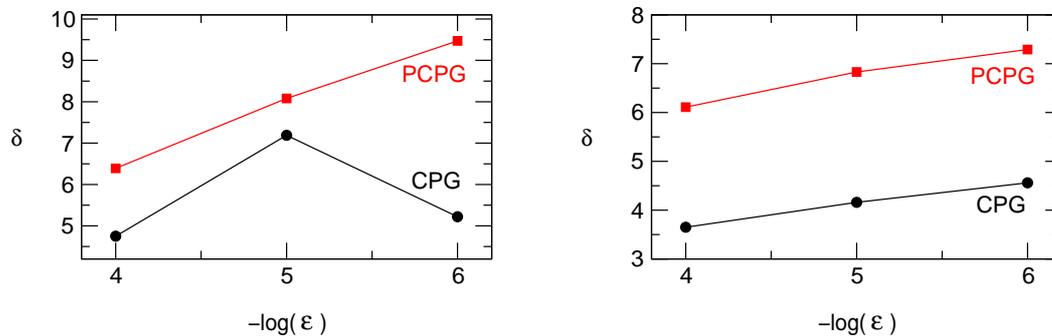


FIG. 4.11 – Gain pour $n_c = 28014$ (gauche) et $n_c = 47531$ (droite).

4.5.2 Influence du frottement

En absence de frottement, le problème lié au système (4.4) est un vrai problème d'optimisation. Sur ce genre de problèmes les méthodes de type gradient ont un bon comportement qui leur permet de converger rapidement. L'introduction du frottement nous oblige à reformuler notre problème comme un problème de quasi-optimisation. Nous avons essayé de quantifier cette perte de propriétés en étudiant la convergence de l'algorithme pour différents coefficients de frottement sur différents échantillons. Les résultats sont illustrés par la figure 4.12 pour un échantillon de 7004 particules, les résultats étant similaires quelque soit ce nombre.

Nous avons étudié la convergence pour 4 coefficients de frottement différents : 0, 0.1, 0.4 et 0.8. Pour $\mu = 0$ nous retrouvons une convergence rapide (connue des méthodes de type gradient conjugué), la convergence du solveur NLGS reste lente même si le nombre d'itérations reste faible. Avec l'introduction du frottement, la convergence du CPG devient chahutée et plus lente. Ce phénomène s'amplifie avec l'augmentation de la valeur de μ . La convergence de l'algorithme de type NLGS est elle aussi affectée mais beaucoup moins que le solveur CPG. De ce fait le nombre d'itérations de ce dernier tend vers le nombre d'itérations Gauss-Seidel. Les courbes de convergence deviennent similaires pour de grandes valeurs de μ , en gardant pour le CPG un caractère chahuté. Le préconditionneur ne semble pas arranger les choses : les courbes de convergence avec et sans préconditionneur accusant le même comportement, même si le nombre d'itérations peut être parfois significativement réduit ($\mu = 0.4$). Il apparaît donc que pour des problèmes de contact avec des coefficients de frottement élevés, la méthode CPG ne présente plus beaucoup d'intérêt par rapport à NLGS. Cependant les coefficients de frottement que nous allons utiliser par la suite reste dans un intervalle ($\in [0, 0.4]$) où le gradient garde un net avantage sur Gauss-Seidel.

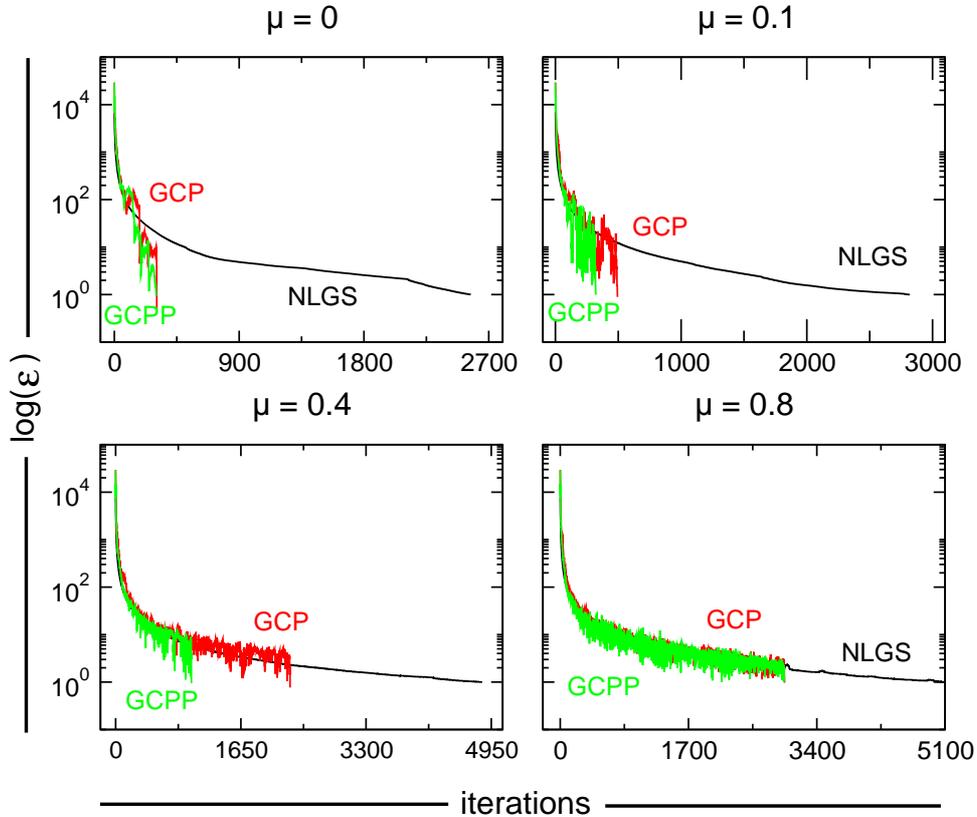


FIG. 4.12 – Influence du coefficient de frottement sur la convergence des différents algorithmes sur un échantillon de 7004 particules.

4.5.3 Sur un processus complet

Pour couvrir de nombreux cas, nous étudions des processus dynamiques et quasi-statiques. Lors de la simulation d'une évolution, nous utilisons généralement la solution du pas de temps précédents pour initialiser l'algorithme. A cause de la multiplicité des solutions dans les problèmes granulaires et des erreurs numériques, la solution au début du second pas de temps sera en principe différent si nous utilisons des méthodes de résolutions différentes. Pour éliminer ou limiter cette difficulté nous pourrions initialiser chaque algorithme par la solution nulle à chaque pas de temps. Mais ce procédé pénalise toutes les méthodes et aucun algorithme ne tire avantage d'une telle situation. Pour comparer nos résultats en terme de qualité, des quantités macroscopiques comme la compacité, l'anisotropie ou la vitesse moyenne, introduites au chapitre 2, sont utilisées.

Tambour tournant

Le tambour tournant est un processus dynamique générant des écoulements de surface (continue ou non) utilisé pour décrire le comportement des avalanches (c.f. figure 4.13). Les enseignements tirés en termes mécanique et physiques sont exposés au chapitre III. Notre modèle est calé sur le dispositif expérimental de Bonamy *et al.* [18] : 7 200 disques (le diamètre moyen est de 3 mm) sont déposés dans un tambour de 450 mm diamètre.

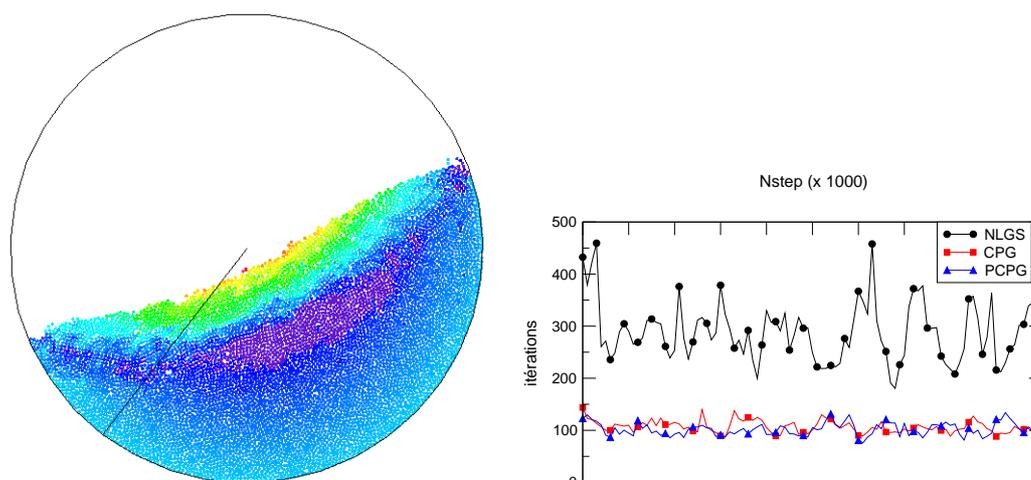


FIG. 4.13 – Simulation numérique en tambour tournant : sur la gauche nous pouvons observer la norme de la vitesse moyenne entre deux pas de temps pour chaque particule. Sur le côté droit, l'évolution des itérations pour différents solveurs.

En complément de la loi de contact frottant, une loi de choc avec restitution normale (coefficient égal à 0.92) est adoptée et accentue les effets dynamiques. Le tambour est soumis à une vitesse angulaire égale à 3 *tpm* (tours par minute). Sur un processus complet (figure 4.13) le nombre moyen d'itérations par pas de temps est égal à 290 pour NLGS et 102 pour le PCPG, ce qui correspond à un gain de 2.75. Cependant la variation du nombre d'itérations durant le processus est moindre pour les algorithmes du CPG et PCPG que pour NLGS. Ici le préconditionneur n'est pas performant : le nombre d'itérations ne décroît pas. Comme le coût d'une itération CPG est plus cher que celui d'une itération NLGS, le gain en temps *CPU* est nul et même négatif (rapport égal à 0.95 dans le tableau 4.8).

Test bi-axial

Il consiste à imposer une déformation biaxiale à un échantillon carré ; une pression constante est appliquée sur la frontière latérale droite alors que le bord supérieur est soumis à un déplacement vertical descendant à vitesse constante (c.f. figure 4.14). Ce chargement est poursuivi jusqu'à 10% de déformation. La figure met en évidence une différence intéressante entre les solutions obtenues à partir des deux types de solveur : le champ des

déplacements est plus homogène au centre de l'échantillon si l'on utilise une méthode de gradient, les valeurs maximales et minimales étant plus concentrées en deux coins.

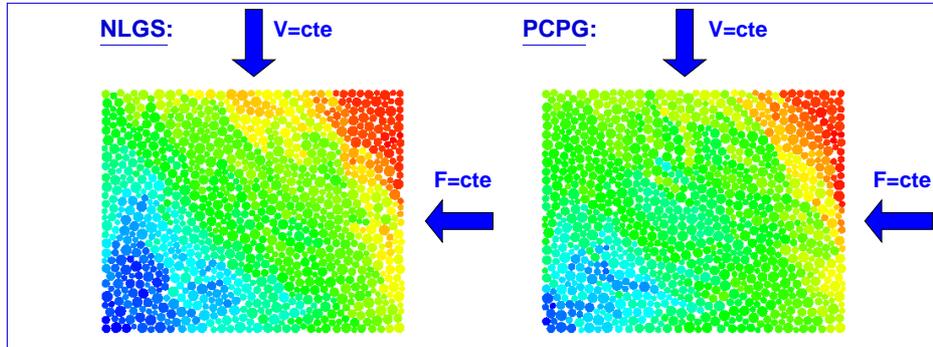


FIG. 4.14 – Champ de déplacement des particules entre deux pas de temps à la fin de la simulation (1024 disques) : contrairement au niveau local où l'on observe de nombreuses différences (position des particules), au niveau global nous distinguons des déplacements global identiques.

Une telle différence peut être expliquée par la transmission globale des informations à travers tout le domaine assuré par les méthodes de gradient par le produit matrice vecteur effectué à chaque itération tandis que l'algorithme NLGS diffuse lentement l'information à la manière d'une onde. Mais ces différences locales ne modifient pas le comportement macroscopique des échantillons accessible par des quantités moyennes significatives [126, 148]. Concernant l'efficacité numérique, une simulation sur un premier échantillon de petite taille (1024 disques) est réalisée utilisant deux précisions différentes. L'évolution du nombre d'itérations est représentée sur la figure 4.15 et les gains moyens en terme d'itérations et de temps CPU sont résumés dans le tableau 4.8. Nous considérons ensuite un échantillon plus conséquent (9216 disques) avec lequel nous procédons au même genre d'étude, résumée par le tableau 4.8. Les gains sont d'autant meilleurs que les échantillons sont gros et les précisions requises exigeantes.

Test de cisaillement simple

Dans un test de cisaillement simple une vitesse angulaire est imposée aux parois latérales de la boîte tandis qu'une pression constante est maintenue sur la paroi supérieure. Deux échantillons sont testés (un grossier et un plus fin) et les résultats sont présentés dans le tableau 4.8 et illustrés par la figure 4.16.

Le gain en itérations croît faiblement, de 3.32 à 3.66 quand on augmente le nombre de grains, de telle sorte que le gain en temps décroît de 2.23 à 1.70. Ceci prouve que l'efficacité en temps *CPU* n'est pas facile à extrapoler à partir d'une simple analyse en terme d'itérations. Pour de gros problèmes la gestion de la mémoire d'un ordinateur peut mo-

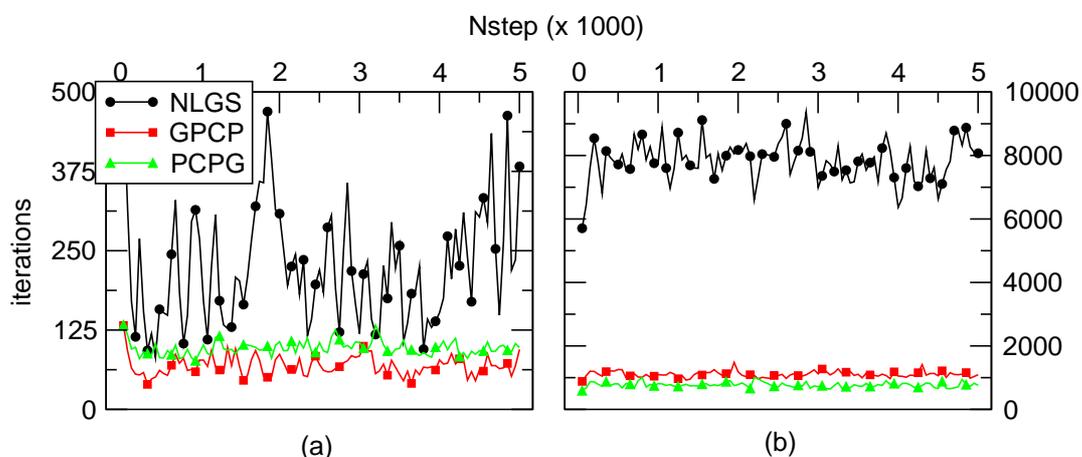


FIG. 4.15 – Evolution du nombre d’itérations obtenue avec les algorithmes NLGS, CPG et PCPG pour un test bi-axial avec 1024 disques pour $\epsilon = 1.66 \cdot 10^{-4}$ (a) et $\epsilon = 1.66 \cdot 10^{-6}$

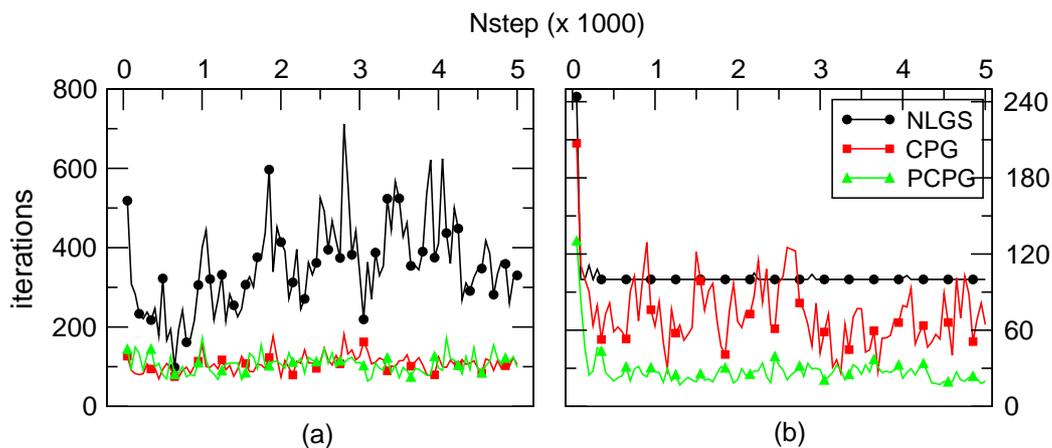


FIG. 4.16 – Evolution du nombre d’itérations dans un test de cisaillement simple (précision demandé : $1.6 \cdot 10^{-4}$).

difier fortement les prédictions. Pour illustrer le comportement global des échantillons, nous présentons sur la figure 4.17 l’évolution de paramètres macroscopiques : la compacité est le volume des particules divisé par le volume total ; le nombre de coordination moyen est le nombre moyen de contacts par particules. L’évolution de telles quantités n’est pas perturbée par l’utilisation des différents algorithmes.

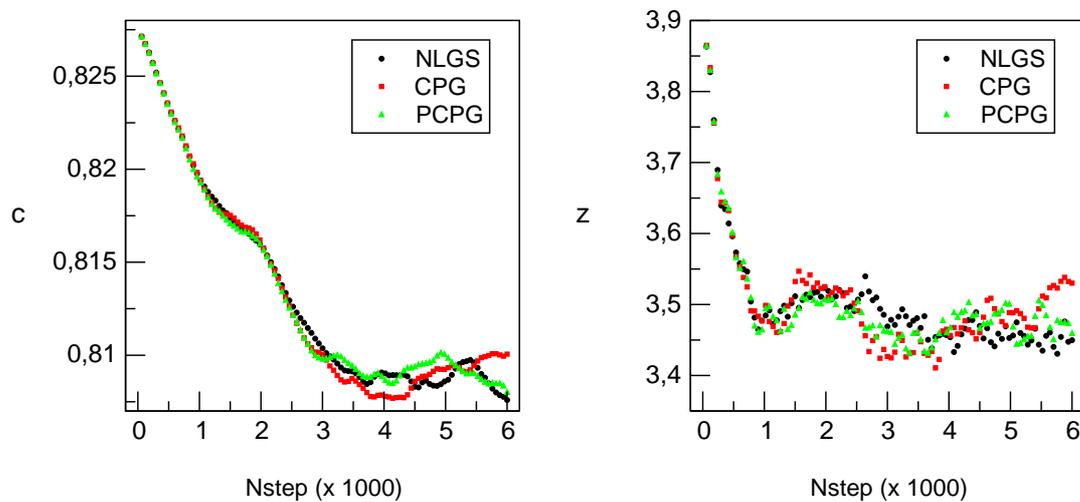


FIG. 4.17 – Evolution des quantités macroscopiques caractéristiques (compacité (gauche) et nombre de coordination (droite)) pour les différents algorithmes dans un essai de cisaillement simple.

Test de compactage à surface libre

Cet exemple diffère fortement des précédents car la sollicitation n'est pas constante en temps. Après dépôt des particules dans une boîte, l'échantillon est compacté par mouvement d'une des parois latérales avec une vitesse non constante en temps. Ceci entraîne une évolution non constante du nombre d'itérations comme le montre la figure 4.18 (contrairement aux exemples précédents). Pour le petit échantillon (1 000 disques) nous trouvons des résultats similaires avec un gain en itérations autour de 3 et un gain en temps *CPU* autour de 1,5 (c.f. tableau 4.8).

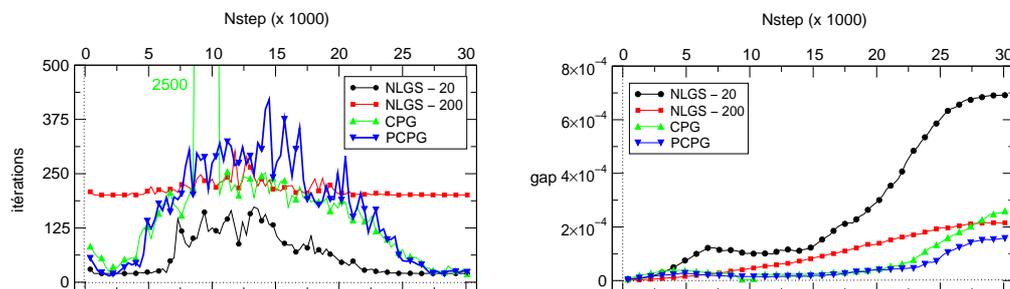


FIG. 4.18 – Pour un compactage à surface libre de 10 000 particules : (a) évolution du nombre d'itérations (gauche) ; (b) évolution de l'interpénétration moyenne (droite).

Pour l'exemple fin (10 000 disques) la figure (4.18a) montre apparemment un meilleur comportement de l'algorithme NLGS. En fait la méthode NLGS converge trop vite en début de simulation ce qui conduit à une accumulation d'interpénétrations comme le prouve la figure (4.18b). L'algorithme NLGS requiert un nombre minimum d'itérations pour diffuser l'information à tout le domaine. Pour éviter cela un minimum d'itérations est imposé (20 pour la version standard). Mais ce seuil n'est pas suffisant : pour retrouver un niveau identique d'interpénétration, 200 itérations au moins sont nécessaires. Aucune règle n'est vraiment valable pour caler ce paramètre : Jean propose une formule - $it_{min} = \sqrt{n_c}$ [26] - mais une telle valeur peut parfois n'être pas suffisante (c.f. exemple) et il appartient alors à l'utilisateur d'être vigilant. Ainsi les méthodes de type gradient deviennent plus efficaces en terme de nombre d'itérations moyen (1.31), mais le gain en temps *CPU* reste inférieur à un.

Résumé

En conclusion de cette étude les méthodes de type gradient mettent en général trois fois moins d'itérations que le solveur NLGS (pour une précision raisonnable). Le gain en temps *CPU* est difficile à évaluer mais est environ autour de 1.5. Les algorithmes de type gradient conjugué semblent être plus robuste pour résoudre des problèmes dynamique où quasi-statique ; l'utilisateur n'a pas à imposer un paramètre supplémentaire pour converger vers une solution acceptable. Si une grande précision est utilisée le gain peut être très appréciable.

processus	n_p	ϵ	rapport <it>	rapport <i>CPU</i>
tambour tournant	7 200 disques	$1.6 \cdot 10^{-4}$	2.75	0.95
test bi-axial	1 024 disques	$1.6 \cdot 10^{-4}$	3.26	1.69
	1 024 disques	$1.6 \cdot 10^{-6}$	7.09	3.38
	9 216 disques	$1.6 \cdot 10^{-4}$	2.17	0.81
	9 216 disques	$1.6 \cdot 10^{-5}$	3.02	1.49
test de cisaillement simple	1 024 disques	$1.6 \cdot 10^{-4}$	3.32	2.23
	9 216 disques	$1.6 \cdot 10^{-4}$	3.66	1.70
compactage à surface libre	1 000 disques	$1.6 \cdot 10^{-4}$	2.99	1.56
	10 000 disques	$1.6 \cdot 10^{-4}$	1.31	0.82

TAB. 4.8 – Performances de l'algorithme CPG vis à vis de la méthode NLGS pour différents processus, tailles d'échantillon et précisions requises.

4.5.4 Multithreading

Les algorithmes de type gradient sont intrinsèquement parallèles car leurs schémas font appel essentiellement à des produits matrice-vecteur. L'implémentation de la version

parallèle n'est pas ainsi une difficulté en elle-même. Comme pour l'algorithme NLGS [130], nous optons pour un multithreading pour ordinateur à mémoire partagée pour éviter la gestion des échanges de messages. L'implémentation est ainsi facilement réalisée grâce aux directives OpenMP [56]. Bien que notre algorithme soit intrinsèquement parallèle, l'utilisation d'un traitement parallèle modifie légèrement la solution sur un processus complet. Sur un pas de temps les versions séquentielle et parallèle donnent les mêmes solutions. Mais lors d'un processus complet, nous accumulons des erreurs numériques qui peuvent avoir des conséquences visibles, et ceci pour les raisons évoquées précédemment. Ainsi durant le processus, l'évolution du nombre d'itérations entre les traitements séquentiel et parallèle divergent après quelques pas de temps. Pour illustrer l'efficacité du calcul parallèle nous utiliserons la notion de speed-up relatif, S_P , définie précédemment, où P est le nombre de processeurs. La figure 4.19 représente l'évolution de S_P pour un test de cisaillement simple et un compactage à surface libre.

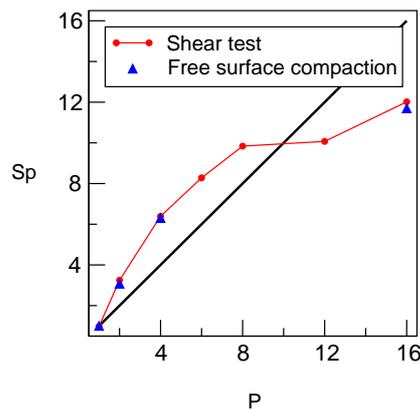


FIG. 4.19 – Evolution du Speed-Up relatif. Les simulations ont été réalisées sur SGI Origin 3800 utilisant des processeurs R14000/500 Mhz.

L'évolution du speed-up peut pour un faible nombre de processeurs ne dépend pas que de la méthode mais également de la manière dont sont administrés les calculs sur le calculateur. C'est ainsi qu'au CINES, où ont été réalisés les tests, l'administrateur système accorde la place mémoire en fonction du nombre de processeurs demandés ; il s'agit cependant toujours de mémoire partagée et non de mémoire distribuée sur chaque processeur. Pour un problème de très grande taille et peu de processeurs, la place mémoire attribuée peut-être insuffisante (0.5 Go par processeur) et les accès aux disques (*swap*) pénalise les calculs ; le calcul de référence à un seul processeur ne prend pas en compte que le comportement de l'algorithme. Ce phénomène intervient sur le cas présenté sur la figure 4.19 jusqu'à quatre processeurs.

4.6 Conclusion

Le schéma numérique proposé se révèle être plus efficace que l'algorithme Gauss-Seidel non linéaire mais le gain n'est pas toujours important. Les investigations pour améliorer la convergence doivent être poursuivies. Plusieurs voies sont à prospecter, certaines classiques comme la sophistication du préconditionneur. On peut aussi chercher à mieux exploiter la conjugaison des gradients qui joue un rôle important (c.f figure 4.9), alors même qu'elle est théoriquement perturbée par les nombreux changements de statuts au cours des itérations. On peut également envisager d'utiliser les spécificités des problèmes considérés et notamment le caractère bi-modal du réseau des forces de contact [126]; on reviendra sur ce point dans les conclusions générales. Mais il convient au préalable d'étendre les approches développées dans les deux derniers chapitres à la modélisation tridimensionnelle; c'est l'objet du chapitre suivant.

Chapitre 5

Modélisation tridimensionnelle

Dans le domaine des milieux granulaires le traitement de problèmes de grande taille en modélisation tridimensionnelle présente un challenge d'autant plus important que la difficulté ne provient pas simplement d'un accroissement du nombre de degrés de liberté par particule, mais également de la loi de frottement qui ne préserve pas le caractère polyédrique des contraintes. Pour NLGS comme pour CPG, ceci a des conséquences sur la mise en oeuvre et même la définition des algorithmes de résolution. Au delà de l'enjeu numérique, la modélisation tridimensionnelle est requise pour appréhender les mécanismes, négligés dans un premier temps dans les simulations comme dans certains dispositifs expérimentaux académiques (tambour tournant), qui pourraient expliquer certains écarts entre expériences et calculs. Il est de plus des matériaux (c.f. conclusion) pour lesquels seul un modèle 3D est pertinent. Les adaptations au cas 3D de la méthode NLGS sont présentées dans une première section ainsi que les performances des différentes implémentations et du traitement parallèle. Les extensions de l'approche CPG au cas 3D présentent des difficultés spécifiques et plusieurs variantes sont proposées et testées dans une deuxième partie.

5.1 NSCD, Newton et Multithreading

Quelque soit le type de problème, bidimensionnel ou tridimensionnel, l'algorithme NLGS présente la même structure globale : une boucle sur les contacts avec une résolution locale du système (à l'échelle du contact) incluant la loi de contact frottant. La différence fondamentale du cas 3D par rapport au 2D réside dans le fait que cette résolution ne peut être en général explicite. Le convexe de frottement ou tranche du cône de Coulomb n'est plus polygonal : il s'agit d'un disque si l'isotropie est supposée. Il est possible alors d'utiliser un solveur classique pour les systèmes non linéaires : la méthode de Newton. Pour l'adapter à la non régularité des lois de contact et frottement, il convient de l'étendre aux opérateurs non différentiables comme les projections présentes dans l'expression du système à résoudre (5.1). La méthode de Newton généralisée a été introduite par Alart et Curnier pour la résolution globale des problèmes de contact frottant [6] ; elle est utilisée ici à la résolution du "petit" système local comme préconisé dans [70]. La présentation de

cette approche est détaillée dans une première section. Pour le contact entre sphères un tel algorithme n'est pas nécessaire, la seconde section est consacrée à la démarche mise en oeuvre.

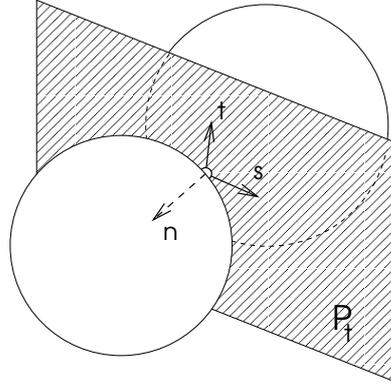


FIG. 5.1 – Repère local pour un contact sphère/sphère : le plan tangent, perpendiculaire à la normale au contact \mathbf{n} , est défini par les deux vecteurs \mathbf{s} et \mathbf{t} .

5.1.1 Cas général : Méthode de Newton Généralisée

Si l'on considère un problème de corps 3D interagissant par contact et frottement, la résolution n'est pas triviale. Une résolution exacte du problème ne peut pas se faire de façon explicite et nécessite l'utilisation d'un algorithme itératif. Pour résoudre des problèmes non linéaires, l'algorithme de Newton est souvent utilisé. Son utilisation pour des problèmes de contact avec/sans frottement entre corps déformables [4] est efficace malgré la non différentiabilité ; des résultats partiels de convergence et de stabilité numérique confortent l'utilisation d'une telle méthode [6]. Nous explicitons l'algorithme sans propriété particulière sur la matrice \mathbb{W} si ce n'est la symétrie et la définie positivité. Ainsi en omettant l'indice de contact α pour alléger l'écriture, on se propose de résoudre le problème élémentaire suivant

$$\begin{cases} \mathbb{W}\mathbf{r} - \mathbf{v} = -\mathbf{v}^{libre}, \\ r_n - proj_{\mathbb{R}^+}(r_n - \rho_n v_n) = 0, \\ \mathbf{r}_t - proj_{\mathcal{C}(\mu r_n)}(\mathbf{r}_t - \rho_t \mathbf{v}_t) = \mathbf{0}, \end{cases} \quad (5.1)$$

Remarque : le système (5.1) est identique au système (3.1) où \mathbf{u} est noté ici \mathbf{v} afin de retrouver la notation \mathbf{u} pour le résidu utilisé au chapitre 4 et dans la deuxième partie de celui-ci. La partie tangentielle de l'impulsion locale possède ici deux composantes, notées r_t et r_s , portées par les deux vecteurs du plan tangent au point de contact et perpendiculaires à la normale au contact. On considère alors par la suite le repère orthonormé direct

$(I; \mathbf{s}, \mathbf{t}, \mathbf{n})$. En 3D, $\mathcal{C}(\mu r_n)$ représente le disque de rayon μr_n . Le système (5.1) revient à la recherche des zéros de la fonction $\mathcal{F}(\mathbf{v}, \mathbf{r})$ où

$$\begin{aligned} \mathcal{F} : \mathbb{R}^3 \times \mathbb{R}^3 &\rightarrow \mathbb{R}^3 \times \mathbb{R}^3 \\ (\mathbf{v}, \mathbf{r}) &\rightarrow \mathcal{F}(\mathbf{v}, \mathbf{r}) \end{aligned} \quad (5.2)$$

définie par,

$$\mathcal{F}(\mathbf{v}, \mathbf{r}) = \mathbf{0} \Leftrightarrow \begin{cases} \mathbf{v} - \mathbf{v}^{libre} - \mathbb{W}\mathbf{r} = \mathbf{0}, \\ \mathbf{r} - \gamma_n^+ \mathbf{n} - \text{proj}(\gamma_t; \mathcal{C}(\gamma_n^+))\mathbf{t} = \mathbf{0}, \end{cases} \quad (5.3)$$

avec $\gamma = \mathbf{r} - \rho\mathbf{v} = \gamma_n \mathbf{n} + \gamma_t$ et $\gamma_n^+ = \text{proj}(\gamma_n; \mathbb{R}^+)$ et $\gamma_t \in \mathbb{R}^2$. Le problème (5.3) peut être écrit sous une forme équivalente à celle des problèmes traités dans [4, 6]. On bénéficie ainsi des propriétés partielles de convergence établies. Pour faciliter l'écriture ainsi que l'étude de cas qui va suivre, on décompose \mathcal{F} en quatre fonctions comme suit,

$$\begin{cases} \mathcal{F}_1(\mathbf{v}, \mathbf{r}) = v_n - v_n^{libre} - W_{nn}r_n - \mathbb{W}_{nt}\mathbf{r}_t & \in \mathbb{R} \\ \mathcal{F}_2(\mathbf{v}, \mathbf{r}) = \mathbf{v}_t - \mathbf{v}_t^{libre} - \mathbb{W}_{tn}\mathbf{r}_t - \mathbb{W}_{tt}\mathbf{r}_t & \in \mathbb{R}^2 \\ \mathcal{F}_3(\mathbf{v}, \mathbf{r}) = r_n - \text{proj}_{\mathbb{R}^+}(r_n - \rho_n u_n) & \in \mathbb{R} \\ \mathcal{F}_4(\mathbf{v}, \mathbf{r}) = \mathbf{r}_t - \text{proj}_{\mathcal{C}(\mu r_n)}(\mathbf{r}_t - \rho_t \mathbf{v}_t) & \in \mathbb{R}^2 \end{cases} \quad (5.4)$$

La méthode de Newton consiste alors à chercher $\mathbf{X} = (\tilde{\mathbf{v}}, \tilde{\mathbf{r}})$ satisfaisant à la condition $\mathcal{F}(\mathbf{X}) = \mathbf{0}$. On définit alors une suite d'itérés $(\mathbf{X}^p)_p$ de la façon suivante,

$$\nabla \mathcal{F}(\mathbf{X}^p) \Delta \mathbf{X} = -\mathcal{F}(\mathbf{X}^p), \quad (5.5)$$

où $\nabla \mathcal{F}(\mathbf{X}^p)$ est la matrice jacobienne de $\mathcal{F}(\mathbf{X}^p)$. Comme la fonction n'est pas différentiable, la matrice jacobienne est remplacée par un des éléments de la base du jacobien généralisé [4]. Formellement, on écrit,

$$\partial \mathcal{F} = \begin{bmatrix} I & \mathbf{0}^t & -W_{nn} & -\mathbb{W}_{nt} \\ \mathbf{0} & \mathbb{I} & -\mathbb{W}_{tn} & -\mathbb{W}_{tt} \\ \partial \mathcal{F}_3 / \partial v_n & \mathbf{0}^t & \partial \mathcal{F}_3 / \partial r_n & \mathbf{0}^t \\ \mathbf{0} & \partial \mathcal{F}_4 / \partial u_t & \partial \mathcal{F}_4 / \partial r_n & \partial \mathcal{F}_4 / \partial r_t \end{bmatrix}. \quad (5.6)$$

ou encore en introduisant l'indice d'itération pour les parties qui en dépendent,

$$\nabla \mathcal{F} = \begin{bmatrix} \mathbb{I} & -\mathbb{W} \\ \mathbb{A}^p & \mathbb{B}^p \end{bmatrix}, \quad (5.7)$$

où \mathbb{I} , \mathbb{A}^p et $\mathbb{B}^p \in \mathbb{R}^3$. En partant de la définition des itérations de Newton,

$$\nabla \mathcal{F}(\mathbf{v}^p, \mathbf{r}^p)(\mathbf{v}^{p+1} - \mathbf{v}^p, \mathbf{r}^{p+1} - \mathbf{r}^p) = -\mathcal{F}(\mathbf{v}^p, \mathbf{r}^p), \quad (5.8)$$

on peut en remplaçant (5.7) dans l'équation (5.8) se ramener au système suivant

$$\begin{cases} \mathbb{W}\mathbf{r}^{p+1} - \mathbf{v}^{p+1} = -\mathbf{v}^{libre}, \\ \mathbb{A}^p(\mathbf{v}^{p+1} - \mathbf{v}^p) + \mathbb{B}^p(\mathbf{r}^{p+1} - \mathbf{r}^p) = -(\mathcal{F}_3(\mathbf{v}^p, \mathbf{r}^p), \mathcal{F}_4(\mathbf{v}^p, \mathbf{r}^p))^T \end{cases} \quad (5.9)$$

équivalent à

$$\begin{cases} \mathbb{W}\mathbf{r}^{p+1} - \mathbf{v}^{p+1} = -\mathbf{v}^{libre}, \\ (\mathbb{A}^p\mathbb{W} + \mathbb{B}^p)(\mathbf{r}^{p+1} - \mathbf{r}^p) = -(\mathcal{F}_3(\mathbf{v}^p, \mathbf{r}^p), \mathcal{F}_4(\mathbf{v}^p, \mathbf{r}^p))^T \end{cases} \quad (5.10)$$

Pour déterminer \mathbb{A}^p , \mathbb{B}^p et ainsi résoudre (5.10), il nous reste à déterminer \mathcal{F}_3 et \mathcal{F}_4 ainsi que leurs dérivées partielles par rapport à r_n , v_n , \mathbf{r}_t et \mathbf{v}_t . Pour pouvoir les déterminer, nous discutons suivant les quantités projetées du système (5.1) et ainsi écrivons dans les différents cas (i.e. en fonction du statut) les fonctions ainsi que leurs dérivées partielles.

Si	$r_n - \rho_n u_n > 0$	alors	
	$\mathcal{F}_3 = \rho_n u_n$	et	$\partial\mathcal{F}_3/\partial u_n = \rho_n \quad \partial\mathcal{F}_3/\partial r_n = 0$
sinon	$\mathcal{F}_3 = r_n$	et	$\partial\mathcal{F}_3/\partial u_n = 0 \quad \partial\mathcal{F}_3/\partial r_n = 1$
Si	$\mathbf{r}_t - \rho_t \mathbf{v}_t \in \mathcal{C}(\mu r_n)$	alors	
	$\mathcal{F}_4 = \rho_t \mathbf{v}_t$	et	$\partial\mathcal{F}_4/\partial \mathbf{r}_n = 0$ $\partial\mathcal{F}_4/\partial \mathbf{v}_t = \rho_t \mathbb{I} \quad \partial\mathcal{F}_4/\partial \mathbf{r}_t = 0$
sinon	$\mathcal{F}_4 = \mathbf{r}_t - \mu r_n \frac{\mathbf{r}_t - \rho_t \mathbf{v}_t}{\ \mathbf{r}_t - \rho_t \mathbf{v}_t\ }$	et	$\partial\mathcal{F}_4/\partial \mathbf{r}_n = -\mu \frac{\mathbf{r}_t - \rho_t \mathbf{v}_t}{\ \mathbf{r}_t - \rho_t \mathbf{v}_t\ }$ $\partial\mathcal{F}_4/\partial \mathbf{v}_t = \rho_t \mu r_n \mathcal{M}(\mathbf{r}_t - \rho_t \mathbf{v}_t)$ $\partial\mathcal{F}_4/\partial \mathbf{r}_t = \mathbb{I} - \rho_t \mu r_n \mathcal{M}(\mathbf{r}_t - \rho_t \mathbf{v}_t)$

où l'application de \mathbb{R}^2 dans $\mathbb{R}^{2 \times 2}$ qui à $\mathbf{V} \rightarrow \mathcal{M}(\mathbf{V})$ est la différentielle de l'application $\mathbf{V}/\|\mathbf{V}\|$ par rapport à $\mathbf{V} = (V_1, V_2)$ égale à

$$\mathcal{M}(V) = \frac{1}{\|\mathbf{V}\|^3} \begin{bmatrix} V_2^2 & -V_1 V_2 \\ -V_2 V_1 & V_1^2 \end{bmatrix}. \quad (5.11)$$

appliquée ici au vecteur de $\mathbf{r}_t - \rho_t \mathbf{v}_t \in \mathbb{R}^2$. Ainsi en procédant à une étude de cas (statuts), nous pouvons déterminer les composantes des matrices \mathbf{A}^p et \mathbf{B}^p , nous permettant de former complètement le gradient de notre application \mathcal{F} . Nous complétons ce schéma avec un test d'arrêt : nous pouvons prendre une extrapolation des tests énergétiques présentés précédemment (voir chapitre 1), appliqués ici à la différence de deux itérés successifs et non pas à la différence d'un itéré Gauss-Seidel et d'un itéré Jacobi sans signification pour ce schéma itératif.

5.1.2 Cas particulier : échantillons de sphères

La méthode de Newton permet de résoudre efficacement les problèmes non-linéaires de contact. Elle a l'avantage d'être générale prenant en compte n'importe quel type de corps.

Toutefois pour ceux possédant de bonnes propriétés géométriques comme les sphères, nous pouvons nous affranchir de cet algorithme de résolution qui n'est pas dans ce cas la méthode la plus rapide. On considère notre problème local (5.1). Dans le cas de contacts entre sphères ou entre sphères et plans, la matrice $\mathbf{W}_{\alpha\alpha}$ égale au produit $\mathbf{H}_\alpha^t \mathbf{M}^{-1} \mathbf{H}_\alpha$ est diagonale. En effet, l'application \mathbf{H}_α étant égale à

$$\mathbf{H}_\alpha = \begin{bmatrix} \mathbf{n}_\alpha & \mathbf{t}_\alpha & \mathbf{s}_\alpha \\ \mathbf{0} & -l_i \mathbf{s}_\alpha & l_i \mathbf{t}_\alpha \\ -\mathbf{n}_\alpha & -\mathbf{t}_\alpha & -\mathbf{s}_\alpha \\ \mathbf{0} & -l_j \mathbf{s}_\alpha & l_j \mathbf{t}_\alpha \end{bmatrix}, \quad (5.12)$$

l_i et l_j étant les rayons respectifs des particules i et j formant le contact α , le produit matriciel précédent nous donne dans le cas du contact sphère/sphère

$$\mathbf{W}_{\alpha\alpha} = \text{diag}(m_{eff,n}^{-1}, m_{eff,t}^1, m_{eff,t}^{-1}) \quad (5.13)$$

où $m_{eff,n}^{-1} = \frac{m_i + m_j}{m_i m_j}$ et $m_{eff,t}^{-1} = \frac{8}{3} m_{eff,n}^{-1}$ en utilisant la valeur de l'inertie d'une sphère pleine par rapport à son centre. Dans le cas de contacts entre sphère et obstacle fixe, la masse m_j dans l'expression de $m_{eff,n}^{-1}$ tend vers l'infini et $m_{eff,n}^{-1} = \frac{1}{m_i}$. Les trois composantes de l'impulsion locale sont donc découplées dans la dynamique réduite. On peut alors exprimer \mathbf{r} de façon explicite en procédant à une étude de cas, revenant à gérer comme pour le cas 2D, le signe des composantes de \mathbf{v}^{free} . Les termes W_{tt} et W_{ss} étant identiques, lorsque l'itéré en cours sort du cône de Coulomb (figure 5.3), il suffit de procéder à une projection radiale sur le disque de rayon μr_n nous fournissant immédiatement la solution. L'algorithme comprenant les tests effectués est contenu dans le tableau 5.2.

Si $u_{n,libre} > 0$	alors $\mathbf{r} = \mathbf{0}$	(statut= séparation)
Sinon		
$r_n = \frac{-u_{n,libre}}{W_{nn}}$		
Si $\ \mathbf{v}_{t,libre}\ < \mu W_{tt} r_n$	alors $\mathbf{r}_t = -\mathbb{W}_{tt}^{-1} \mathbf{v}_{t,libre}$	(statut= adhérent)
Sinon $norm = \mu r_n \frac{W_{tt}}{\ \mathbf{v}_{t,libre}\ }$	et $\mathbf{r}_t = -norm \mathbb{W}_{tt}^{-1} \mathbf{v}_{t,libre}$	(statut= glissement)

FIG. 5.2 – Méthode de résolution explicite pour les problèmes traitant avec des sphères (l'indice α est omis).

Cette méthode donne la même solution que l'algorithme de Newton (à la tolérance près). Ainsi pour des simulations mettant en jeu des sphères elle est plus rapide et ne perturbe pas la solution. Ce procédé n'est évidemment valable que dans le cas de contact sphère/sphère ainsi que pour les contacts sphère/plan.

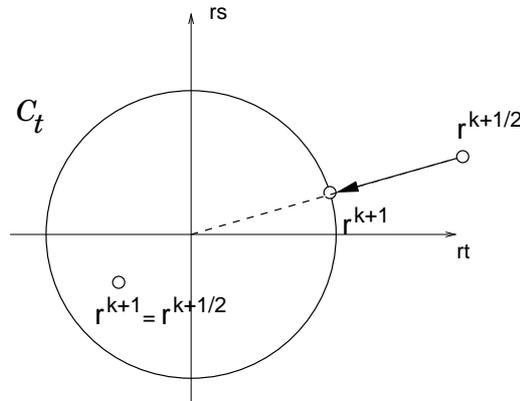


FIG. 5.3 – Cas des sphères : l’opérateur local \mathbb{W}_t étant sphérique, on peut corriger simplement la prédiction de l’impulsion locale en la projetant sur le cône de Coulomb.

5.1.3 Implémentations

Dans le chapitre 3 nous avons détaillé les différentes implémentations de l’algorithme NLGS mises en oeuvre. Le principe des deux implémentations (SDL et ELG) ne dépend nullement de la dimension de notre problème : que celui-ci soit 2D ou 3D, la seule chose qui change est le nombre de termes des matrices locales à calculer (i.e. 9 pour le cas tridimensionnel). Ainsi pour des simulations tridimensionnelles l’espace mémoire occupé par la méthode SDL est beaucoup plus conséquent que pour les simulations bi-dimensionnelles. Pour un même nombre de corps, l’ensemble des données locales, hors matrices blocs \mathbb{W} , est multiplié par 1.5 et l’espace de stockage des matrices est multiplié par plus de 2. Nous avons comparé comme pour les problèmes bi-dimensionnels le rapport du coût d’une itération ELG sur celui d’une itération SDL en fonction du nombre de processeurs utilisés et ce sur deux échantillons (figure 5.4). Contrairement aux résultats en 2D (c.f. figure 3.13), la procédure ELG s’avère rapidement (quand le nombre de corps et le nombre de processeurs augmentent) moins consommatrice de temps de calcul. Les valeurs pour 0 processeurs correspondent à une exécution séquentielle sans directives OpenMP alors que pour 1 processeur, les directives sont conservées.

Pour le petit échantillon ($n_c = 3100$), le rapport Tt , défini au chapitre 3, passe en dessous la valeur seuil de 1 dès l’utilisation de 4 processeurs soulignant la supériorité de la méthode ELG sur SDL, en remarquant que les valeurs de Tt pour 1 et 2 processeurs ne dépasse pas 1.5. L’évolution au-delà de 4 reste constante et toujours en dessous de 1. Le passage à un échantillon plus conséquent ($n_c = 8100$) ramène directement les valeurs de Ts en dessous de 1 même pour une utilisation séquentielle, avec une accentuation du gain pour un calcul parallèle important : pour $P = 16$, Ts est égal à 0.25. On peut également noter que la performance algorithmique est beaucoup plus chahutée que lors d’une exécution bi-dimensionnelle puisque l’on atteint des surplus ou des gains d’itérations de l’ordre de 30%. On retrouve ainsi les observations faites en 2D : diminution de $Ts(0)$ avec l’augmentation

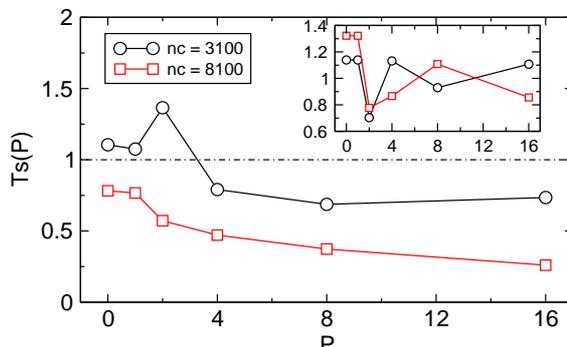


FIG. 5.4 – Evolution du rapport coût d’une itération ELG / coût d’une itération SDL pour des simulations tri-dimensionnelles : la méthode ELG apparaît vite comme étant la plus efficace. En encart évolution de la performance algorithmique.

de la taille des échantillons et diminution de T_s lors d’une exécution parallèle. Pour des problèmes de grande taille, il est donc plus intéressant d’effectuer plus de calculs par des allers-retours entre les niveaux local et global (ELG) que de stocker la matrice \mathbb{W} (SDL). Le phénomène de ”*swap*” lorsque toutes les données ne peuvent être stockées en mémoire centrale vient handicaper l’approche SDL.

5.1.4 Multithreading

La technique parallèle développée pour les simulations bi-dimensionnelles s’étend sans problème, au même titre que les deux techniques d’implémentation, aux problèmes tridimensionnels. La résolution du problème local restant locale, les processeurs vont pouvoir continuer à travailler sur des contacts différents sans se soucier réellement du travail effectué par le processeur voisin. Il est évident que les comparaisons, discussions, et mise en garde vues au chapitre 3 restent valables, notamment le problème de ”*race condition*”. L’observation de la figure 5.5 nous permet de retrouver également le fait que la méthode ELG se comporte mieux en parallèle que la méthode SDL, résultat attendu après la lecture de la figure 5.4. La méthode ELG est toujours plus performante avec l’augmentation du nombre de contacts dans les échantillons.

Nous avons effectué des dépôts sous pesanteur avec 1350 et 3530, sous *SGI Origin3800* en utilisant respectivement 1, 2, 4, 8 et 16 processeurs. Les quantités comme le speed-up relatif et la performance ont été calculées suivant les définitions introduites dans le chapitre 3 pour chaque technique d’implémentation. Le speed-up de la méthode ELG, au même titre que les simulations 2D, devient plus important avec la taille de l’échantillon. Cependant la performance algorithmique est plus chahutée, probablement à cause de la grande largeur de bande des matrices \mathbb{W} lors de nos simulations : pour des simulations de dépôts sous pesanteur, le nombre de contacts sphère/plan devient plus important que le nombre de contact disque/plan. Ainsi l’algorithme Gauss-Seidel est plus perturbé que

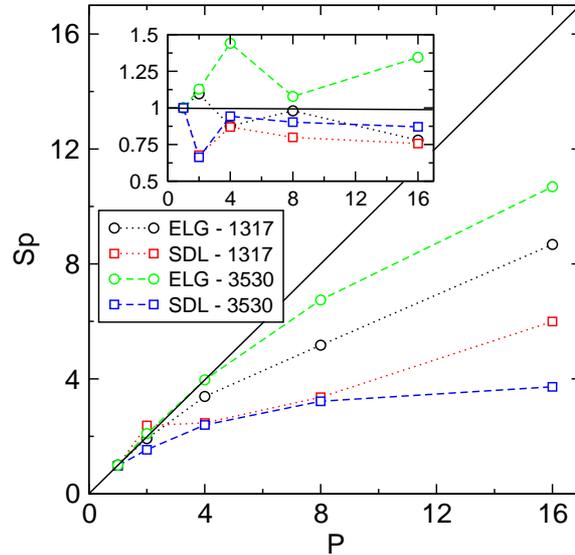


FIG. 5.5 – Evolution du Speed-Up des méthodes ELG et SDL sous *SGI Origin3800*. En encart évolution de la performance algorithmique p_a .

lors des simulations 2D. La taille importante des données à stocker localement détériore complètement les temps de calcul de la méthode SDL, et ceci même lors d'exécution en parallèle. Le speed-up suit alors une évolution contraire à celui-observé pour des simulations bi-dimensionnelles : il devient moins intéressant avec l'augmentation du nombre de particules. La performance algorithmique est très perturbée pour la méthode ELG pouvant atteindre un surplus de 40%.

5.2 Extension du GCP au 3D

Pour une modélisation bidimensionnelle, la stratégie du gradient projeté conjugué, bénéficie du caractère polyédrique de l'ensemble des contraintes. C'est ainsi que la projection des gradients sur le cône tangent revient à chercher une solution provisoire (itéré) dans un ensemble de contraintes actives que l'on peut identifier à un sous espace vectoriel support d'une facette ou d'une arête de l'ensemble des contraintes (comme du cône tangent). Si l'itéré ne sort pas de la facette ou arête, il reste alors dans l'ensemble des solutions admissibles sans avoir à être corrigé. Dans une modélisation tridimensionnelle et frottement isotrope, une telle propriété est perdue car l'ensemble des contraintes n'est pas polyédrique comme nous le verrons dans la première section où est conservée dans le schéma CPG la projection des gradients sur le cône tangent même si ce dernier n'approche pas "suffisamment bien" localement l'ensemble des contraintes. Une alternative consiste à approcher le cône de Coulomb par une pyramide à base polygonale. Cette voie a été explorée par Klarbring [83], Pang et Trinkle [120] et Haslinger *et al.* [59] dans le

but de récupérer les solveurs directs des Problèmes de Complémentarité Linéaire (Linear Complementarity Problem - LCP). La seconde section est consacrée à cette approche permettant une extension relativement immédiate du schéma bi-dimensionnel du CPG. Les inconvénients de ces deux stratégies conduisent à définir et tester un nouveau schéma hybride comprenant des ingrédients des algorithmes CPG et NLGS.

5.2.1 Une première approche

Une première stratégie consiste à appliquer le schéma numérique invoquant la projection des gradients sur le cône tangent, qui est parfaitement défini, "sans se poser de questions". Si l'on s'en pose on s'attend bien sûr à "sortir" systématiquement de l'ensemble des contraintes : sur un seul contact et dans le plan tangent à ce contact l'ensemble des contraintes à l'impulsion tangentielle est un disque de rayon provisoire μr_n^k . Le cône tangent à $\mathcal{C}(\mu r_n^k)$ en \mathbf{r}_t^k est un demi-espace ; si le gradient \mathbf{u}^k n'appartient pas à $T_{\mathcal{C}}$, il est projeté sur la droite séparatrice du demi-espace. L'itéré provisoire $\mathbf{r}_t^{k+\frac{1}{2}}$ a alors toutes les chances de "sortir" du convexe \mathcal{C} et sera donc quasi systématiquement corrigé. Cette perturbation du schéma est théoriquement préjudiciable au bon comportement du gradient conjugué, la conjugaison n'opérant plus dans le même sous espace vectoriel, comme c'est le cas en dimension 2D où ce sous espace est celui des contraintes actives $A(\mathbf{w}^k)$. Cette dernière notion de contraintes actives n'est plus opérante en 3D. Il est vrai que, même dans le cas 2D, à cause du caractère non associé du contact frottant de Coulomb, l'itéré, s'il est de statut *glissant* (*avant* ou *arrière*), doit être systématiquement corrigé afin d'adapter le seuil de glissement μr_n en fonction de l'impulsion normale ; ceci conduit même à "projeter" $\mathbf{r}^{k+\frac{1}{2}}$ de l'intérieur sur le bord du cône de Coulomb (c.f. figure 4.6). Mais cette perturbation ne modifie pas les contraintes actives $A(\mathbf{w}^k)$ sur lesquelles projeter les gradients, tant que le statut reste *glissant* ($l = k, k+1, \dots, k+p$) ; la conjugaison reste donc opérationnelle dans $A(\mathbf{w}^k)$ pour $l = k, k+1, \dots, k+p$. Comme dit précédemment, il n'en est pas de même en modélisation tridimensionnelle. Une dernière remarque : la projection de $\mathbf{r}^{k+\frac{1}{2}}$ sur le bord du cône quand $\mathbf{r}^{k+\frac{1}{2}}$ est à l'intérieur est naturellement effectuée radialement.

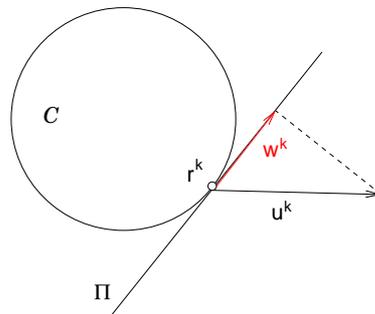


FIG. 5.6 – Projection sur le plan tangent en \mathbf{r}^k à $\mathcal{C}(\mu r_n)$

5.2.2 Approximation du cône de Coulomb

En alternative à la première approche, une plus classique consiste à approcher le cône de Coulomb par un cône pyramidal à base polygonale. Cette approche est utilisée pour la résolution de problèmes de contacts frottants tridimensionnels en utilisant généralement une base carrée [120] ou plus complexe [59]. L'extension du CPG aux problèmes tridimensionnels devient alors immédiate, mais ne nous permet plus de prendre en compte du frottement isotrope. Mais notre but premier étant de traiter des problèmes possédant un grand nombre de contacts, nous supposons que cette perte de propriété locale n'aura pas ou peu d'influence sur le comportement global du milieu. Les procédures développées au chapitre 4 pour un problème bi-dimensionnel (i.e. projection des gradients et correction de l'itéré) peuvent donc être également mises en place : nous devons alors définir correctement la projection des composantes des gradients et la correction de l'itéré dans le plan tangent. Pour cela deux composantes tangentielles sont considérées, "T" et "S", liées respectivement aux composantes r_T et r_S de l'impulsion tangentielle locale \mathbf{r}_t . Nous définissons ainsi un statut tangentiel différent pour chacune des deux composantes et les discussions des différentes procédures seront étendues à cette double information contenue par le statut.

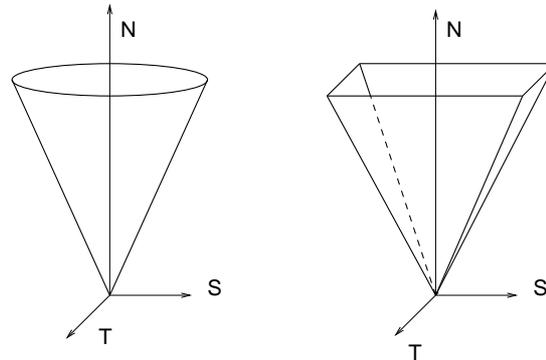


FIG. 5.7 – Utilisation d'un cône pyramidal à base carrée pour approcher le cône de Coulomb.

Correction de l'itéré

Le traitement de la composante normale nous permettant d'actualiser le seuil de glissement reste inchangé. Si $r_{\alpha,n}^{k+\frac{1}{2}}$ est négatif l'impulsion locale est mise à zéro correspondant à un statut de "séparation". Dans le cas contraire le contact est actif et il nous

Si $r_{\alpha,n}^{k+\frac{1}{2}} < 0$	alors $\mathbf{r}_{\alpha}^{k+1} = \mathbf{0}$	$(statut_{\alpha}^{k+1} = \text{séparation})$
sinon		
$r_{\alpha,n}^{k+1} = r_{\alpha,n}^{k+\frac{1}{2}}$	$\epsilon_T = sign(r_{\alpha,T}^{k+\frac{1}{2}})$	$\epsilon_S = sign(r_{\alpha,S}^{k+\frac{1}{2}}) \quad s_{\alpha}^{k+1} = \mu r_{\alpha,n}^{k+1}$
Si $ r_{\alpha,T}^{k+\frac{1}{2}} > s_{\alpha}^{k+1}$	alors $r_{\alpha,T}^{k+1} = \epsilon_T s_{\alpha}^{k+1}$	$(statut_{\alpha}^{k+1} = \text{glissement } \mathbf{T}\epsilon_T)$
sinon		
si $p_{\alpha,T}^k = 0$	alors $r_{\alpha,T}^{k+1} = \epsilon_T s_{\alpha}^{k+1}$	$(statut_{\alpha}^{k+1} = \text{glissement } \mathbf{T}\epsilon_T)$
sinon	$r_{\alpha,T}^{k+1} = r_{\alpha,T}^{k+\frac{1}{2}}$	$(statut_{\alpha}^{k+1} = \text{adhérent } \mathbf{T})$
Si $ r_{\alpha,S}^{k+\frac{1}{2}} > s_{\alpha}^{k+1}$	alors $r_{\alpha,S}^{k+1} = \epsilon_S s_{\alpha}^{k+1}$	$(statut_{\alpha}^{k+1} = \text{glissement } \mathbf{S}\epsilon_S)$
sinon		
si $p_{\alpha,S}^k = 0$	alors $r_{\alpha,S}^{k+1} = \epsilon_S s_{\alpha}^{k+1}$	$(statut_{\alpha}^{k+1} = \text{glissement } \mathbf{S}\epsilon_S)$
sinon	$r_{\alpha,S}^{k+1} = r_{\alpha,S}^{k+\frac{1}{2}}$	$(statut_{\alpha}^{k+1} = \text{adhérent } \mathbf{S})$

TAB. 5.1 – Correction de l’itéré pour l’approximation du cône de Coulomb par une pyramide à base carrée.

faut déterminer si il s’agit d’un contact ”*glissant*” ou ”*adhérent*”. Pour cela nous allons tester si l’itéré est à l’extérieur du carré $\mathcal{B}^{k+1} = \mu r_n^{k+1}[-1, 1] \times \mu r_n^{k+1}[-1, 1]$. Si il se trouve à l’extérieur, il est projeté sur \mathcal{B} . Cette projection est effectuée en traitant séparément les deux composantes (c.f. figure 4.6 du chapitre 4). Le statut ”*adhérent*” distingue les états ”*adhérentT*” et ”*adhérentS*”. Le statut ”*glissement*” est quant à lui décomposé en {”*glissementT*”, ”*glissementS*”}, $\epsilon_T = \pm$ et $\epsilon_S = \pm$. Comme pour le cas bi-dimensionnel, l’itéré peut être projeté sur la frontière de \mathcal{B}^{k+1} même si il se trouve à l’intérieur de celui-ci. C’est le cas si le statut précédent était glissant et que le déplacement s’est effectué sur le bord du cylindre de base \mathcal{B}^k . Le tableau 5.1 détaille les différentes procédures.

Projection des gradients

La projection des différents gradients revient à effectuer de simple projection sur le cylindre de base \mathcal{B} . Lorsque l’itéré courant possède un statut *séparation*, les différents gradients sont mis à zéro si leur composante normale est négative ; ils sont conservés dans l’autre cas. Lorsque l’itéré à un statut *adhérent*, les gradients restent également inchangés. Lorsque l’itéré est caractérisé par un statut *glissement*, il convient alors de projeter les gradients sur l’enveloppe de \mathcal{B} si ces derniers s’en éloignent. Ces projections s’avèrent être une combinaison de différentes projections : celle illustrée par la figure 4.7 du chapitre 4 réalisée ici dans le plan (T, n) ou (S, n) suivant le statut considéré, et celle détaillée sur la figure 5.8 restreinte au plan (T, S) . Le tableau 5.2 explicite les différentes projections.

si $statut_{\alpha}^k = \mathbf{séparation}$ alors	si $u_{\alpha,n}^k \leq 0$ alors $\mathbf{w}_{\alpha}^k = \mathbf{0}$
	sinon $\mathbf{w}_{\alpha}^k = \mathbf{u}_{\alpha}^k$
si $statut_{\alpha}^k = \mathbf{adhérent}$ alors	$\mathbf{w}_{\alpha}^k = \mathbf{u}_{\alpha}^k$
sinon	
$w_{\alpha,n}^k = u_{\alpha,n}^k$	
si $statut_{\alpha}^k = \mathbf{glissementT+}$	alors $w_{\alpha,T}^k = \min(0, u_{\alpha,T}^k)$
sinon si $statut_{\alpha}^k = \mathbf{glissementT-}$	alors $w_{\alpha,T}^k = \max(0, u_{\alpha,T}^k)$
sinon	$w_{\alpha,T}^k = u_{\alpha,T}^k$
si $statut_{\alpha}^k = \mathbf{glissementS+}$	alors $w_{\alpha,S}^k = \min(0, u_{\alpha,S}^k)$
sinon si $statut_{\alpha}^k = \mathbf{glissementS-}$	alors $w_{\alpha,S}^k = \max(0, u_{\alpha,S}^k)$
sinon	$w_{\alpha,S}^k = u_{\alpha,S}^k$

TAB. 5.2 – Projection des gradients sur l'approximation du cône de Coulomb.

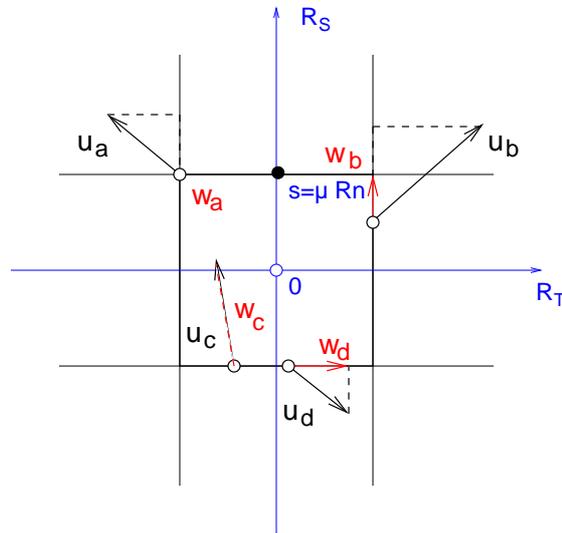


FIG. 5.8 – Section tangente perpendiculaire à l'axe normal.

5.2.3 Schéma hybride

Etant donné les inconvénients de l'algorithme CPG standard mentionnés en section 5.2.1, le schéma dit hybride présenté ici emprunte des éléments des algorithmes NLGS et CPG. Nous considérons, pour un seul contact, le disque de rayon provisoire μr_n^k , et non une approximation polygonale, qui définit donc l'ensemble des impulsions tangentielles admissibles. Les grandeurs tangentes sont définies par

$$\begin{cases} \mathbf{r}_t = \mathbf{r} - r_n \mathbf{n} \\ \mathbf{u}_t = \mathbf{u} - u_n \mathbf{n} \\ \dots \end{cases},$$

et on rappelle que, dans le cas particulier des sphères, $\mathbb{W}_{\alpha\alpha} = \text{diag}(W_{\alpha\alpha,nn}, W_{\alpha\alpha,tt}, W_{\alpha\alpha,tt})$ et $\mathbb{W}_{\alpha\alpha,tt} = W_{\alpha\alpha,tt}\mathbb{I}$. Dans cette nouvelle approche si la composante tangentielle de \mathbf{u}^k n'appartient pas à T_C , elle n'est pas projetée sur la droite séparatrice du demi-espace comme présenté sur la figure 5.6 mais mise à zéro (c.f. figure 5.9) ; la composante normale de \mathbf{u}^k est elle conservée permettant d'adapter le rayon du prochain disque de frottement. Le tableau 5.3 détaille la projection des gradients.

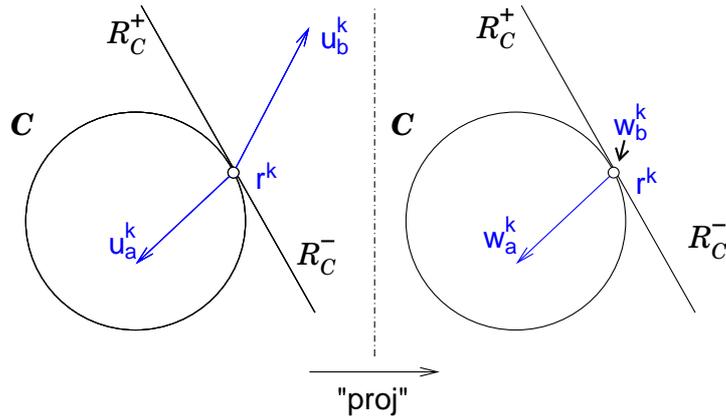


FIG. 5.9 – Projection des gradients : lorsque le statut est glissant, dans le cas b, seule la composante normale des gradients est conservée (les composantes tangentielles sont mises à zéro) ; dans le cas a, le gradient reste inchangé.

Quant à la correction $\mathbf{r}^{k+\frac{1}{2}} \rightarrow \mathbf{r}^{k+1}$, on ne peut plus se contenter de la projeter radialement sur le bord du cône, car l'annulation des composantes tangentielles du gradient aboutirait à figer artificiellement la direction de glissement. Pour éviter cette situation nous utilisons une procédure proche de la stratégie NLGS, mais qui s'apparente plus à une stratégie Jacobi non linéaire (NLJac), en résolvant explicitement le problème local (5.1) cependant restreint aux composantes tangentielles ; la composante normale étant elle définie comme elle l'était jusqu'à présent. Ainsi dans le cas particulier où l'on se déplace sur le bord du

si $statut_\alpha^k = \mathbf{séparation}$ alors	si $u_{\alpha,n}^k \leq 0$ alors $\mathbf{w}_\alpha^k = \mathbf{0}$
si $statut_\alpha^k = \mathbf{adhérent}$ alors	sinon $\mathbf{w}_\alpha^k = \mathbf{u}_\alpha^k$
sinon (<i>i.e.</i> si $statut_\alpha^k = \mathbf{glissement}$)	$\mathbf{w}_\alpha^k = \mathbf{u}_\alpha^k$
$w_{\alpha,n}^k = u_{\alpha,n}^k$ et on définit $\mathbf{e}_t = \frac{\mathbf{r}_t}{\ \mathbf{r}_t\ }$	
si $\mathbf{u}_{\alpha,t}^k \cdot \mathbf{e}_t < 0$	alors $\mathbf{w}_{\alpha,t}^k = \mathbf{u}_{\alpha,t}^k$
sinon	$\mathbf{w}_{\alpha,t}^k = \mathbf{0}$

TAB. 5.3 – Projection des gradients pour le schéma hybride.

cylindre, pour déterminer \mathbf{r}_t nous résolvons le problème suivant,

$$\begin{cases} \mathbb{W}_{\alpha\alpha}\mathbf{r}_\alpha^{k+1} + \mathbf{u}_\alpha^{k+1} = \mathbf{b}_\alpha - \sum_{\beta \neq \alpha} \mathbb{W}_{\alpha\beta}\mathbf{r}_\beta^k \\ \text{Loi}_\alpha(\mathbf{r}_\alpha^{k+1}, \mathbf{u}_\alpha^{k+1}) = \text{vraie} \end{cases}, \quad (5.14)$$

où \mathbf{u}^k est le résidu ($= \mathbf{b} - \mathbb{W}\mathbf{r}^k$) égal à $-\mathbf{v}^k$ représentant la vitesse relative notée \mathbf{u}^k dans les chapitres 1 et 3 consacrés à la méthode NLGS. Dans ce schéma hybride les composantes normales étant déterminées par l'algorithme CPG, le système (5.14) se réduit aux composantes tangentielles contact par contact,

$$\begin{cases} \mathbb{W}_{\alpha\alpha,tt}\mathbf{r}_{\alpha,t}^{k+1} + \mathbf{u}_{\alpha,t}^{k+1} = \mathbf{b}_{\alpha,t} - \sum_{\beta \neq \alpha} (\mathbb{W}_{\alpha\beta}\mathbf{r}_\beta^k)_t = \mathbf{b}_{\alpha,t}^k \\ \mathbf{r}_{\alpha,t}^{k+1} - \text{proj}_{\mathcal{C}(\mu r_{\alpha,n}^{k+1})}(\mathbf{r}_{\alpha,t}^{k+1} + \rho_t \mathbf{u}_{\alpha,t}^k)^{k+1} = \mathbf{0} \end{cases}, \quad (5.15)$$

où par définition de \mathbf{u}^k et de $\mathbb{W}_{\alpha\alpha}$ pour le cas particulier des sphères, nous obtenons,

$$\begin{aligned} \mathbf{u}_{\alpha,t}^k &= \mathbf{b}_{\alpha,t} - \sum_{\beta} (\mathbb{W}_{\alpha\beta}\mathbf{r}_\beta^k)_t \\ &= \mathbf{b}_{\alpha,t}^k - (\mathbb{W}_{\alpha\alpha}\mathbf{r}_\alpha^k)_t \\ &= \mathbf{b}_{\alpha,t}^k - \mathbb{W}_{\alpha\alpha,tt}\mathbf{r}_{\alpha,t}^k \end{aligned}$$

d'où

$$\mathbf{b}_{\alpha,t}^k = \mathbf{u}_{\alpha,t}^k + \mathbb{W}_{\alpha\alpha,tt}\mathbf{r}_{\alpha,t}^k. \quad (5.16)$$

La résolution du problème tangentiel local (5.15) nous fournit alors la valeur de la composante tangentielle correspondant à la valeur r_n^{k+1} ainsi que le statut pouvant être *glissant* ou *adhérent*. Rappelons que cette résolution locale n'est effectuée que si l'on s'est déplacé le long du cylindre. On trouve dans le tableau 5.4 un résumé synthétique de la correction de l'itéré.

si $r_{\alpha,n}^{k+\frac{1}{2}} < 0$ alors	$\mathbf{r}_{\alpha}^{k+1} = \mathbf{0}$	($statut_{\alpha}^{k+1} = \text{séparation}$)
sinon		
$r_{\alpha,n}^{k+1} = r_{\alpha,n}^{k+\frac{1}{2}}$ et $s_{\alpha}^{k+1} = \mu r_{\alpha,n}^{k+1}$		
si $\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ > s_{\alpha}^{k+1}$ alors	$\mathbf{r}_{\alpha,t}^{k+1} = \frac{s_{\alpha}^{k+1}}{\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ } \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}$	($statut_{\alpha}^{k+1} = \text{glissement}$)
sinon		
si $\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ > s_{\alpha}^{k+1}$ alors	$\mathbf{r}_{\alpha,t}^{k+1} = \frac{s_{\alpha}^{k+1}}{\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ } \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}$	($statut_{\alpha}^{k+1} = \text{glissement}$)
sinon		
si $p_{\alpha,t}^k = 0$ alors		
$\mathbf{b}_{\alpha,t} = \mathbf{u}_{\alpha,t}^k + \mathbb{W}_{\alpha\alpha,tt} \mathbf{r}_{\alpha,t}^k$ et	$\mathbf{r}_{\alpha,t}^{k+\frac{1}{2}} = \mathbb{W}_{\alpha\alpha,tt}^{-1} \mathbf{b}_{\alpha,t}^k$	
si $\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ < s_{\alpha}^{k+1}$ alors	$\mathbf{r}_{\alpha,t}^{k+1} = \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}$	($statut_{\alpha}^{k+1} = \text{adhérent}$)
sinon	$\mathbf{r}_{\alpha,t}^{k+1} = \frac{s_{\alpha}^{k+1}}{\ \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}\ } \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}$	($statut_{\alpha}^{k+1} = \text{glissement}$)
sinon	$\mathbf{r}_{\alpha,t}^{k+1} = \mathbf{r}_{\alpha,t}^{k+\frac{1}{2}}$	($statut_{\alpha}^{k+1} = \text{adhérent}$)

TAB. 5.4 – Correction de l'itéré associée au schéma hybride.

5.3 Tests numériques

Afin d'apprécier l'efficacité des différentes projections, nous comparons l'évolution des différents critères de convergence pour chacune des stratégies. A titre indicatif, l'évolution des critères de convergence du solveur NLGS est également représentée sur les différents graphes.

5.3.1 Cycles

Parmi les différentes méthodes mises en œuvre, la première approche possède l'avantage d'être simple, mais ne donne la majeure partie du temps pas de résultats concluants. En effet sur la plupart des cas étudiés, ce type de projection conduit à la divergence de l'algorithme, l'évolution des différents critères étant stationnaire. Une étude fine sur un cas simple (une bille reposant sur 3 autres dans une boîte) nous a permis de mettre en évidence un (le ?) phénomène venant perturber la convergence. Pour certains statuts glissants, l'algorithme cycle entre deux valeurs (toutes deux ayant le même statut glissant) suivant le procédé mis en évidence par la figure 5.10. Pour cette raison, ce type de projection ne peut être retenu et ne sera pas utilisé pour les comparaisons qui suivent.

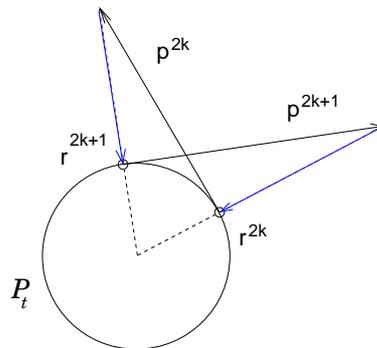


FIG. 5.10 – Manifestation d'un cycle d'ordre deux venant perturber la convergence.

5.3.2 Sur un pas de temps

Dans ce contexte tridimensionnel nous nous sommes restreint à l'étude sur un pas de temps. En effet à la vue des résultats entre les implémentations ELG et SDL, ils ne semble pas très opportun de lancer des processus sur plusieurs pas de temps pour des échantillons de grande taille. Le stockage des matrices pouvant venir perturber le temps de calcul, il n'est pas intéressant de comparer les méthodes connaissant cette influence. Nous présentons ici les résultats effectués sur deux échantillons comportant 1 500 et 3 500 sphères.

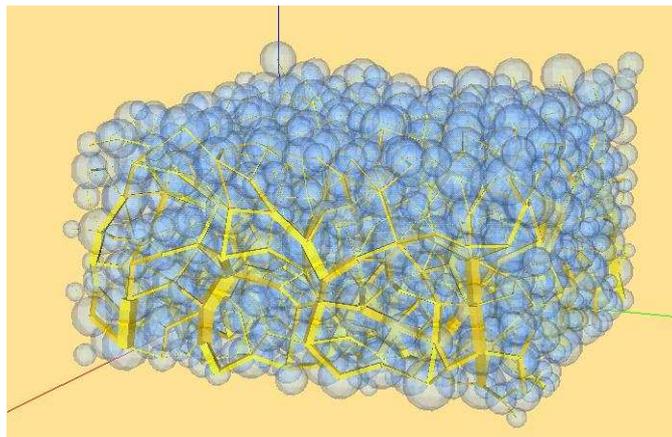


FIG. 5.11 – Echantillon test tridimensionnel composé de 1 500 sphères reposant sous gravité.

Contact sans frottement

Lorsque le frottement n'est pas pris en compte, notre problème est un vrai problème d'optimisation au même titre que les problèmes sans frottement bidimensionnels. Sous ces hypothèses nous attendons de l'algorithme un bon comportement puisqu'aucune approximation ou pseudo projection n'est utilisée.

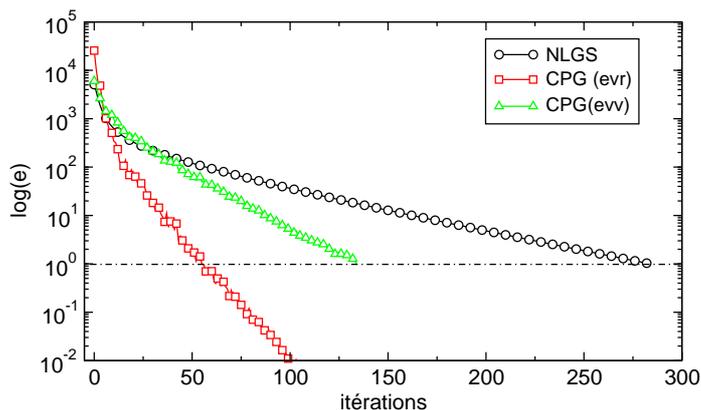


FIG. 5.12 – Evolution des différents critères sur un pas de temps pour un échantillon de 1500 sphères

Les courbes de convergence qui sont présentées sur la figure 5.12 confirment nos attentes. L'algorithme NLGS possède toujours une convergence régulière mais lente : en témoigne l'allure du logarithme de l'erreur. L'algorithme CPG, pour lequel dans le cas sans frottement les différents types de projections sont équivalents, possèdent une convergence rapide mais également peu chahuté. Ce bon comportement se répercute sur le rapport du nombre d'itérations NLGS sur CPG supérieur à 2.

Lorsque le frottement est activé, les résultats sont plus mitigés. Sur les deux échantillons nous avons effectués des simulations en faisant varier le coefficient de frottement μ pour des valeurs comprises entre 0 et 0.8. Nous mesurons le *gain* défini comme le rapport entre le nombre d'itérations de l'algorithme NLGS sur celui du CPG. La figure 5.14 résume l'ensemble des simulations. La figure 5.13 représente l'évolution des critères de convergence pour l'échantillon de 3500 sphères en fonction du frottement entre particules. On retrouve des résultats analogues aux résultats 2D.

Le premier constat est que, sur la gamme des simulations effectuées, le gain ne dépasse pas 3, contrairement aux simulations 2D où le gain peut atteindre 9. Pour de petits échantillons, en utilisant les deux méthodes, le gain est souvent inférieur à 1, en particulier pour $\mu = 0.6$ et $\mu = 0.8$. Cependant, lorsque le nombre de particules augmente, le gain devient plus important et même pour de forts coefficients de frottement. La méthode hybride semble également mieux se comporter que la méthode pyramidale lorsque le nombre

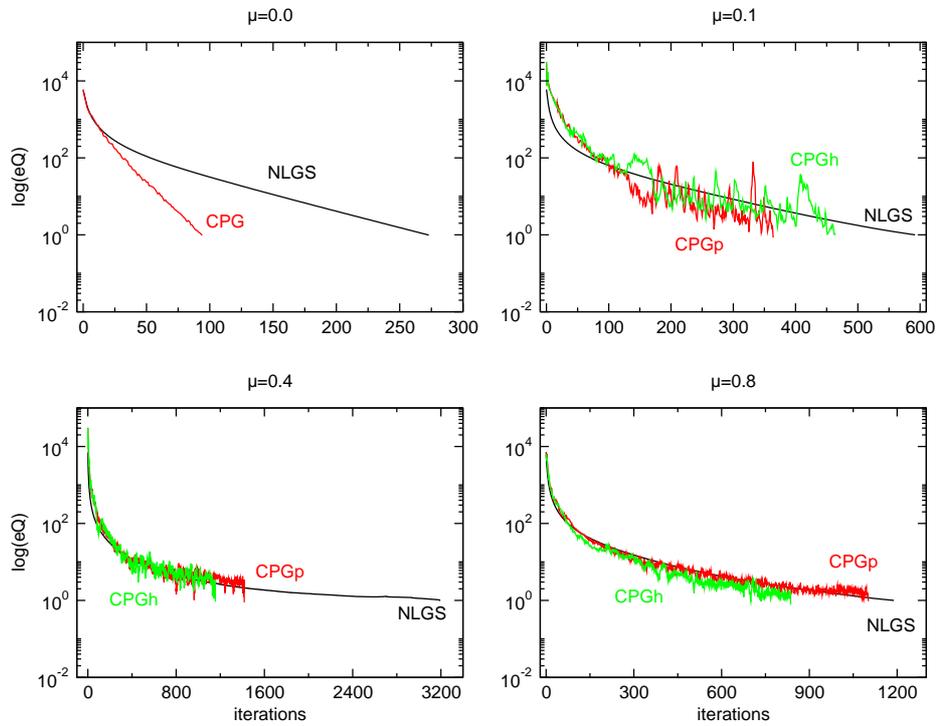


FIG. 5.13 – Evolution des critères de convergence pour l'échantillon de 3500 sphères en fonction du frottement entre particules.

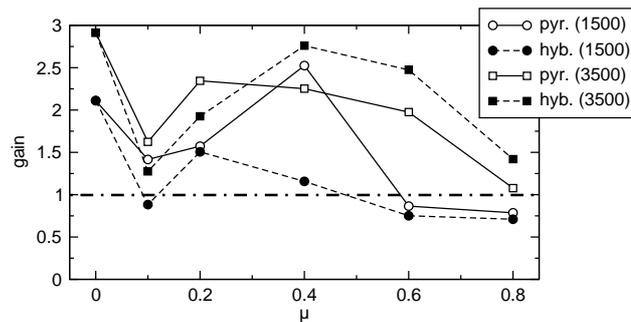


FIG. 5.14 – Evolution du gain pour les échantillons de 1500 et 3500 sphères en fonction du frottement entre particules.

de particules devient plus conséquent. Ces résultats moins probants qu'en 2D ne sont pas surprenants : les approximations et "projections" ne permettant plus de traiter le problème exact, nous nous attendions à certaines perturbations. Toutefois celles-ci semblent ne pas

être si importantes et une étude plus complète doit être effectuée.

5.4 Conclusion

En modélisation tridimensionnelle la difficulté provient essentiellement du caractère non polyédrique des contraintes associées à la loi de frottement. Cependant à travers les approches présentées, autant pour NLGS que pour CPG, cette difficulté peut être détournée en se servant d'approximation des contraintes ou de particularités géométriques des corps en contact. Les résultats numériques comparatifs moins prononcés que pour une modélisation bi-dimensionnelle, reste en faveur des méthodes de type gradient. Pour une amélioration de ces algorithmes nous renvoyons à la conclusion générale qui présentera des axes possibles d'études.

Troisième partie

Applications : entre fluide et solide

Les applications liées aux milieux granulaires sont nombreuses et ont été largement étudiées aussi bien expérimentalement que numériquement [79]. Nous en avons évoqué certaines lors des premiers chapitres : la dynamique des maçonneries [1], les évolutions géophysiques [42, 86], le stockage et la vidange des silos de blé [52], le comportement d'un ballast au passage répété d'un train [139, 140]. On peut encore citer les expériences d'écoulements réalisées sur plan incliné [122, 143, 145] ou en tambour tournant [18, 68, 156], et les essais de cisaillement annulaire [64, 102]. Ces recherches ont toute pour but de comprendre le comportement des granulats tantôt proche du liquide tantôt proche du solide.

Les deux approches communément utilisées (expérimentale et numérique) semblent souvent contradictoires mais sont bel et bien complémentaires pour comprendre les mécanismes des matériaux granulaires. Les expériences permettent de travailler avec le milieu réel mais ne sont pas capables de rapporter des quantités locales comme les forces de contacts. Les simulations utilisent des systèmes idéalisés mais permettent d'obtenir l'évolution des différentes informations locales. Ces dernières ont souvent besoin de retrouver des résultats similaires à ceux obtenus expérimentalement afin de "valider" l'approche utilisée. Parmi les comparaisons expérience/simulation qui ont déjà été réalisées, citons les travaux de Lanier et Jean [88], Lanier et Radjai [89], Burbridge et Braun [25].

Cette partie est consacrée aux investigations numériques réalisées sur deux types d'applications mettant en jeu des granulats denses : le tambour tournant et la boîte à sable (ou "sandbox"). La première partie est liée à l'étude des écoulements granulaires dont la compréhension est toujours une des préoccupations majeures des physiciens et mécaniciens [101]. Simuler des écoulements en tambour tournant ne nécessite pas nécessairement un grand nombre de particules pour des simulations bidimensionnelles (moins de 8 000) mais le processus de simulation peut demander un grand nombre de pas de temps pour atteindre des conditions d'écoulement satisfaisantes : les simulations en tambours tridimensionnels nécessitent entre 10 000 et 50 000 particules pour reproduire les expériences devenant encore plus coûteuses. La seconde application concerne l'évolution quasi-statique de milieux granulaires puisqu'elle fait référence à l'étude du mouvement des plaques tectoniques. Les simulations de boîtes à sables demandent un grand nombre de particules pour observer les premiers phénomènes intéressants (de 10 000 à 100 000 suivant le type de géométrie) sans avoir forcément un grand nombre de pas de temps : pour les boîtes à sable très allongées

ce n'est plus forcément vrai. Quoi qu'il en soit, la nécessité d'avoir des méthodes rapides et robustes se fait ressentir. Ainsi grâce aux différentes optimisations (informatiques et/ou algorithmique) réalisées nous sommes armés pour effectuer des simulations en des temps raisonnables afin de procéder à une étude numérique de ces différents milieux.

Chapitre 6

Écoulements granulaires

Nous allons dans cette partie étudier la rhéologie des écoulements de surface. Les simulations numériques réalisées mettent en jeu, des disques pour le cadre 2D et des sphères en 3D, faiblement polydisperses, confinés dans un tambour tournant. Nous pouvons ainsi suivre l'évolution de quantités telles que les forces de contacts non accessibles expérimentalement. A notre connaissance, peu de résultats via l'approche NSCD apparaissent dans la littérature alors que l'on peut trouver de nombreux travaux utilisant MD [132] ou DEM [96, 133]. Il nous paraît intéressant d'effectuer de telles simulations à l'aide de notre approche pour valider celle-ci dans le cas d'écoulements granulaires, mais aussi pour essayer de compléter les résultats expérimentaux. Le traitement rigoureux du contact nous apparaît essentiel pour étudier un tel phénomène. Nos travaux s'inscrivent dans le prolongement des expériences réalisées par Bonamy [18, 17] et Rajchenback [127]. Après un panorama des écoulements granulaires, précisant le cadre théorique dans lequel nous nous plaçons, nous présenterons les résultats numériques obtenus avec des simulations bi-dimensionnelles et tri-dimensionnelles.

6.1 Panorama

Les matériaux granulaires présentent de nombreux et inhabituels comportements : ils peuvent s'écouler comme un liquide mais, sous certaines circonstances, ils peuvent se bloquer et résister aux forces extérieures sans se déformer. La plus spectaculaire manifestation de cette dualité liquide/solide se produit durant une avalanche lorsqu'une fine couche de grains roule à la surface d'un lit où la plupart des grains semblent statiques. L'évolution globale de la surface d'un écoulement peut être approchée par des modèles dérivant de la physique non linéaire [11, 12, 21] ou de la mécanique des fluides [43, 18]. Cependant, certains résultats expérimentaux restent encore inexplicables : par exemple, les profils de vitesses expérimentaux mesurés dans les écoulements quasi 2D [127, 18] ou tridimensionnel [18, 54, 68, 156] montrent clairement la sélection d'un gradient de vitesse constant à l'intérieur de l'épaisseur roulante tandis que l'équilibre des moments implique que la contrainte de cisaillement augmente avec la profondeur. Ces observations ne sont compatibles avec aucune loi constitutive reliant contrainte et déformation. De nombreux modèles

ont été récemment proposés pour décrire la rhéologie des écoulements granulaires denses en prenant en compte les effets non-locaux [10, 127], en adaptant la théorie cinétique [141], en modélisant les écoulements denses comme des écoulements partiellement fluidisés [12]. Certains d'entre eux ont été utilisés pour décrire les écoulements de surface [10, 12, 127], mais aucun d'entre eux ne réussit à capturer un gradient de vitesse constant à l'intérieur de l'avalanche. Une compréhension claire des avalanches fait donc défaut.

6.2 Cadre théorique

Nous présentons le cadre théorique grâce auquel nous voulons décrire l'écoulement de notre milieu. On peut trouver dans [43] une présentation des équations classiques décrivant l'évolution de la surface libre, valide aussi bien pour un fluide que pour du sable. L'épaisseur de la phase roulante n'étant a priori pas connue, nous sommes en présence d'un système d'équations ouvert. On considère une couche en mouvement, d'épaisseur faible, s'écoulant sur un lit statique (figure 6.1). On peut extrapoler cet écoulement à un écoulement confiné dans un tambour tournant. On définit alors le repère orthormé $(O; \mathbf{e}_x, \mathbf{e}_z)$ où O est le centre du tambour, \mathbf{e}_x étant un vecteur parallèle de même sens que l'écoulement. On note ρ la densité du milieu, g la gravité, σ le tenseur des contraintes local et $\dot{\mathbf{q}}$ la vitesse décomposée dans le repère lié au tambour comme $\dot{\mathbf{q}} = \dot{q}_x \mathbf{e}_x + \dot{q}_z \mathbf{e}_z$. Ω représente la vitesse de rotation du tambour tandis que $\Sigma(x)$ correspond à la surface d'abscisse x perpendiculaire à l'écoulement. On note par la suite

$$\langle A \rangle = \frac{1}{R} \int_0^R A dz, \quad (6.1)$$

défini comme la moyenne de la quantité A sur l'intervalle $[0, R]$. Ainsi en supposant que notre milieu est incompressible (ce qui revient à supposer une compacité constante dans l'écoulement [43]), les équations de conservation de la masse et de la quantité de mouvement, projetées sur \mathbf{e}_x et moyennées sur l'épaisseur de la couche en mouvement s'écrivent

$$\begin{cases} \frac{\partial}{\partial t}(R \langle \rho \rangle) + \frac{\partial}{\partial x}(R \langle \rho \dot{q}_x \rangle) + (\rho \dot{q}_z)|_{z=-R} = 0 \\ \frac{\partial}{\partial t}(R \langle \rho \dot{q}_x \rangle) + \frac{\partial}{\partial x}(R \langle \rho \dot{q}_x^2 \rangle) + (\rho \dot{q}_x \dot{q}_z)|_{z=-R} = F \end{cases}, \quad (6.2)$$

où F représente la composante suivant \mathbf{e}_x de la force appliquée sur le volume d'intégration. Les termes $R \langle \rho \dot{q}_x \rangle$ et $R \langle \rho \dot{q}_x^2 \rangle$ du système d'équations (6.2) représentent respectivement le débit et le flux de quantité de mouvement projeté sur \mathbf{e}_x traversant la surface $\Sigma(x)$ tandis que les termes $(\rho \dot{q}_z)|_{z=-R}$ et $(\rho \dot{q}_x \dot{q}_z)$ représentent eux le débit et le flux de quantité de mouvement projeté sur \mathbf{e}_x traversant l'interface fluide/statique.

En supposant ρ constant, le système (6.2) se réécrit

$$\begin{cases} \frac{\partial}{\partial t}(R) + \frac{\partial}{\partial x}(R \langle \dot{q}_x \rangle) + (\dot{q}_z)|_{z=-R} = 0 \\ \frac{\partial}{\partial t}(R \langle \dot{q}_x \rangle) + \frac{\partial}{\partial x}(R \langle \dot{q}_x^2 \rangle) + (\rho \dot{q}_x \dot{q}_z)|_{z=-R} = \frac{F}{\rho} \end{cases}, \quad (6.3)$$

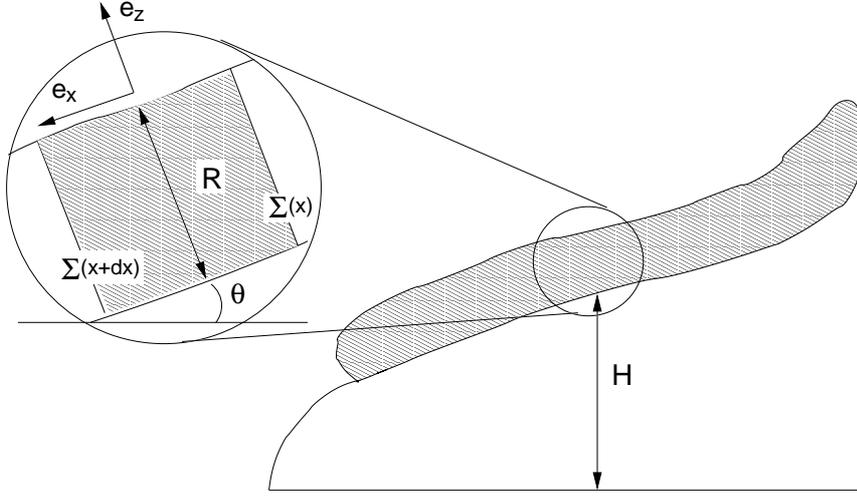


FIG. 6.1 – Ecoulement d’une couche de faible épaisseur sur un lit statique. On note θ l’angle local entre l’interface et l’horizontale. Le repère $(O; \mathbf{e}_x, \mathbf{e}_z)$ est pris tangent à l’écoulement. On note R l’épaisseur de la phase roulante. $\Sigma(x)$ et $\Sigma(x+dx)$ représentent les deux surfaces délimitant notre volume d’étude.

Du système (6.3) seules les expressions $(R, \theta) \rightarrow \langle \dot{q}_x \rangle (R, \theta)$, $(R, \theta) \rightarrow \langle \dot{q}_x^2 \rangle (R, \theta)$ et $(R, \theta) \rightarrow F(R, \theta)$ restent inconnues et sont nécessaires pour fermer le système. En effet, les expressions $(\dot{q}_z)|_{z=-R}$ et $(\rho \dot{q}_x \dot{q}_z)|_{z=-R}$ qui dépendent explicitement du mouvement de la phase statique par rapport au référentiel s’écrivent dans le cas des tambours tournants

$$\begin{cases} (\dot{q}_z)|_{z=-R} = \Omega r \\ (\rho \dot{q}_x \dot{q}_z)|_{z=-R} = 0 \end{cases} \quad (6.4)$$

La fermeture de ces équations a été obtenue via de nombreux modèles [17, 101, 127]. Nous proposons dans la partie qui suit de compléter les informations jusqu’ici obtenues, grâce aux simulations numériques 2D dans un premier temps, puis 3D.

6.3 Analyse 2D

6.3.1 Méthodologie

Les systèmes simulés sont similaires à ceux étudiés expérimentalement dans [18, 127]. Nous avons modélisé un tambour tournant bi-dimensionnel de diamètre D_0 égal à 450 mm à moitié rempli avec 7 183 disques rigides de masse volumique $\rho_0 = 2.7 \text{ g.cm}^{-2}$ et de diamètre uniformément distribué entre 3 mm et 3.6 mm. Cette faible polydispersité prévient des effets cristallins bi-dimensionnels pouvant entraîner des effets non génériques. Une plus grande polydispersité peut conduire à des phénomènes de ségrégation à l’intérieur du tambour [54] pouvant créer eux aussi des effets indésirables : la taille des particules n’est

alors plus distribuée uniformément dans la tranche centrale du tambour. Le coefficient de restitution normal entre disques (resp. entre disques et tambour) est égal à 0.46 (resp. 0.46) et le coefficient de frottement à 0.4 (resp. 0.95). Le coefficient de frottement avec le tambour est pris volontairement plus important pour contraindre les disques à suivre la rotation du tambour et ne pas glisser sur la frontière. Le pas de temps est égal à $\delta t = 6.10^{-3}$ sec.. La procédure de simulation est la suivante : tous les disques sont placés dans un tambour fixe. Après stabilisation de l'échantillon une vitesse de rotation Ω est donnée au tambour (allant de 2 rpm à 15 rpm). Après une phase de transition, nous atteignons un régime continu d'écoulement. Nous commençons alors à prendre des clichés de l'échantillon à une fréquence de 100Hz. Le nombre de pas de temps nécessaires pour achever cette phase est compris entre 4.10^3 et 1.10^4 dépendant de la vitesse de rotation. Au final nous obtenons 400 clichés pour chaque simulation. Sur le ordinateur *SGI Origin 3800* avec 16 processeurs, environ 20 h de calcul sont nécessaires pour achever cette phase de l'expérience (un calcul séquentiel coûterait entre 10 et 12 fois plus cher). Toutes les simulations ont été réalisées avec la version parallèle (solveur NLGS et implémentation ELG) détaillée dans le chapitre 3 implémentée dans le logiciel *LMGC90* [45]. Pour chaque bille et pour chaque cliché nous enregistrons la position \mathbf{q} du centre de masse, la vitesse "instantanée" $\dot{\mathbf{q}}$ du centre de masse mesurée à travers une fenêtre temporelle de largeur $\delta_t = 6.10^{-3}$ sec. et le tenseur de contrainte local σ . Dans ce qui suit les distances, temps et contraintes sont adimensionnées par d , $\sqrt{d/g}$ et $\rho_0 g d$. Les comparaisons entre expériences sur plan incliné et en tambour tournant suggèrent que le principale paramètre de contrôle adimensionné est le débit moyen $Q = (\Omega D_0^2)/(8d\sqrt{gd})$ [101]. Le débit moyen est égal à $Q = 8.93, 17.86, 22.32, 26.79, 44.64, 66.96$ pour $\Omega = 2, 2, 4, 5, 10, 15$ rpm (*round per minute* - tour par minute). Lorsque le tambour n'est pas rempli parfaitement à 50%, cette valeur doit être réajustée : $Q^* = (\Omega(R_0 + H_0)^2)/(2d\sqrt{gd})$, H_0 étant la plus courte distance séparant la surface libre du centre du tambour.

6.3.2 Frottement dynamique

La figure 6.2 montre le cliché d'une simulation typique sur lequel est représenté le champ de vitesse obtenu pour une vitesse de rotation $\Omega = 6$ rpm. Deux phases peuvent être distinguées : une "phase statique" qui tourne avec le tambour et une couche en mouvement qui s'écoule sur le lit statique. L'angle moyen θ de l'écoulement est défini comme l'orientation de la vitesse moyennée sur l'ensemble des particules pour lesquelles

- la vitesse est supérieure à la vitesse de rotation du tambour $\Omega D_0/2$ et ce sur toute la simulation
- la compacité est supérieure à un seuil fixé (égal généralement à 0.5)

Cet angle coïncide avec le profil local de la surface libre au centre du tambour et peut également être obtenu par une régression linéaire effectuée sur le profil des positions des particules à la surface libre. Le repère (\vec{e}_x, \vec{e}_z) est alors choisi de façon à ce que le vecteur \vec{e}_x (resp. \vec{e}_z) soit parallèle (resp. perpendiculaire) à la surface libre. L'origine est prise au centre du tambour. Pour représenter les différents profils, les grandeurs cinématiques et sthéniques attachées à chaque particule sont moyennées à travers une fenêtre centrée

au centre du tambour d'épaisseur 40 diamètres de grains et de profondeur le rayon du tambour. L'axe de cette zone est perpendiculaire à l'écoulement.

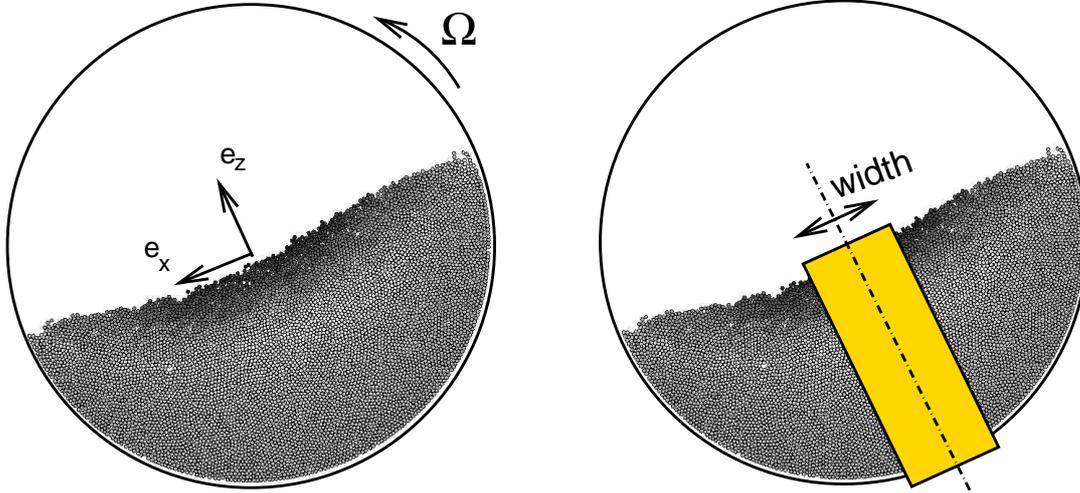


FIG. 6.2 – Gauche : cliché d'une simulation numérique en tambour tournant (vitesse moyenne en échelle de gris). Droite : visualisation de la zone d'étude "spatio-temporelle".

Au centre du tambour, le flot peut être considéré comme homogène dans la direction de l'écoulement [18, 101]. Sous cette hypothèse, la distribution des contraintes dans la partie centrale du tambour ($x = 0$) est donné par $\sigma_{xz} = -\nu \sin \theta z$ et $\sigma_{zz} = -\nu \cos \theta z$. L'angle moyen de l'écoulement θ peut être assimilé à un coefficient de frottement effectif μ_{eff} défini comme le rapport entre la composante tangentielle et normale de la contrainte et égale à $\mu_{eff} = \tan \theta$. Son évolution en fonction de Q , le débit, est représenté sur la figure 6.3. La pente augmente légèrement avec Q :

$$\mu_{eff} = \mu_{eff}^0 + \mu_Q Q \quad (6.5)$$

où $\mu_{eff}^0 = 0.31$ et $\mu_Q = 1.7 \cdot 10^{-3}$. Cette faible dépendance peut être attribuée aux effets de parois comme conjecturé expérimentalement [18, 101].

6.3.3 Profil de vitesse

La figure 6.4 montre un profil typique de vitesse. Il est composé de deux parties : dans la phase roulante, la vitesse v_x évolue linéairement avec la profondeur z comme $v_x = \dot{\gamma}(z+H)$ où $\dot{\gamma}$ et H sont respectivement le gradient de vitesse dans la couche roulante (constant) et l'épaisseur de la phase roulante. Dans la phase statique, le profil de vitesse coïncide avec le mouvement de solide rigide du tambour : $v_x = \Omega z$. L'épaisseur de la phase roulante H est alors définie par l'intersection de ces deux comportements (voir figure 6.4). La transition

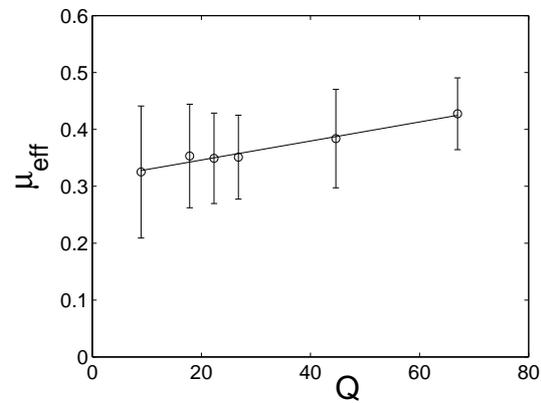


FIG. 6.3 – Variation du coefficient de frottement effectif $\mu_{err} = \tan \theta$ de la surface de l'écoulement en fonction du coefficient d'écoulement Q (non dimensionné). Les barres d'erreur montrent l'écart à la séquence pour Q constant. La droite est une régression linéaire.

entre ces deux comportements peut alors être décrite par une exponentielle de longueur de décroissance caractéristique λ (voir encart de la figure 6.4).

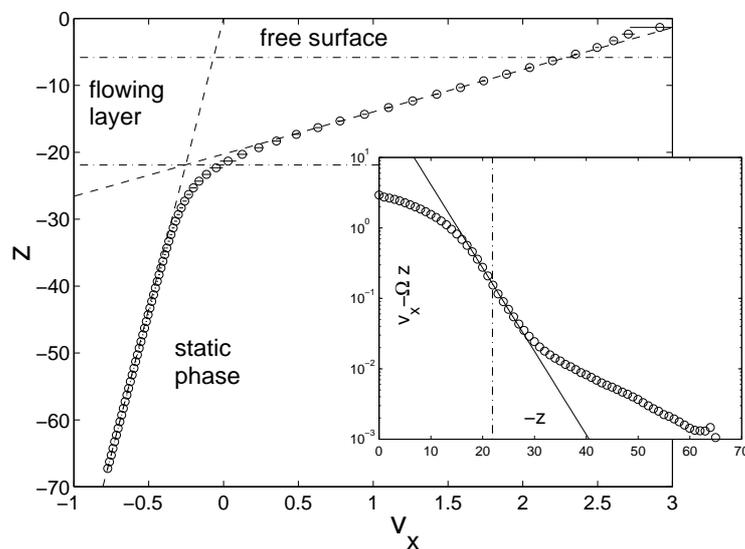


FIG. 6.4 – Profil de vitesse $v_x(z)$ (moyenné sur 400 instants) dans la tranche centrale du tambour pour une vitesse de rotation $\Omega = 6$ rpm. La barre d'erreur correspond à un intervalle de confiance de 99%.

La droite pointillée correspond à une régression linéaire : $v_x = \dot{\gamma}(z + H)$ où $\dot{\gamma} \simeq 0.15$. La droite pleine correspond à la droite Ωz . L'interface liquide/statique est défini à la profondeur où ces deux droites se rencontrent (ligne mixte). L'épaisseur de la phase roulante H peu alors être déduite : $H \simeq 16$. En encart, nous retranchons à v_x la composante dû au mouvement de solide rigide Ωz et représentons cette quantité en fonction de la profondeur $-z$ en échelle semi-logarithmique en ordonnée. La droite pleine correspond à une régression exponentiel de longueur caractéristique : $\lambda \simeq 3.7$.

L'allure des différents profils de vitesse tracés est très similaire à ceux observés expérimentalement [18]. Les profils sont caractérisés par un ensemble de trois paramètres, l'épaisseur roulante H , le gradient de vitesse $\dot{\gamma}$ dans la phase roulante, et la longueur caractéristique λ de l'exponentielle joignant les deux phases. L'évolution de l'épaisseur roulante H en

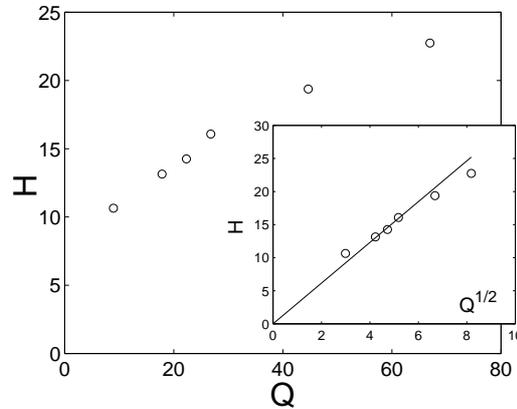


FIG. 6.5 – Epaisseur roulante H dans la tranche centrale du tambour vue comme une fonction du débit Q . En encart, H vs \sqrt{Q} . La droite est une régression linéaire $H = 3\sqrt{Q}$.

fonction de Q est reportée sur la figure 6.5. L'épaisseur roulante est approchée par \sqrt{Q} comme observée expérimentalement [18, 101]. La figure 6.6 montre la variation du gradient de vitesse $\dot{\gamma}$ dans l'épaisseur roulante en fonction de Q . Dans nos simulations, $\dot{\gamma}$ montre une dépendance vis à vis de Q et varie typiquement de 0.1 à 0.25 quand Q varie de 10 à 80. Cette dépendance est compatible avec celles observées expérimentalement en tambour tournant 2D par Rajchenbach [127] : cet auteur propose que $\dot{\gamma}$ soit approché par $\dot{\gamma} \propto (\sin \theta - \sin \Phi)^{1/2} / \cos^{1/2} \Phi$, ce qui est compatible avec nos résultats (encart de la figure 6.6). Ceci suggère des effets non-triviaux des parois latérales. Cette similitude entre simulations et expériences quasi-2D n'est pas observée avec les simulations MD ou DEM. Ceci souligne l'importance de résoudre le contact de façon exacte et non pas par des lois régularisantes. La plupart des simulations MD ne simulent pas des écoulements denses mais bel et bien des écoulements lâches où les contacts sont exclusivement binaires. La variation de la longueur caractéristique λ de l'exponentielle reliant les deux phases en fonction de Q est représentée sur la figure 6.7. λ est indépendant de Q , et est de l'ordre de 3.5. Ce comportement est similaire à ceux observés dans les expériences sur plan incliné

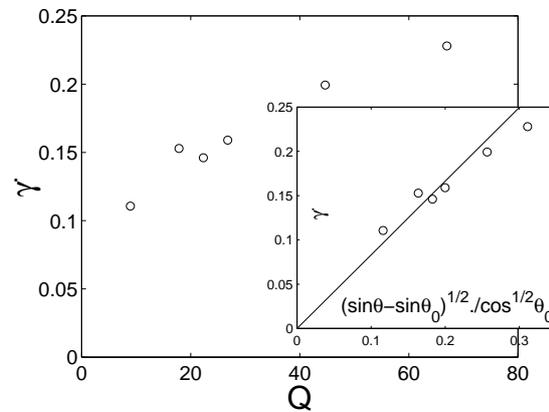


FIG. 6.6 – Gradient de vitesse $\dot{\gamma}$ dans la phase roulante en fonction du débit Q . En encart : $\dot{\gamma}$ vs. $(\sin\theta - \sin\Phi)^{1/2} / \cos^{1/2}\Phi$ où l'angle de frottement de Coulomb $\Phi = 17.4^\circ$ a été identifié avec la valeur μ_{eff}^0 définie dans la figure 6.3. La droite est une régression linéaire $\dot{\gamma} = 0.8(\sin\theta - \sin\Phi)^{1/2} / \cos^{1/2}\Phi$.

[84] et en tambour tournant [18]. Notons que dans ces expériences 3D, λ est plus petit et de l'ordre de 2.

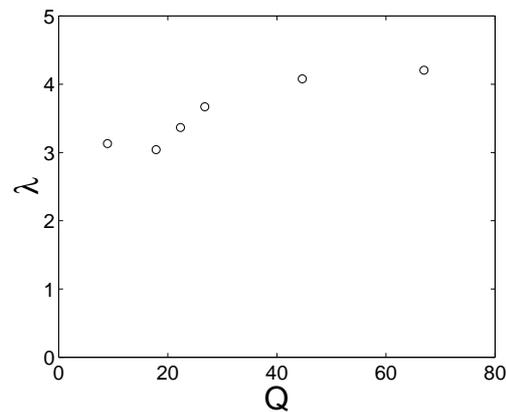


FIG. 6.7 – Longueur de décroissance caractéristique reliant le profil linéaire de la phase liquide et la rotation solide de la phase statique.

6.3.4 Vitesse angulaire

Un profil typique de vitesse angulaire $\omega(z)$ est représenté sur la figure 6.8. Tracer ω en fonction du gradient de vitesse local dv_x/dz est très intéressant (voir encart de la figure 6.8). Il existe une relation évidente entre ces deux quantités : $\omega = \frac{1}{2} \frac{dv_x}{dz}$ dans *tout* l'échantillon indépendamment de Q . Cette relation est analogue à celle obtenue en hydrodynamique classique où la vitesse de rotation moyenne des particules est égale à la vorticité de l'écoulement. Une telle relation ont été observé lors de simulations en Dynamique Moléculaire d'écoulement granulaire dilué [97], mais impossible à observer pour de forte compacité. Dans ce dernier cas, les grains seraient supposés s'organiser en couche tournant dans la même direction. Ceci ferait décroître la vitesse angulaire moyenne des grains et ω serait alors plus petit que $1/2 dv_x/dz$. Dans nos simulations numériques, un tel comportement n'est pas observé, ce qui suggère que la rotation des grains ne s'organise pas en couche en dépit de l'importante densité de l'écoulement.

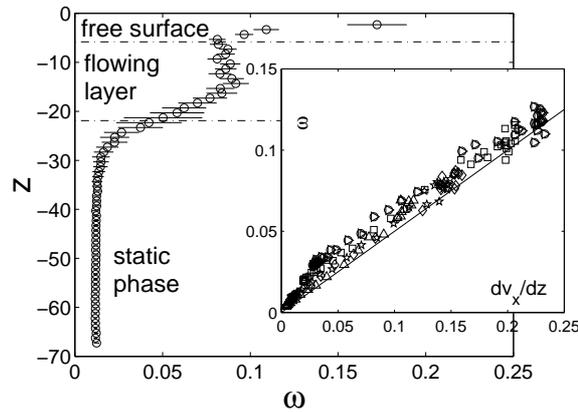


FIG. 6.8 – Profil de vitesse angulaire moyenne $\omega(z)$ (moyennée sur 400 instants) dans la tranche centrale du tambour obtenu pour $\Omega = 6$ rpm. Les deux lignes horizontales mixtes délimitent la surface libre et l'interface solide/liquide. En encart, ω vs le gradient de vitesse dv_x/dz pour différentes vitesses de rotations Ω . La droite est une régression linéaire $\omega = \frac{1}{2} \frac{dv_x}{dz}$.

6.3.5 Compacité

On peut sur la figure 6.9 observer un profil typique de compacité mesuré dans la tranche centrale du tambour. Notons que ce profil ainsi que les profils de coordonnée et de contrainte sont impossibles à mesurer expérimentalement pendant l'écoulement. On peut tout d'abord remarquer que c est constant, autour de 0.8 à l'intérieur du tambour, excepté dans deux faibles zones situées à la surface libre et à la paroi du tambour. A la surface libre, c atteint rapidement une valeur fixe dans une faible zone d'épaisseur égale à

trois/quatre diamètres de billes indépendamment de Q . Dans la suite la surface frontière est placée à la frontière inférieure de cette petite région (ligne pointillée sur la figure 6.9) définie au point où c devient plus grand que 0.7. Près de la paroi du tambour, c chute pour atteindre une faible valeur à l'intérieur d'une faible région d'épaisseur égale à deux/trois diamètres de billes attribuée à frontière courbe du tambour. Un zoom sur le profil de

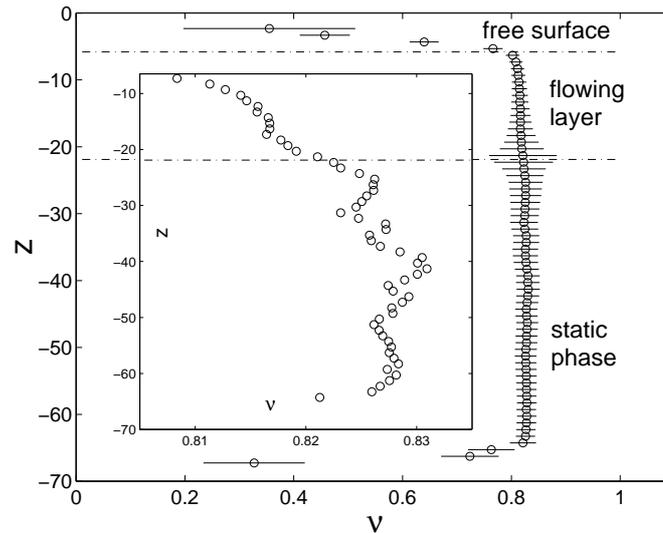


FIG. 6.9 – Profil de compacité (z) (moyenné sur 400 intervalles de temps) dans la tranche centrale du tambour pour une vitesse de rotation de $\Omega = 6$ rpm. encart, zoom dans la région "constante" souligne les faibles variations de c dans les deux phases.

compacité (encart de la figure 6.9) montre que c est faiblement perturbé dans les deux phases et reste pratiquement constant. On peut dénoter un saut à l'interface entre les phases liquide et solide.

6.3.6 Coordinence

L'évolution de la coordinence est plus représentatif de la transition liquide/solide que la compacité. L'interface entre les deux phases peut clairement être distinguée : dans la phase statique le nombre de coordinence possède une évolution constante contrairement à la phase liquide où l'évolution de la coordinence croît avec la profondeur de façon linéaire. La vitesse de rotation ne semble pas perturber ce comportement : seule la valeur de la coordinence dans la phase statique est modifiée, valeur qui décroît avec l'augmentation de la vitesse de rotation. A la surface libre Z_c décroît légèrement jusqu'à 2.4 sur une faible région. La première ligne en pointillée (en haut) représente la surface libre, tracée au point où Z_c croise la valeur 2.4. Près de la paroi du tambour Z_c chute vers une faible valeur sur une faible épaisseur. La seconde ligne en pointillée (en bas) situe l'interface solide/liquide.

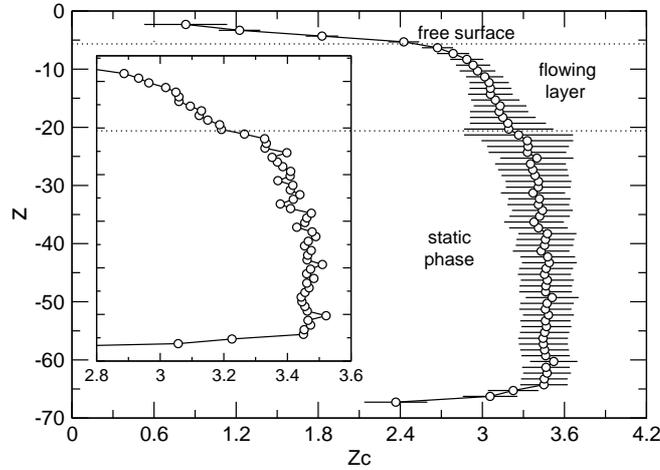


FIG. 6.10 – Profil de la coordonnée $Z_c(z)$ (moyenné sur 400 intervalles de temps) dans la tranche centrale du tambour pour une vitesse de rotation de $\Omega = 6$ rpm.

Z_c apparaît comme constant dans la phase statique autour de 3.4. En encart, zoom dans la région constante soulignant les faibles variations de Z_c dans les deux phases.

6.3.7 Contraintes

Les différents profils obtenus jusqu'ici peuvent être comparés aux profils expérimentaux. Nous proposons d'étudier les profils des différentes composantes de σ , impossible à obtenir expérimentalement. La figure 6.11 trace l'évolution de deux des composantes du tenseur de contrainte statique σ : σ_{xx} et σ_{zz} . L'évolution de $\sigma_{zz}(z)$ confirme nos attentes : une décroissance linéaire caractéristique avec la profondeur ($\sigma_{zz} \approx -\rho g z \cos(\theta)$). La composante $\sigma_{xx}(z)$ ne suit pas globalement la même loi. Bien que dans la phase roulante une régression linéaire avec la loi précédente puisse être réalisée, identique à celle obtenue avec la composante σ_{zz} , à l'entrée de la phase statique cette approximation ne concorde plus. La décroissance du profil est beaucoup plus erratique, apparemment linéaire avec un gradient différent. Ce comportement erratique est due aux voûtes se créant à l'intérieur de la phase statique où le réseau de contact est dense (figure 6.12).

De longues chaînes de contacts se créent, parallèlement à l'écoulement (portées par \mathbf{e}_x). Cette répartition est in-homogène dans la profondeur engendrant ainsi les fluctuations observées. En surface lorsque ces chaînes se forment, leur intensité est moindre ne perturbant pas la statistique.

6.3.8 Discussion

Nos échantillons diffèrent des expériences sur plusieurs points. Le premier est que nos simulations sont bi-dimensionnelles se rapprochant des expériences "quasi-2D" de Rajchen-

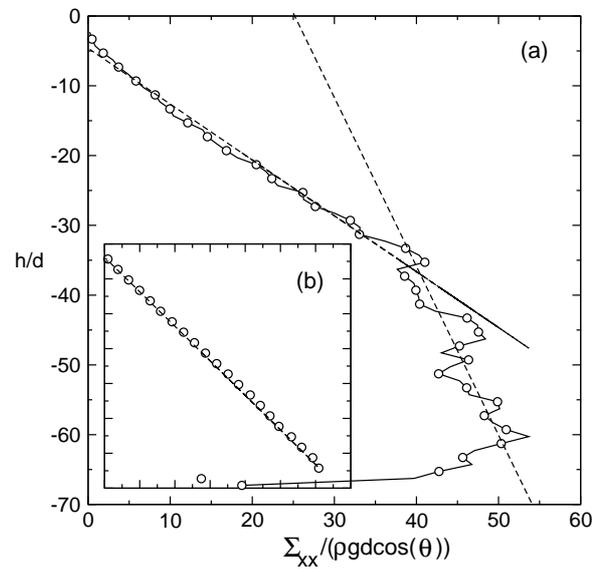


FIG. 6.11 – Profils de deux des trois composantes du tenseur de contrainte statique σ , *i.e.* σ_{xx} (a) et σ_{zz} (b). Dans l’encart (b), nous pouvons apprécier l’approximation de σ_{zz} par $\rho g d \cos(\theta)$, où θ est l’angle à la surface libre.

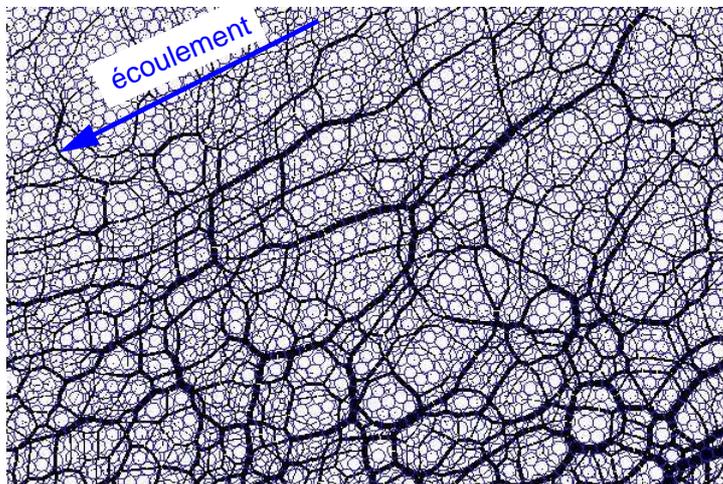


FIG. 6.12 – Visualisation des chaînes de forces à l’intérieur de la phase statique.

bach [127]. Les expériences en tambour tournant effectuées dans [18, 156] sont purement 3D et prennent en compte des effets impossibles à quantifier dans nos simulations (frotte-

ment avec les parois latérales). Le second point à prendre en compte est le fait que dans nos simulations le tambour n'est pas parfaitement rempli à moitié. Ce décalage de la surface libre au repos doit être pris en compte. Le dernier point concerne la mesure des profils obtenus. Dans nos simulations les profils sont obtenus en moyennant différentes quantités liées aux billes appartenant à la tranche centrale étudiée. Dans les simulations tridimensionnelles, les profils sont obtenus en prenant toujours en compte les billes appartenant à la tranche centrale mais également proche de la paroi du tambour. Ces deux types de mesure peuvent donc avoir une influence sur les profils 2D et 3D.

Ces différences ne jouent pas sur l'évolution de différentes quantités. Le profil de vitesse a un comportement linéaire dans la phase statique et dans l'épaisseur roulante avec un pied exponentiel reliant ces deux comportements conformément aux expériences effectuées en tambour tournant [101]. Cependant ce profil semble dépendre de la vitesse de rotation, dépendance non observée expérimentalement avec des tambours 3D, mais envisagée pour les simulations quasi-2D [127]. La correspondance des profils des quantités ω et $\frac{\partial u_x}{\partial z}$ quelque soit le débit Q est aussi un point intéressant retrouvant un résultat analogue à ceux observés en hydrodynamique.

L'évolution des profils de contraintes fournit aussi des résultats cohérents : un comportement isotrope du tenseur des contraintes dans la phase roulante, perturbé dans la phase statique par les chaînes de forces parallèles à l'écoulement.

6.4 Analyse 3D

6.4.1 Méthodologie

Pour nos premières simulations tridimensionnelles, nous avons créé des échantillons différents de ceux étudiés dans la section précédente. Les systèmes simulés sont similaires à ceux étudiés expérimentalement dans [127], le diamètre du tambour étant plus petit, nous permettant pour une première approche de ne pas avoir un nombre trop important de particules. Le tambour tournant possède un diamètre D_0 égal à 250 mm rempli avec 11 147 sphères rigides de masse volumique $\rho_0 = 2.7 \text{ g.cm}^{-2}$ et de diamètres uniformément distribués entre 3 mm et 3.6 mm. Les paramètres internes restent inchangés : le coefficient de restitution normal entre sphères (resp. entre sphères et tambour) est égal à 0.46 (resp. 0.46) et le coefficient de frottement à 0.4 (resp. 0.95). Nous devons définir également les valeurs de ces deux coefficients pour le contact entre les sphères et les parois latérales : la restitution normale est égale à 0.46 et le coefficient de frottement est égal dans un premier temps à 0.1. L'influence de ce dernier peut être prépondérante sur les profils : nous serons donc amené à jouer avec lui. Le pas de temps est égal à $\delta t = 1.10^{-3} \text{ sec.}$. La procédure de simulation est identique à la section précédente. Comme pour les simulations 2D, nos calculs ont été réalisés avec la version parallèle (solveur NLGS et implémentation ELG) détaillée dans le chapitre 3 et implémentée dans la version 3D du logiciel *LMGC90*.

Les mesures des profils ont été effectuées suivant trois procédures. La première comptabilise les particules appartenant à la tranche centrale sur toute l'épaisseur du tambour, la seconde ne le fait que dans la zone centrale de l'épaisseur et la troisième sur une tranche près de

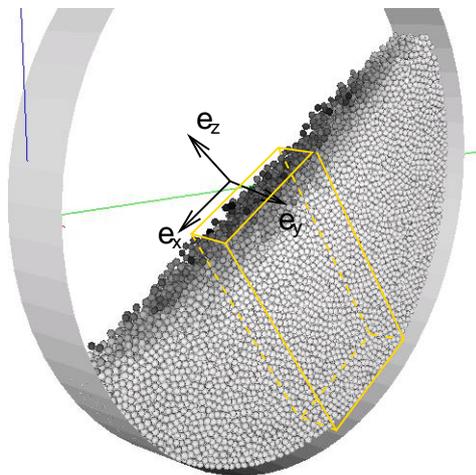


FIG. 6.13 – Echantillons 3D : les particules sont coloriées en fonction de leur vitesse moyenne (valeur maximale en noir). La zone d'étude est également représentée ainsi que le repère lié à l'écoulement.

la paroi. Au vu de la faible épaisseur du tambour (4 diamètres de bille), les résultats sont identiques. Les profils que nous présentons par la suite sont obtenus suivant la première méthode.

6.4.2 Profils de vitesses

La figure 6.14 montre le profil de vitesse pour une vitesse de rotation $\Omega = 6$ rpm, ce qui correspond à un débit Q égal à 8.27. On retrouve un profil composé en deux parties : dans la phase roulante, la vitesse v_x évolue linéairement avec la profondeur z comme $v_x = \dot{\gamma}(z+H)$ où $\dot{\gamma}$ et H sont respectivement le gradient de vitesse dans la couche roulante (constant) et l'épaisseur de la phase roulante : ici $H = 6.79$ et $\dot{\gamma} = 0.288$ pour un angle à surface libre de 29.22. Dans la phase statique, le profil de vitesse coïncide également avec le mouvement de solide rigide du tambour : $v_x = \Omega z$. L'exponentielle joignant les deux comportements à une longueur de décroissance caractéristique λ égale à 3.7.

6.4.3 Coördinance

Parmi les premiers profils tracés, nous avons également regardé l'évolution de la coördinance moyenne dans la tranche centrale, évolution illustrée par la figure 6.15. On distingue, comme pour le cas 2D, les trois phases constituant l'écoulement : la surface libre, l'épaisseur roulante et la phase statique. A la surface libre Z_c croît rapidement (sur une épaisseur de 2 à 3 diamètres de billes), passant d'un régime de vol libre à un régime plus fluide où la coördinance moyenne est de 4.3 à l'interface. A l'intérieur de l'épaisseur

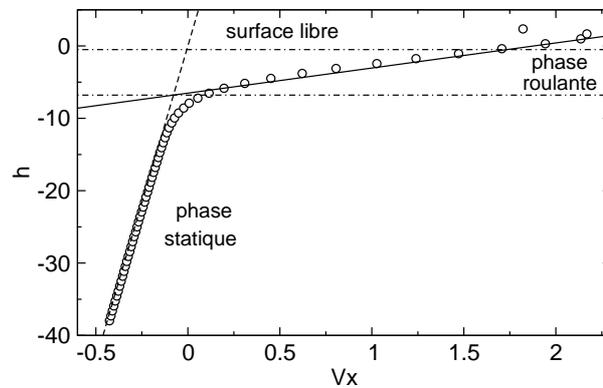


FIG. 6.14 – Profil de vitesse $v_x(z)$ (moyenné sur 300 instants) dans la tranche centrale du tambour pour une vitesse de rotation $\Omega = 6$ rpm

roulante, Z_c continue de croître pour atteindre 5.7 à l'entrée de la phase statique. Dans cette dernière partie, la coordinnce reste constante autour d'une valeur moyenne égale à 6.

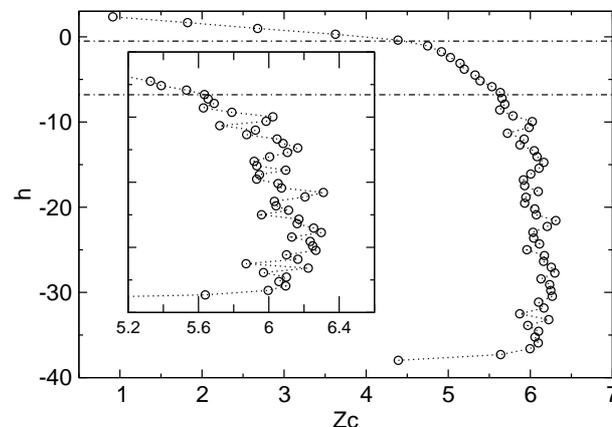


FIG. 6.15 – Profil de coordinnce $Z_c(h)$ (moyenné sur 300 instants) dans la tranche centrale du tambour pour une vitesse de rotation $\Omega = 6$ rpm

La comparaison avec les simulations bidimensionnelles est très intéressante : un comportement identique dans les trois phases caractéristiques avec dans la phase statique une valeur de la coordinnce égale dans les deux cas au nombre de coordinnce moyen des échantillons denses. Comme pour les simulations 2D on ne peut pas comparer cette évolution avec des résultats expérimentaux : les profils de coordinnce et de compacité étant impossible à

mesurer expérimentalement lors d'écoulements denses.

6.5 Frottement latéral et vitesse de rotation

Dans les simulations 2D en tambour tournant, le gradient $\dot{\gamma}$ dépend de la vitesse de rotation du tambour comme l'a montré Rajchenback dans ses expériences quasi 2D [127]. Nous voulons vérifier si, dans nos expériences 3D, cette dépendance existe toujours; rappelons quelle n'est pas établie d'un point de vue expérimental [18]. A l'heure actuelle, à cause des longs temps de calcul, une étude paramétrique fine n'a pu être menée. La mise en régime continu, seul état pertinent pour analyser de façon identique nos différents profils, s'avère être très longue. En témoigne la figure 6.16 illustrant l'évolution de l'angle moyen d'écoulement pour différentes valeurs du coefficient de frottement sphère/paroi et vitesse de rotation.

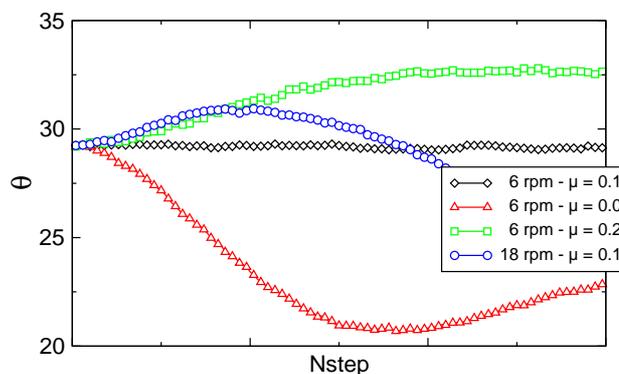


FIG. 6.16 – Evolution de l'angle moyen d'écoulement pour différentes valeurs du coefficient de frottement sphère/paroi et vitesse de rotation

L'évolution régulière correspond à l'angle d'écoulement continu des simulations précédentes. Si l'on change les paramètres que sont la vitesse de rotation et le frottement grain/paroi, le régime d'écoulement continu ne se met pas en place rapidement, spécialement lorsque le frottement diminue ou lorsque la vitesse de rotation augmente. Les données actuelles ne nous permettent aucune conclusion.

Chapitre 7

Mouvements tectoniques

7.1 Etude du terrain

La tectonique [100] se présente comme l'étude des structures naturelles en relation avec les mouvements (cinématiques) et les forces (dynamiques) qui les ont créées : les plus beaux résultats de ces manifestations peuvent s'observer directement sur les structures naturelles (photo 7.1). Une partie des déformations terrestres résulte du mouvement de grandes plaques continentales (figure 7.2). Lorsque ces plaques s'écartent (divergent), les bordures vont travailler en traction et sont alors soumises à des efforts en allongement ou extension. Lorsque les plaques convergent, les bordures travaillent en compression et la déformation subie est un raccourcissement (on parle de zone de *collision* ou de *subduction*). Les plaques peuvent encore glisser les unes par rapport aux autres sans observer les phénomènes précédents : on parle alors de coulissement.

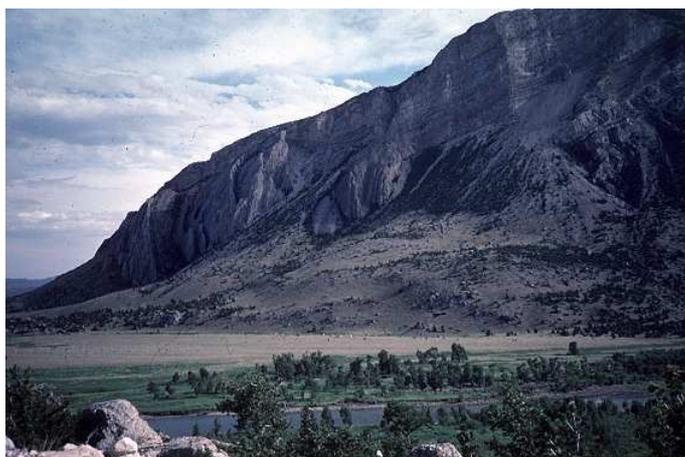


FIG. 7.1 – Photo des "Bear Tooth Mountains" dans le Wyoming (cliché de R.W. Allmendinger - 2000). Le profil gauche laisse supposer un développement d'une faille inverse.

Ces phénomènes sont en majeure partie la cause des transformations aux quelles est confrontée notre planète. En comprendre l'évolution et les mécanismes peut nous permettre de comprendre plus en détail l'histoire de la terre. Lors de ces mouvements ou combinaisons de ceux-ci, on trouve parmi les principales déformations hétérogènes continues les failles et les plis, que l'on retrouve au coeur des systèmes plissés et des zones de cisaillement ductiles [149].

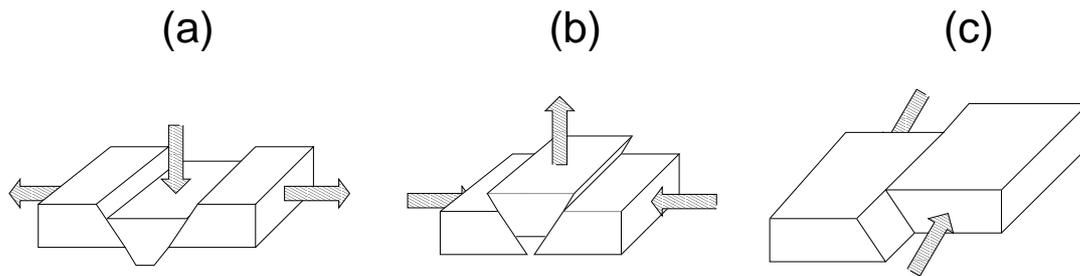


FIG. 7.2 – Différents types de déformations liées aux mouvements des plaques continentales engendrant des failles normales lors d'allongement (a), inverses lors de raccourcissement (b) ou de décrochement lors de coulissement (c).

Une faille est une déformation cassante pouvant être vue comme une fracture accompagnée du déplacement relatif des blocs qu'elle sépare parallèlement à son plan appelé *surface de faille* 7.2. Lorsque le mouvement de plaque se fait horizontalement (pas de création de dénivelé entre les blocs) on parle de faille de décrochement. Elle est caractérisée de *normale* lorsque la surface de faille a sa pente en direction du bloc abaissé ; dans le cas inverse on parlera de faille *inverse*. Généralement ce troisième type de faille résulte d'une compression dont l'angle entre les deux blocs peut varier de façon importante : le cas limite où l'angle est plat conduit au phénomène de chevauchement. Ce cas permet d'observer le deuxième type de déformation : les plis.

Les plis relèvent eux de déformations plastiques, déformations beaucoup plus douces que les failles. Les plis sont caractérisés de droits et symétriques lorsque le plan axial est vertical (plan de symétrie des deux flans), déjetés lorsque l'un des flans est vertical (la plan axial ne l'est alors plus), renversés ou couchés (figure 7.3) lorsque l'angle d'un des flans est inférieur à 90° , possédant dans chaque cas une partie anticlinale (concave) et une partie synclinale (convexe). Lors de superposition de couches, on parle de plis concentriques lorsque l'épaisseur de chaque couche reste inchangée et de plis semblables lorsque l'épaisseur varie. On trouve également différentes combinaisons de ces deux phénomènes : des plis-failles, failles sur plan, chevauchement.

L'étude des failles et des plis à l'échelle naturelle se fait par des observations effectuées sur le terrain. Par cette approche, qui est essentielle, les géophysiciens, dressent un constat des conséquences des mouvements des plaques sur le relief actuel. Leurs analyses leur permettent ainsi de retracer, ou d'intuire, une partie de l'histoire du milieu étudié [42].



FIG. 7.3 – Différents types de plis : 1) le pli droit et symétrique 2) le pli déjeté 3) le pli renversé 4) le pli couché.

L'une des techniques en laboratoire pour reproduire et comprendre les phénomènes liés aux mouvements des plaques tectoniques, et permettre ainsi de confirmer ou infirmer des hypothèses faites sur le terrain, est la mise en place de "boîte à sable" plus communément appelé "sandbox" utilisées par M.K. Hubbert dès 1951 [65]. La *sandbox* est une boîte contenant essentiellement du sable dont la taille des grains est variable. Les boîtes sont généralement élancées (pour donner un ordre d'idée le rapport longueur / hauteur (L/H) est souvent supérieur à 8) et peuvent avoir des géométries plus ou moins différentes afin de reproduire le plus fidèlement possible les structures naturelles que les géophysiciens étudient. Le sable est généralement déposé en couches colorées de façon à pouvoir suivre la déformation du milieu et reproduire notamment les plis et failles observées dans les structures géologiques. Les expériences réalisées en laboratoire ont pour but d'étudier divers phénomènes et tester l'influence de certains paramètres comme par exemple le type de sable [95]. L'expérience consiste généralement à faire subir à l'échantillon un raccourcissement, dans le champ de la pesanteur, de façon quasi-statique afin de comprimer les strates et ainsi observer les plis pouvant se développer dans celles-ci. Cobbold *et al.* [34, 35] couplent des boîtes à sable avec un fluide afin de définir l'influence de celui-ci sur la formation des plis et failles et autres détachements pouvant intervenir lors d'essais de raccourcissement. Dans [121], le modèle de boîte à sable est légèrement modifié en donnant à la paroi venant en butée différentes inclinaisons. Cette inclinaison est alors reliée à la structures des plis et des failles obtenues. Koyi [85] étudie l'épaisseur du prisme obtenue par raccourcissement ainsi que l'évolution des différentes strates; le prisme étant l'amas se créant au niveau de l'avancée. L'aire de la surface libre du prisme est également calculée pour déterminer son évolution en fonction de la déformation. Les boîtes à sable sont souvent utilisées pour modéliser des structures naturelles : dans [86], Kukowski *et al.* utilise le modèle MPI (c.f. figure 7.4) pour étudier la chaîne méditerranéenne occidentale (Western Mediterranean Ridge - WMR).

Les modèles analogiques ne sont plus les seuls outils pour étudier ces systèmes. Il peut parfois être difficile de trouver des matériaux granulaires avec des coefficients internes [95] correspondant aux structures que l'on veut étudier. Encore une fois pour venir épauler les expériences *in labo*, la modélisation numérique apparaît comme un formidable outil. Les premiers modèles numériques développés utilisent une approche par éléments finis, plus flexibles que les modèles analogiques car ils peuvent jouer avec différentes conditions limites et types de rhéologie [73]. Cette approche a toutefois ses limites : pour arriver à



FIG. 7.4 – Modèle analogique de boîte à sable utilisé par Kukowski *et al.* lors de l'étude du Western Mediterranean Ridge [86].

capturer la localisation des failles et les phénomènes qu'elles engendrent, le maillage doit être extrêmement fin entraînant des calculs très coûteux. De plus les zones où les crises se produisent ne sont généralement pas connues par avance, et seule une extrapolation de leur localisation peut être effectuée. C'est ainsi que lorsque les failles se succèdent les modèles continus arrivent mal à capturer ces successions de failles. Pour pouvoir espérer les observer, l'approche numérique par éléments discrets nous semble plus pertinente. Elle permet également de traiter plus finement les problèmes de localisation, chaque élément ayant un mouvement propre. Parmi les études numériques réalisées par éléments discrets notons celles de Burbidge et Braun [25], utilisant l'approche *DEM* pour étudier l'évolution des prismes et des failles lors d'expériences de raccourcissement sur des modèles MS. Finch *et al.* [49] utilise la même approche numérique pour prédire les déformations liées aux failles inverses se développant en profondeur lors de la séparation de plaques. De la cohésion entre particules est introduite dans le modèle numérique. Leurs observations diffèrent de certains modèles par éléments finis mais reproduisent des expériences analogiques.

Malgré l'intérêt que nous portons aux mouvements tectoniques, l'étude que nous entreprenons doit être vue comme une approche "naïve" du problème, ne connaissant pas ou très peu tous les phénomènes plus ou moins complexes dont débordent la géophysique. Nous présentons les prémices des études entreprises sur les boîtes à sables d'un point de vue numérique. Le but de ce chapitre est prospectif montrant qu'il est possible de travailler avec des échantillons conséquents, permettant d'observer des phénomènes locaux non triviaux, en des temps *CPU* raisonnables. Dans ce qui suit l'influence du nombre de particules pour l'observation des plis et des failles, ainsi que la géométrie des particules, ont été analysées. Deux études plus spécifiques sur la propagation des failles et la création des plis sont également présentées.

7.2 Investigations numériques

Avant d'aborder l'étude de la propagation de failles et de plis par une approche par éléments discrets, les questions du nombre de particules (ou de la taille) et de leur géométrie doivent être posées. Afin de faire au mieux nous effectuons une étude préliminaire pour essayer de fournir une réponse à ces deux questions.

7.2.1 Choix du nombre de particules

Le choix d'un Volume Élémentaire Représentatif (VER) est une question qui revient souvent lors de l'étude des milieux granulaires. C'est une question que nous devons également nous poser lors de simulations de boîtes à sable numériques via une approche par éléments discrets. Ce VER peut très bien être différent pour l'étude des failles, des plis ou des profils à surface libre.

Afin de quantifier, dans un premier temps, le nombre de particules nécessaires à l'observation nette de plis lors de raccourcissement, nous avons modélisé une boîte de 6 mètres de long sur 1.15 mètre de haut (rapport $L/H = 5$) en utilisant différentes tailles de particules, 0.2, 0.1, 0.05 et 0.025 *cm* de diamètre correspondant respectivement à un nombre de particules égal à 750, 2 500, 10 000 et 40 000. Après stabilisation de l'échantillon, un raccourcissement de 60% est effectué. La configuration de chaque échantillon en fin de simulation est représentée sur la figure 7.5.

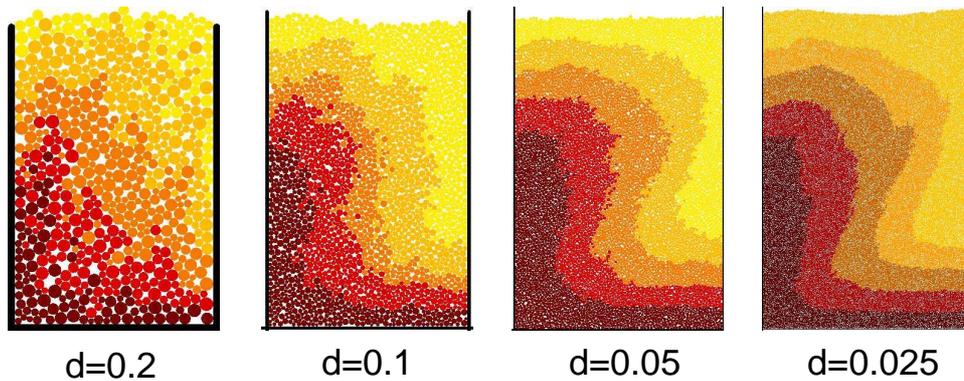


FIG. 7.5 – Profils lors de simulation de raccourcissement effectuées avec différentes tailles de particules.

Sur les quatre échantillons les similitudes sont assez grandes. Les allures des différentes strates sont semblables même si pour l'échantillon (a) de la figure 7.5 les effets de parois sont encore importants, rendant les profils très chaotiques. Parmi les différences entre les différents échantillons l'augmentation de la compacité avec le nombre de particules en est une. On peut remarquer également l'apparition de protubérances à l'intérieur de l'échantillon (*d*), phénomène que l'on n'observe pas sur les trois autres simulations. L'utilisation d'un plus grand nombre de particules nous permet d'observer des formations plus fines, au prix bien entendu d'un temps de calcul plus conséquent. Cependant la géométrie de l'échantillon, peu élancée, et le fort raccourcissement ne permettent pas l'apparition de formations fines. La taille minimale apparaît comme étant celle de l'échantillon (b), soit 2 500 particules pour un rapport L/H égal à 5.

Pour compléter ces premières observations, le procédé précédent est appliqué à une boîte de 100 *m* de long sur 5 *m* de hauteur afin de faire varier le rapport L/H , égal à 20 dans

ce cas. Nous créons quatre échantillons dont le diamètre moyen des particules est respectivement de 0.9, 0.45, 0.225 et 0.1125 mètres ce qui correspond à un nombre de particules égal à 560, 2 400, 9 900 et 40 000 (figure 7.6).

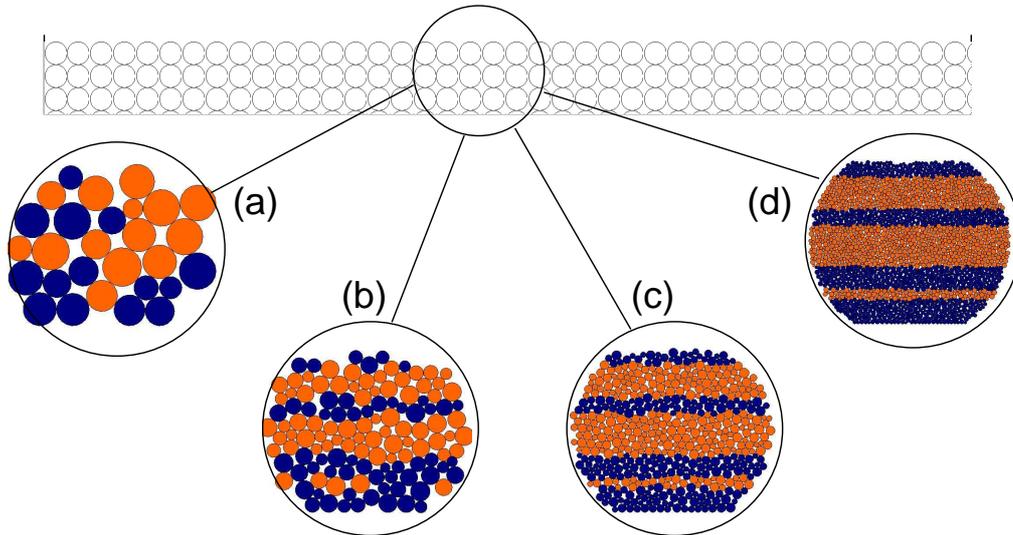


FIG. 7.6 – Zoom sur le nombre de particules dans l'épaisseur de l'échantillon au repos pour les différentes simulations : $n_b = 560$ (a), $n_b = 2\,400$ (b), $n_b = 9\,900$ (c) et $n_b = 40\,000$ (d).

L'échantillon (a) n'a pas beaucoup de signification. En effet la boîte est très allongée et la hauteur ne fait que quelques diamètres de bille (moins de 10). L'observation de strates et de plis n'est alors pas pertinente pour cet échantillon, l'effet des parois étant trop prépondérant sur les arrangements entre particules. Dans ce qui suit nous ne présenterons pas les résultats obtenus avec cet échantillon. Sur les échantillons (b), (c) et (d) nous effectuons un raccourcissement. Nous présentons sur la figure 7.7 les différents échantillons après un raccourcissement de 12.5 m. Nous nous focalisons tout d'abord sur deux zones. La première, la zone A, est localisée près de la paroi en mouvement, tandis que la seconde, la zone B est un peu plus en avant. On observe des similitudes dans la zone A pour les échantillons (c) et (d) : le développement de plis symétriques concentriques. Sur l'échantillon (b), le nombre de particules n'est pas suffisant pour observer cette formation. Dans la zone B, ce sont des micro-failles inverses qui apparaissent dans l'échantillon (d), ainsi que dans le (c) mais avec moins de netteté. Il est impossible d'observer un tel phénomène dans l'échantillon (b).

Pour finir avec ces échantillons, nous pointons également l'apparition d'un nouveau pli situé en aval de l'avancée (zone C) : cette formation n'est visible que dans le plus grand échantillon soulignant ainsi l'importance de l'utilisation d'échantillons conséquents. Pour cette géométrie, les échantillons (a) et (b) sont donc inutilisables pour observer finement l'évolution des strates. L'échantillon (c) apparaît donc comme l'échantillon minimal, ce

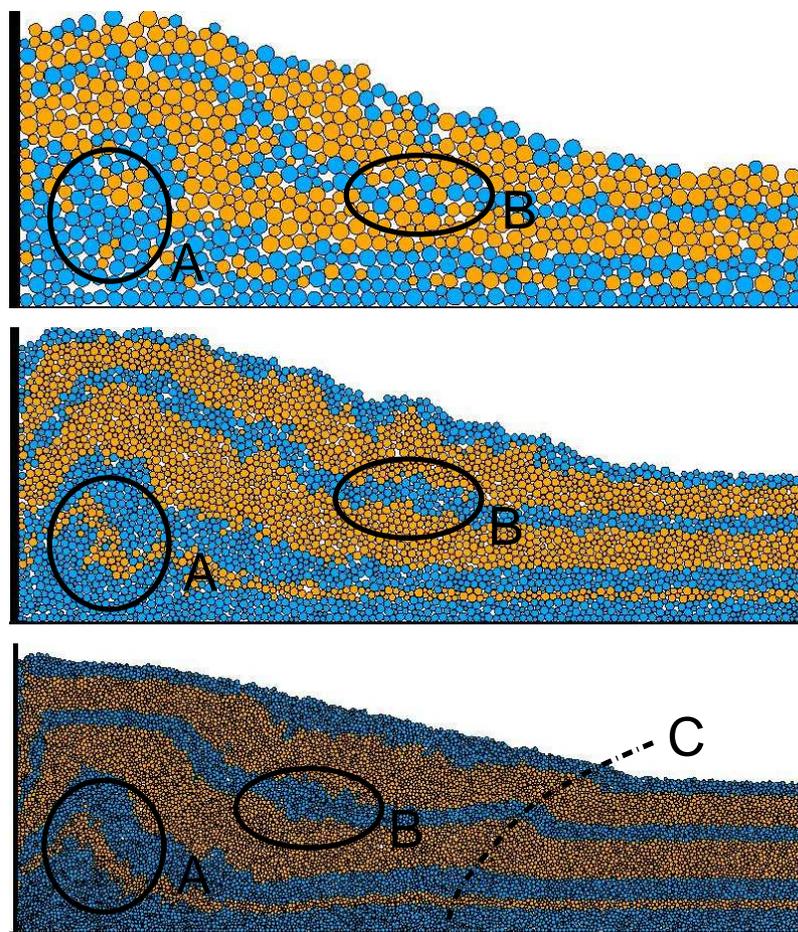


FIG. 7.7 – Configuration des trois échantillons après un déplacement de 12.5 m .

qui correspond à un nombre minimal d'environ 10 000 particules. En reprenant le premier échantillon, nous avons donc besoin d'au-moins 2500 particules si L/H est égal à 5 et 10 000 si L/H vaut 20. Pour une première approximation (interpolation linéaire) on peut donc définir un nombre minimal de particules à utiliser en fonction de L/H : $n_{min} = 500L/H$ pour commencer à observer proprement des successions de strates.

7.2.2 Géométries

Dans les modèles analogiques de boîtes à sable, Lohrmann *et al.* [95] ont souligné l'influence des propriétés du sable sur le comportement global du système. Les propriétés mécaniques de nos particules influencent nos échantillons ; des coefficients de frottement différents ne conduisent pas aux mêmes observations. Mais de quelle façon joue la géométrie des particules sur la formation de failles ou de plis ? Nous modélisons une boîte de 100

mètres de long par 15 mètres de haut ($L/H = 6.7$) en utilisant 15 000 disques et 100 mètres de long par 12 mètres de haut ($L/H = 8.3$) en utilisant 15 000 polygones ; à rayon d'encombrement égal, l'aire d'un polygone est plus faible que celle d'un disque, d'où la variation de L/H . Les deux échantillons sont colorés en strates pour des raisons de visualisation ; aucun paramètre mécanique (frottement, restitution) ne leur est associé. Le frottement inter-grains est identique dans tout l'échantillon. Nous faisons subir à chacun d'eux différents types de sollicitation : un développement de faille (surélévation de bloc) et un raccourcissement. Pour la propagation de faille nous tenons compte de l'élançement en déplaçant la paroi gauche inférieure proportionnellement à l'épaisseur de chaque échantillon.

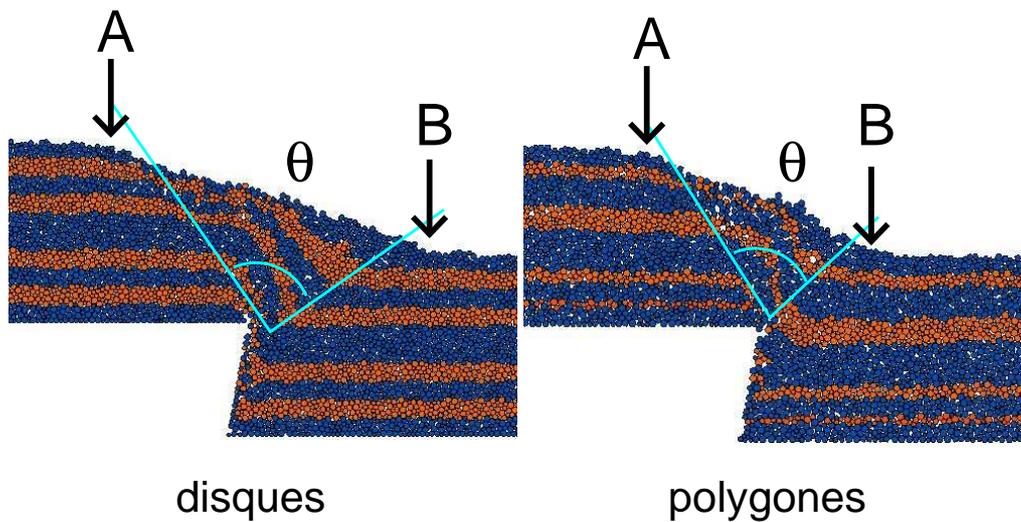


FIG. 7.8 – Développement d'une faille dans un échantillon de 15 000 disques (gauche) et polygones (droites).

Nous représentons sur la figure 7.8 les profils en fin de simulations pour les deux échantillons. Notons que la coloration des strates n'est pas rigoureusement identique, différence liée au dépôt de l'échantillon. Malgré cela on peut observer quelques différences entre les deux échantillons. Nous pointons sur les deux clichés le début (point A) et la fin (point B) de la pente liée à la propagation de la faille et signalons par θ l'angle de la zone plissée. Contrairement au point A qui, sur les deux simulations, est situé sur la même verticale (proche de l'angle d'avancée), le point B est plus proche de l'avancée pour l'échantillon constitué de polygones que pour celui constitué de disques. Ainsi θ_d (disques) est supérieur à θ_p (polygones), ce qui rend une pente plus abrupte pour l'échantillon polygonal.

La figure 7.9 montre les échantillons précédents après une simulation de raccourcissement. Ici bien que les élançements soient différents les phénomènes observés semblent être pilotés par la géométrie des particules. Là encore des différences apparaissent lors de la formation de plis. La première est la taille de l'amas dû à l'avancée de la paroi : celui-ci est beaucoup plus long pour les disques que pour les polygones (position du point A). À droite du point

A, la surface libre reste horizontal pour les disques, mais prend une pente non nulle pour les polygones.

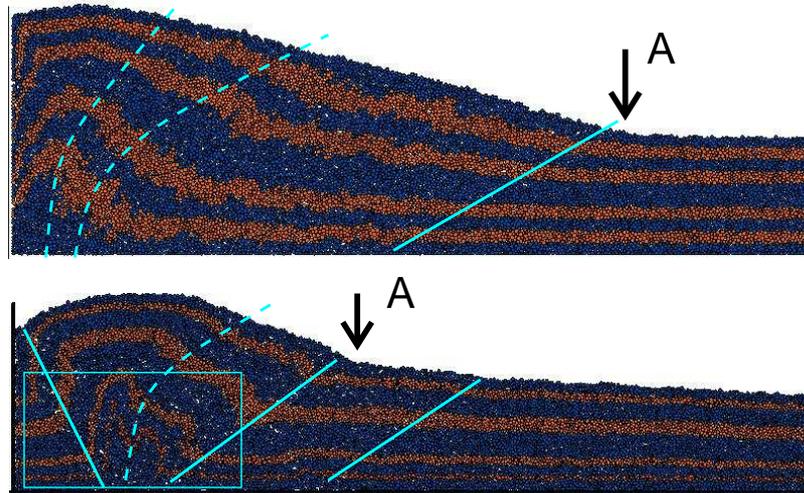


FIG. 7.9 – Profil final de deux simulations de raccourcissement comprenant 15 000 particules circulaires (haut) et polygonales (bas).

Les formations de plis sont également différentes. Dans l'échantillon constitué de polygones, on observe une "zone morte" à gauche, de forme triangulaire, se déplaçant avec la paroi mobile, zone qui n'apparaît pas avec les disques. La formation de plis concentriques se fait très proche de la paroi avec les disques et après la zone morte pour les polygones. La géométrie moins régulière des polygones est favorable à la création d'arches pouvant entraîner des blocages. Les déformations sont alors beaucoup moins souples, expliquant le côté anguleux des plis et micro-failles dans l'échantillon de polygones.

Pour nous plonger un peu plus en détail dans cette comparaison nous regardons l'évolution de l'orientation des normales au contact dans une portion de l'échantillon pour les deux types d'échantillons (figure 7.10) ainsi que l'orientation de la contrainte principale dans cette même zone (figure 7.11). La zone étudiée, fixe, est la zone où se trouve, en fin de simulation, les plis concentriques. On représente à quatre instants de la simulation l'orientation des contacts : à 25, 50, 75 et 100% de la simulation. L'évolution est assez similaire pour les deux échantillons. A 25% de la simulation, la zone est peu affectée par le déplacement de la paroi (encore moins avec les disques qu'avec les polygones), et l'orientation des contacts garde la signature du dépôt. Après 50% de simulation la zone est affectée par le déplacement de la paroi et les contacts viennent s'orienter de façon à faire opposition à la paroi. Les facettes des polygones induisent une direction principale du tenseur de texture plus importante que pour les disques. En fin de simulation les contacts sont en majeure partie horizontaux. Toutefois la direction principale est beaucoup plus marquée pour l'échantillon composé de polygones qui est sensible à la poussée de la paroi

beaucoup plus rapidement que l'échantillon composé de billes. Les réarrangements sont beaucoup moins faciles avec les polygones. Les billes pouvant mieux se réarranger, les contacts pointent moins vers la paroi mobile.

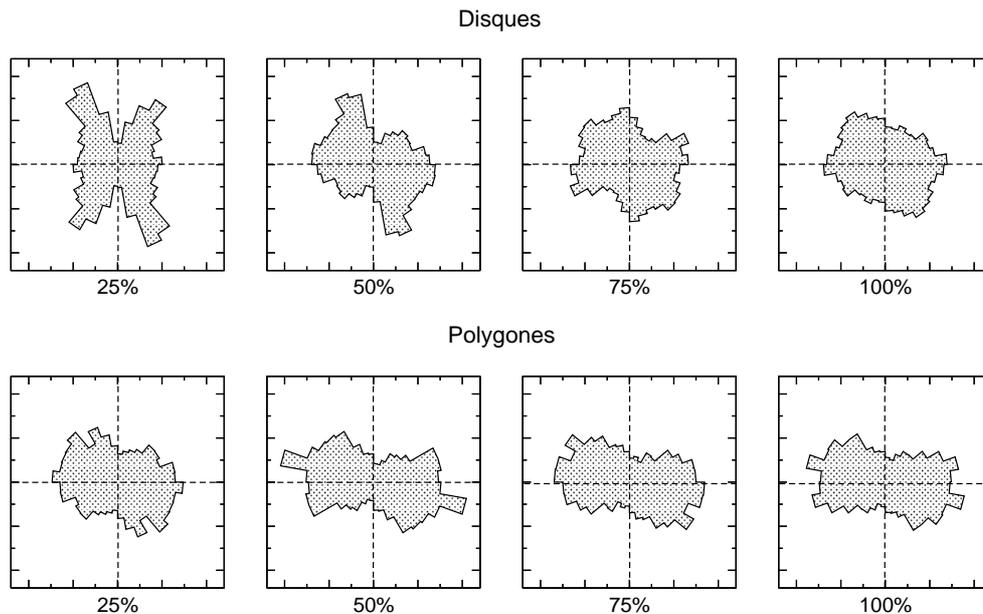


FIG. 7.10 – Evolution de l'orientation des normales au contact dans une portion de l'échantillon pour les deux types d'échantillons.

Nous évaluons également la première direction principale du tenseur de contrainte σ dont l'évolution est représentée sur la figure 7.11. On retrouve un comportement similaire à l'orientation principale du tenseur de fabrique : la direction de sollicitation est proche de l'horizontale, le réseau de contacts s'opposant au sens de progression de la paroi.

Le comportement mécanique (tenseurs de fabrique et de contrainte) du matériau présente donc de fortes similitudes malgré la différence de géométrie. Cependant celle-ci influence énormément le profil des plis et de l'échantillon. Ainsi même si à l'intérieur d'une zone on retrouve des comportements similaires, à l'échelle globale les comportements sont différents.

Le choix de la géométrie semble donc important, celui-ci pilotant certains mécanismes. Ceci étant, pour une première approche, nous allons nous affranchir d'une géométrie polygonale, pour se focaliser sur des échantillons de disques, étant maintenant conscients des différences pouvant se produire. L'influence du frottement n'a pas été étudié dans le détail, mais c'est également un paramètre important à prendre en compte.

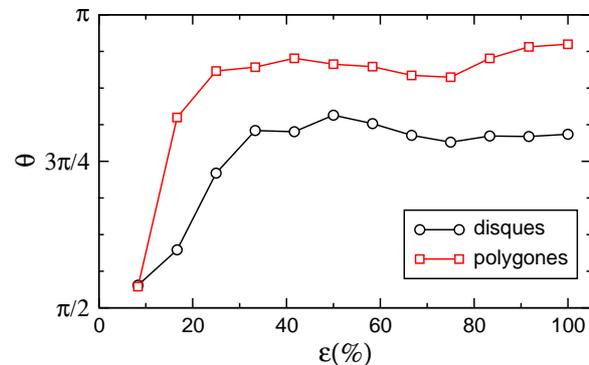


FIG. 7.11 – Evolution de la direction principale du tenseur de contrainte σ pour les deux échantillons.

7.3 Propagation de failles

7.3.1 Modèle

La compréhension du développement de failles dans les structures géologiques est l'une des préoccupations des géophysiciens s'intéressant aux mouvements des plaques tectoniques. On trouve ainsi quelques modèles pour décrire le développement des failles et le comportement du milieu autour de celles-ci. Un des premiers modèles est le modèle "Parallel-Kink" introduit par Suppe (1990), décrivant la faille essentiellement par des droites parallèles. Le modèle "Tri-shear" est introduit par Erslev peu de temps après (1991). De nombreuses extensions sont portées à ce deuxième modèle, plus pertinent que le premier [73]. Le modèle "Tri-shear" consiste à partager la région de la faille en trois parties (c.f. figure 7.12).

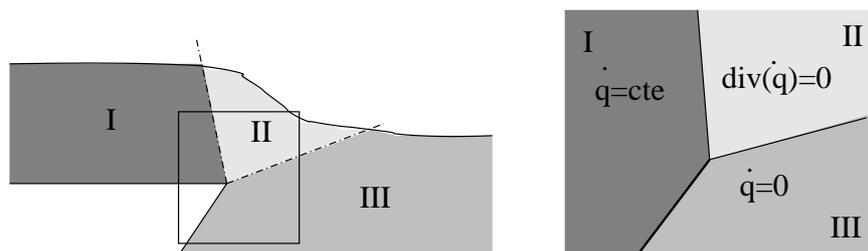


FIG. 7.12 – Modèle "Tri-shear" : décomposition géométrique de la zone de faille. La partie I possède un mouvement de solide rigide donc la vitesse est celle de la propagation de la faille. La zone III a une vitesse nulle. Le champ de vitesse de la zone II satisfait à la continuité du champ de vitesse global.

De nombreuses hypothèses sont faites sur le matériau de la zone II : l'invariance du volume ($\rho = cte$) conduisant à l'équation $div(\dot{\mathbf{q}}) = 0$, la vitesse le long de la paroi gauche est constante et égale à la vitesse de propagation de la faille et la vitesse le long de la paroi droite est nulle. D'autres hypothèses peuvent être faites pour reconstruire la zone de faille : on trouve différentes approches dans [58]. D'un point de vue numérique, ce sont essentiellement des approches par éléments finis qui sont utilisées [28, 73, 137] fournissant des résultats pertinents avec les observations des structures naturelles. Plus récemment des méthodes par éléments discrets ont été utilisées [25, 49], permettant de prendre en compte les dislocations locales, difficiles à capturer en éléments finis. Nous allons prendre également cette voie. Nous présentons tout d'abord la méthodologie utilisée dans les différentes simulations, puis les simulations effectuées, illustrées par différents clichés, puis nous terminerons par une discussion sur l'éventuel apport de notre méthode au modèle *Tri-shear*.

7.3.2 Méthodologie

Nous avons modélisé une boîte rectangulaire de 150 *m* de long sur 15 *m* de hauteur, remplie par des disques polydispersés de rayon moyen 0.20 ± 0.06 *m*. Le frottement entre grains est pris égal à 0.4. Le procédé consiste à soulever une partie de la boîte afin de créer une faille dans la zone centrale de l'échantillon, aboutissant ainsi à une zone surélevée. Le déplacement est effectué à vitesse constante. La simulation comprend 2000 pas de temps de longueur $\Delta t = 3 \cdot 10^{-2}$ sec. On utilise pour la simulation l'algorithme PCPG (pour une précision de $1.66 \cdot 10^{-4}$), exécuté en parallèle sur 16 processeurs sur *SGI Origin3800*. En utilisant l'algorithme CPG sur 16 processeurs, le temps de simulation est relativement court (moins de 5*h*) allant dans le sens de l'étude numérique effectuée au chapitre 4. Remarquons que nos temps de simulations sont plus courts que ceux obtenus dans les simulations effectuées dans [49] en utilisant l'approche DEM, et ceci malgré un nombre de particules 2.5 fois plus important. Rappelons que la démarche NSCD permet d'appréhender en outre en toute rigueur les efforts de contact non réguliers. Lors des différentes simulations, nous récupérons plusieurs configurations du système afin d'en analyser son évolution. On donne à l'angle d'avancée du plateau montant les valeurs 30°, 45°, 60° et 80°. La composante horizontale vaut respectivement 0.2806, 0.1625, 0.09382 et 0.02806 *m.s*⁻¹. La composante verticale de la vitesse de propagation de la faille est égale à 0.1625 *m.s*⁻¹ pour les quatre simulations. Le tableau 7.1 présente un récapitulatif des différents paramètres et temps de calculs obtenus pour les différentes simulations.

7.3.3 Analyses

Pour observer l'évolution de la zone II décrite dans le modèle *Tri-shear*, à différents instants de l'évolution de la faille nous localisons les particules dont la vitesse est supérieure à 1.01 fois la vitesse de propagation de la faille (points rouges) ainsi que les particules dont la vitesse est quasi nulle (0.05 fois la vitesse d'avancée) et inférieure à 0.99 fois la vitesse de propagation (points noirs). Par ce procédé nous espérons faire apparaître la zone

Simulation	$\langle it \rangle$	Temps CPU	solveur
Faille- $\theta = 30$	260	4h44 (75h34min)	42.3%
Faille- $\theta = 45$	222	5h00 (78h40min)	33.2%
Faille- $\theta = 60$	183	4h25 (69h55min)	26.2%
Faille- $\theta = 80$	168	3h57 (63h00min)	29.8%

TAB. 7.1 – Temps de calcul des différentes simulations effectuées sous *SGI – Origin3800* ($P = 16$)

décrite par le modèle. Nous avons représenté la surface libre de l'échantillon par une ligne pointillée. L'axe d'avancée de la faille est signalé par la ligne mixte.

Pour les différentes simulations, nous constatons que la zone "en mouvement" (différent du solide rigide) correspond à une zone plus ou moins triangulaire dont l'un des sommets est proche de l'angle d'avancée du bloc supérieur (c.f. figure 7.13 et 7.14). La taille de la zone est plus ou moins grande, certains réarrangements dans le bloc supérieur peuvent entraîner des fluctuations de vitesse. Si l'on se restreint à une zone proche de la faille, nous trouvons également des zones où les particules ont une vitesse supérieure à la vitesse de propagation, phénomène non observable dans les modèles par éléments finis. Ce phénomène correspond à des amas de particules qui se décrochent du bloc soulevé et se déplacent en direction du bloc abaissé. En chute libre leur vitesse va devenir supérieure à la vitesse du bloc soulevé puis chuter pour atteindre une vitesse moindre. On retrouve ainsi les décrochements souvent observés dans les zones faillées et les affaissements se produisant dans les structures géologiques. Les décrochements sont d'autant plus importants que l'avancée de la faille est importante ; ce phénomène est observé plus particulièrement pour les grandes déformations. Le dénivelé entre le bloc soulevé et le bloc abaissé augmentant, l'angle de surface libre de la zone II devient plus important, rendant la zone instable et facilitant ainsi les décrochements. Pour les angles importants (c.f. figure 7.14) nous pouvons également observer des décrochements à l'intérieur de la zone II. Les affaissements sont liés à des réarrangements des particules en profondeur, créant des zones moins denses dans lesquelles les particules supérieures peuvent s'engouffrer. On peut alors observer des successions de réarrangements jusqu'à obtenir une stabilisation des zones affectées de l'échantillon.

Nous effectuons un zoom à l'intérieur d'un des échantillons ($\alpha = 60^\circ$) afin d'observer localement l'évolution des grains. Le zoom est représenté par la figure 7.15 capturant un instant typique de la simulation ($N_{step} = 1400$). On représente sur la figure deux lignes séparatrices séparant la zone triangulaire du bloc soulevé ayant un mouvement de solide rigide (ligne séparatrice 1 - LS1) et du bloc abaissé ayant une vitesse nulle (ligne séparatrice 2 - LS2). La direction de propagation de la faille est également représentée. On représente

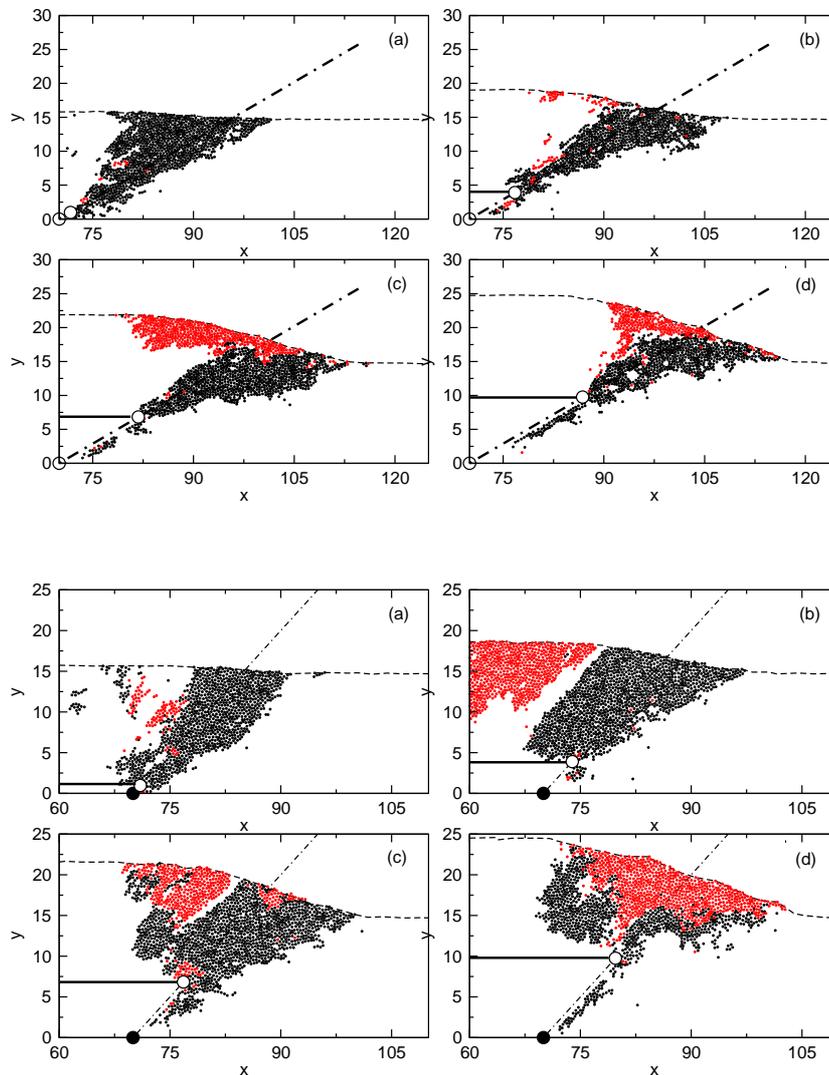


FIG. 7.13 – Evolution de l'échantillon pour une avancée suivant un angle $\alpha = 30^\circ$ (haut) et $\alpha = 45^\circ$ (bas). Les différents clichés correspondent à 10%(a), 40%(b), 70%(c) et 100%(d) de temps de simulation.

ensuite le champ de vitesse instantanée des particules, affiché en échelle logarithmique. Les particules à gauche de LS1 possèdent toute une vitesse colinéaire à la vitesse de propagation de la faille de même intensité. A droite de LS2, les particules ont toutes de très faibles vitesses. Leur direction ne les pousse pas vers la droite mais en direction de la faille, créant une zone de "turbulence" proche de LS2. Dans la zone triangulaire, la vitesse des particules n'est plus uniforme. On observe à la surface libre des particules ayant une

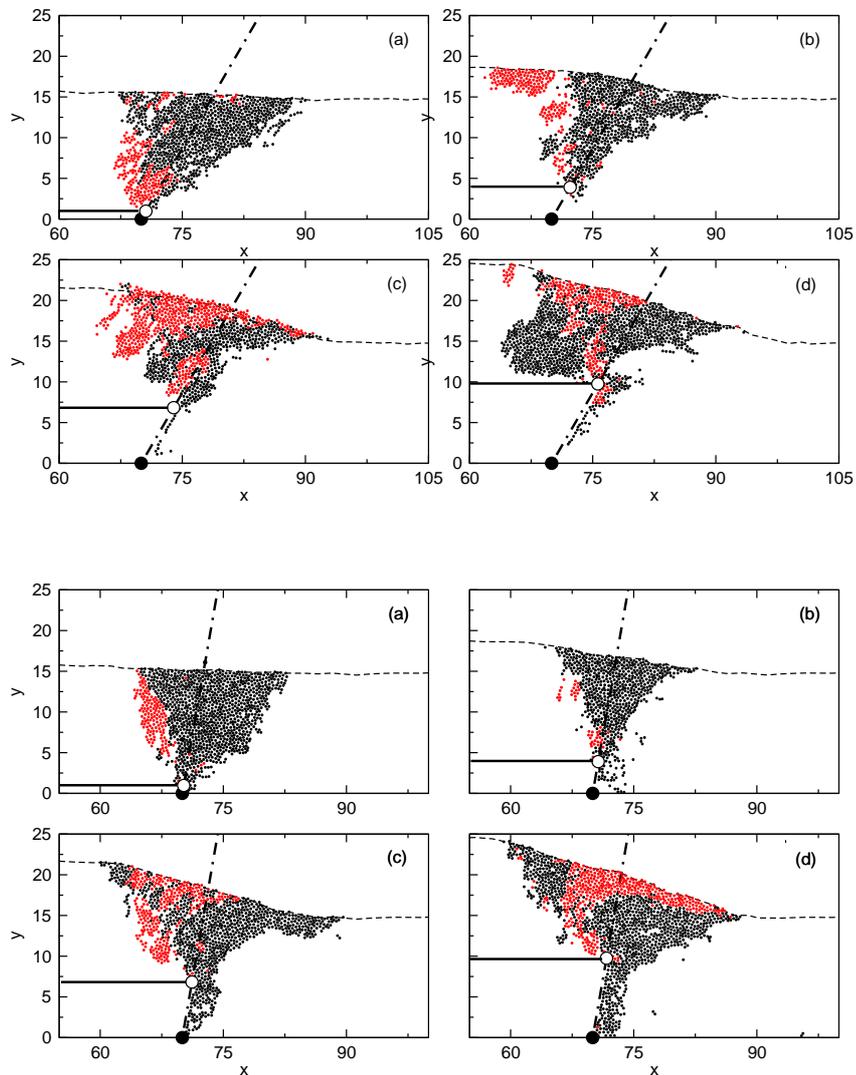


FIG. 7.14 – Evolution de l'échantillon pour une avancée suivant un angle $\alpha = 60^\circ$ (haut) et $\alpha = 80^\circ$ (bas). Les différents clichés correspondent à 10%(a), 40%(b), 70%(c) et 100%(d) de temps de simulation.

vitesse supérieure à la vitesse de propagation de la faille. Ces particules ont un mouvement parallèle à la surface libre. A l'intérieur de la zone, la direction des vitesses des particules est plus hétérogène, mais favorise la direction d'avancée de la faille.

Avant de conclure nous regardons plus en détail les échantillons, en se fixant sur différentes zones de celui-ci. Pour cette étude nous regardons les échantillons en fin de simulation et nous faisons des zooms sur différentes zones de l'échantillon comme représentées sur

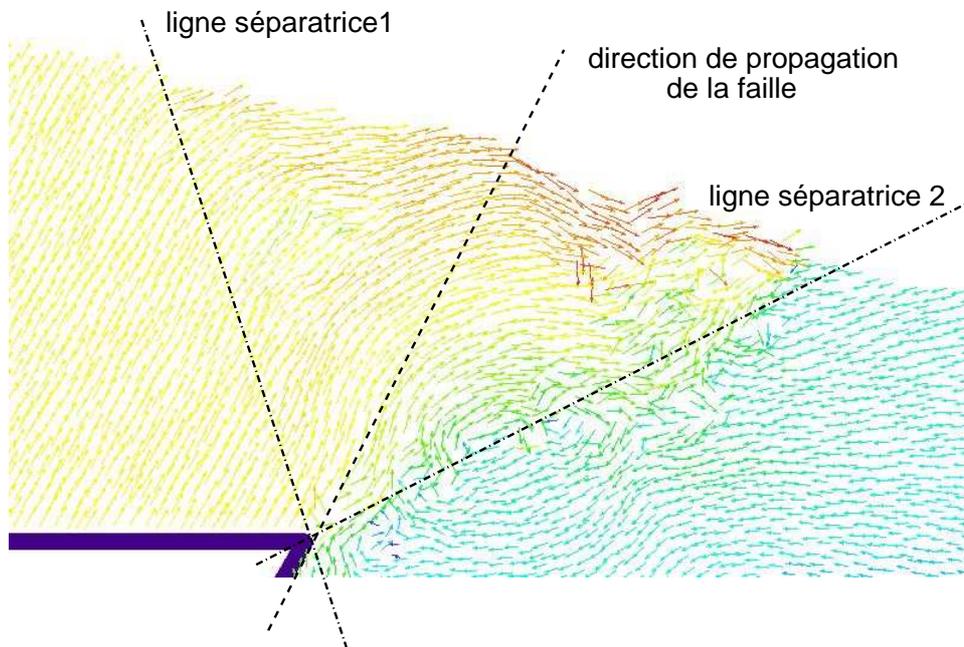


FIG. 7.15 – Champs de vitesses de la faille d'angle de propagation $\alpha = 60^\circ$ pris à 70% de de temps de simulation.

la figure 7.16 pour lequel $\alpha = 80^\circ$: cet échantillon nous sert d'exemple illustratif (les résultats sont similaires pour les autres échantillons moyennant l'angle de la faille).

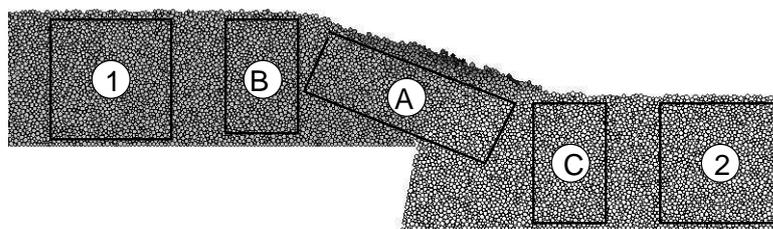


FIG. 7.16 – Localisation des zones d'études de l'échantillon $\alpha = 80^\circ$.

Pour chaque zone nous déterminons l'orientation des normales au contact en début et fin de simulation. La figure 7.16 collecte les différents résultats. Les courbes noires représentent l'orientation des contacts en début de simulation et les rouges l'orientation en fin. Les zones 1 et 2 qui sont situées en amont et aval de la faille ne sont pas affectées par la faille, excepté la zone 1 qui subit un mouvement de solide rigide. Ces deux zones restent

invariante au cours de la simulation nous pouvons regarder l'évolution des quantités qui leurs sont rattachées. La compacité des deux zones reste constante au cours de la simulation (0.82 pour les deux zones) ainsi que les contraintes principales du tenseur de contrainte notées respectivement σ_1 et σ_2 : le rapport q/p est égal à 0.33.

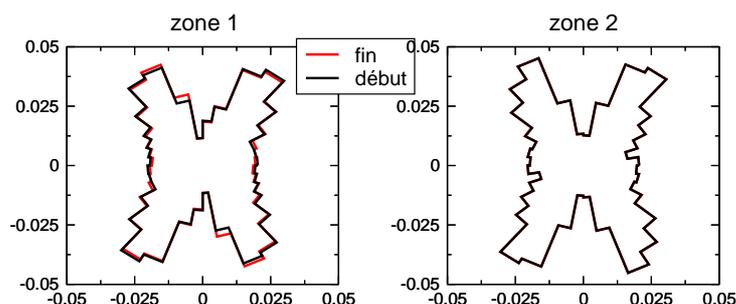


FIG. 7.17 – Orientation des normales aux contacts en début et fin de processus dans les zones 1 et 2 représentées sur la figure 7.16.

Les zones A , B et C plus proches de la faille sont étudiées uniquement en fin de simulation. Pour chaque zone, nous représentons l'orientation des normales aux contacts et la direction principale des contraintes (figure 7.18). Les zones B et C situées respectivement à gauche et à droite de la faille gardent, malgré leur proximité, la signature du dépôt comme les zones 1 et 2. La direction principale des contraintes θ_1 associée à σ_1 a un angle de 131° pour la zone C et de 123° pour la zone B . La zone C , située en contre-bas, porte plus de charge que la zone B . La contrainte principale de la zone C est supérieure de 15% à celle de la zone B .

La direction principale de la zone A , chevauchant la faille, est voisine de celle de la zone C ($\theta_1 = 129^\circ$). La zone étant soumise à des déformations (écoulements de surface) beaucoup plus importantes que les autres zones, la compacité y est moins importante ainsi que les contraintes (quatre fois moindre que les contraintes en amont et aval). Le développement de la faille affecte peu le milieu autour duquel elle se propage. Dans la zone de propagation le milieu se réarrange afin de supporter au mieux les déformations engendrées. La faille crée une zone fragile, la zone triangulaire, de faible compacité (environ de 0.7) dont les contacts sont orientés suivant l'angle de surface libre.

Les résultats obtenus présentent des similitudes avec les simulations numériques par éléments discrets [49] mais également par éléments finis [28, 73]. Par notre approche nous sommes aptes à capturer les réarrangements locaux et les crises dynamiques locales. Nous pouvons également venir étudier la structure en analysant les contraintes, la texture ainsi que la compacité.

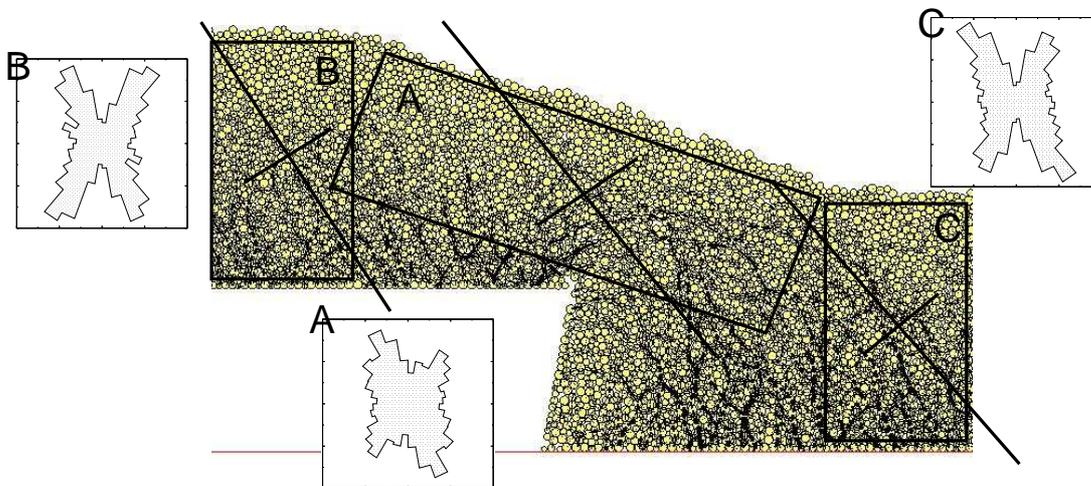


FIG. 7.18 – Zoom sur l'échantillon $\theta = 80^\circ$ en fin de simulation. Le réseau des contacts est représenté ainsi que l'orientation des normales aux contacts dans les zones *A*, *B* et *C*. La direction des contraintes principales est également indiquée sur la figure pour chaque zone.

7.4 Formation de plis

7.4.1 Entrée en matière

La part de l'observation en géophysique est très importante. Les observations effectuées sur le terrain guident toutes les recherches pouvant être effectuées expérimentalement, analytiquement et/ou numériquement. Les observations se font en général sur des structures ayant subi des éboulements, des glissements de terrains, des subductions ou tout autre phénomène dont les traces restent visibles sur les structures naturelles actuelles. Parmi ces traces visibles l'apparition des plis, liée la plupart du temps à des convergences de plaques, est l'une des plus spectaculaires, pouvant se combiner à la formation de failles. En laboratoire la mise en place de dispositifs tels que les boîtes à sables (figure 7.4) permet de comprendre la formation des plis en reproduisant à petite échelle (le *cm*) ces phénomènes de grande taille (le *km*).

D'un point de vue numérique l'observation de ces phénomènes demande plusieurs choses. Le modèle doit permettre de capturer les réarrangements locaux, sans connaître a priori le comportement du milieu : une approche par éléments discrets est donc la mieux adaptée. Le nombre de particules en hauteur doit être assez conséquent, pour pouvoir observer finement l'évolution des strates tout au long du processus. Suivant les dimensions de la boîte utilisée, ce nombre peut devenir très vite important ($> 40\,000$) pouvant ainsi entraîner des calculs coûteux. Cependant avec une telle approche, en plus des observations topologiques, nous pouvons compléter l'étude grâce à un traitement de données mécaniques au niveau local (contraintes, texture).

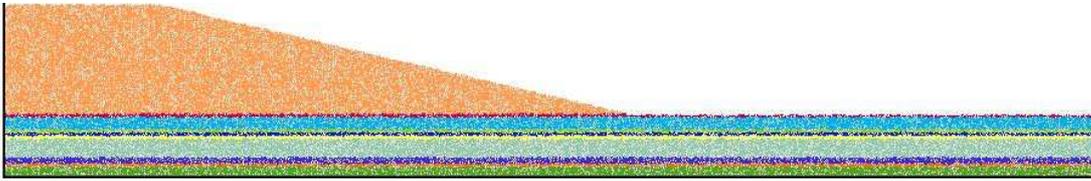


FIG. 7.19 – Boite à sable numérique composée de 43 000 particules.

Nous avons donc modélisé une boîte à sable d'un kilomètre de long (figure 7.19). La partie gauche est constituée d'une butée faisant un angle de front initial de 10° et occupant une longueur de 300 m . L'échantillon est coloré en couches. Les différentes couches ont toutes les mêmes propriétés mécaniques. Le frottement entre particules est égal à 0.4 . La paroi de gauche est ensuite animée d'une vitesse constante. Une évacuation est faite au pied de la paroi permettant ainsi aux particules de ne pas se retrouver coincées dans l'angle d'avancée, zone pouvant vite devenir un "nid" à violations et détruire la qualité de la simulation. Le processus prend fin après un déplacement total de la paroi de 300 m . La simulation est effectuée en $90h$ sous *SGI Origin 3800* avec 16 processeurs utilisant l'algorithme introduit au chapitre 4.

7.4.2 Observations

Nous représentons sur la figure 7.20 différentes étapes de la simulation. L'image du haut nous permet de voir l'évolution des strates après un raccourcissement de $75m$.

A chaque image, l'échantillon subit un incrément de déplacement de $75m$. A droite de chaque image, la position du front de la butée est précisée correspondant à la longueur de la butée (distance du front à la paroi). Lorsque l'incrément de déplacement Δx est égal à $75m$, on observe la formation des premiers plis concentriques déjetés. Pour Δx égal à $150m$, le raccourcissement crée une faille inverse de flan avant dans la zone où les premiers plis se sont formés. La longueur de la butée décroît, dû au chevauchement engendré par la faille : la partie à gauche de la faille chevauche la partie droite de sorte que l'extrémité de la butée n'est que peu affectée par l'avancée de la paroi. Ceci a pour conséquence de faire croître l'angle de front. Pour Δx égal à $225m$, la faille inverse n'évolue plus (la structure est globalement invariante), subissant simplement un mouvement de solide rigide. On observe la formation de nouveaux plis concentriques en aval de cette faille. Le front s'éloigne de la paroi tandis que l'angle de front reste constant. Sur la dernière image, i.e. pour Δx égal à $300m$, nous devinons la formation d'un nouveau pli en aval entraînant une légère déformation des plis en amont de celui-ci.

En nous attardant sur l'échantillon en fin de processus, nous pouvons remarquer la for-

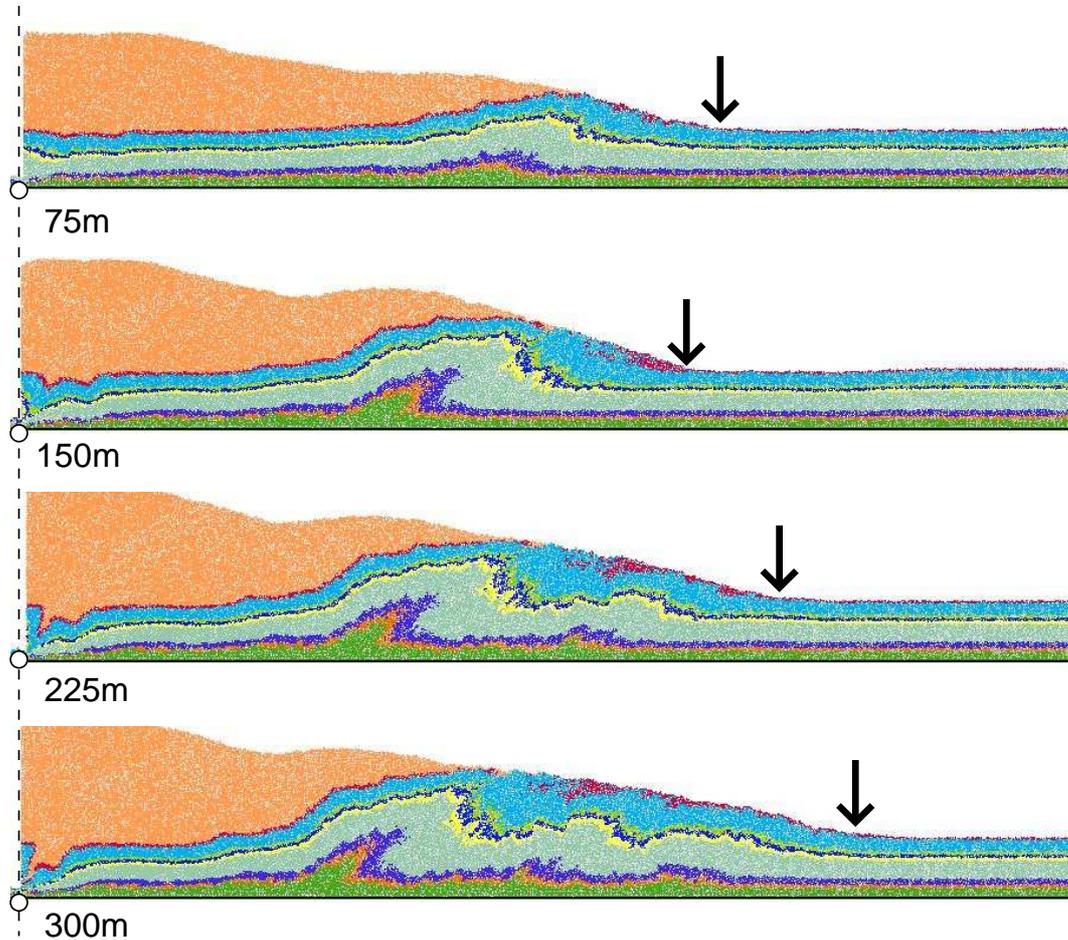


FIG. 7.20 – Echantillon à différents stades de la simulation.

mation de différentes structures que nous soulignons (zones 1, 2, 3, 4 et 5) sur la partie haute de la figure 7.21. Le premier type de structure observé est constitué de plis parallèles (zone 1). Cette zone est la plus proche de la paroi. Le poids de la butée faisant pression sur le fond de l'échantillon, cette zone ne subit pas de grande déformation. Lorsqu'une faille se crée (souligné en noir sur la figure), la structure glisse sur le bloc inférieur de la faille en se déformant d'un angle de 40° tout en gardant les strates parallèles. Lorsque les forces s'opposant au mouvement deviennent moindre la structure locale reprend un angle d'avancée plus aigu (moins de 10°). La zone 2, marquée par une ligne mixte, correspond à un plan de faille séparant la structure 1 et le reste de l'échantillon. A droite de la zone 2, le milieu vient s'écraser et s'aplatir donnant au plan de faille une légère courbure. La structure sous le plan de faille se trouve coincée et contrainte à avancer dans le mouvement engendré par la paroi. On remarque que cette zone de l'échantillon correspond au modèle

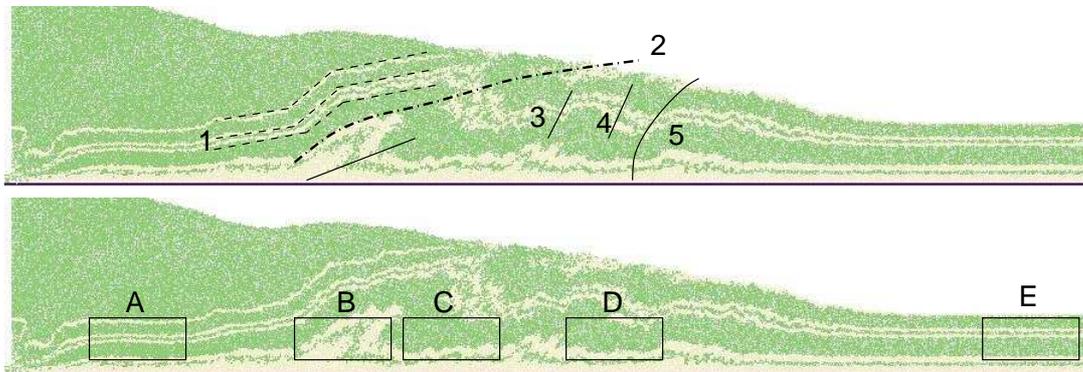


FIG. 7.21 – Echantillon en fin de processus : en haut les différentes structures formées durant la simulation sont marquées ; en bas zoom sur quatre zones étudiées le long du processus.

de faille proposé par Suppe et repris par Hardy [57], qui se développe ici à l'intérieur de la structure et non à la surface libre. Les structures 3, 4 et 5, apparaissant à droite du plan de faille, correspondent à des micro-failles et des initiations de faille. La déformation du milieu dans son état final ne nous permet pas d'en dire d'avantage.

En plus d'une analyse en fin de déformation, il est intéressant de regarder l'évolution de zones situées à l'intérieur de l'échantillon. Pour cela nous avons marqué cinq zones différentes se déplaçant avec la paroi mobile. En fin de processus les cinq zones occupent les places montrées sur la partie du bas de la figure 7.21. Pour chaque zone, pour différents instants nous évaluons la direction principale des tenseurs des contraintes σ et de fabrique Φ (figure 7.22) ainsi que la composante de pression p (figure 7.23). Pour les quatre premières zones, la direction principale des deux tenseurs est comprise entre 140° et 180° . Dans la première zone, la direction principale du tenseur de fabrique n'évolue pas, restant proche de 150° . Celle du tenseur de contrainte fluctue autour de cette valeur avec des écarts pouvant atteindre 10° (en moins). Dans la zone B , localisée sur la faille inverse, l'évolution des directions est croissante et quasi-monotone. La direction principale du tenseur des contraintes est légèrement plus grande que celle du tenseur de fabrique. Dans les zones C et D , le comportement n'est plus monotone. Dans les zones situées en aval de la faille inverse, les directions principales subissent une brusque variation après un incrément de $120m$ (création de la faille inverse), instant à partir duquel les évolutions devient croissantes et monotones, avec cependant quelques fluctuations dans la direction principale du tenseur de contraintes dans la zone C , liées aux observations faites précédemment sur la figure 7.20. La zone E beaucoup plus en aval, garde la signature du dépôt sous gravité ($\theta = 60^\circ/120^\circ$), mais le mouvement en amont rend la structure instable, faisant prendre aux directions principales tantôt la valeur 60 tantôt la valeur 120 . De plus les deux directions sont en déphasage, de sorte que lorsque la direction principale des contraintes est

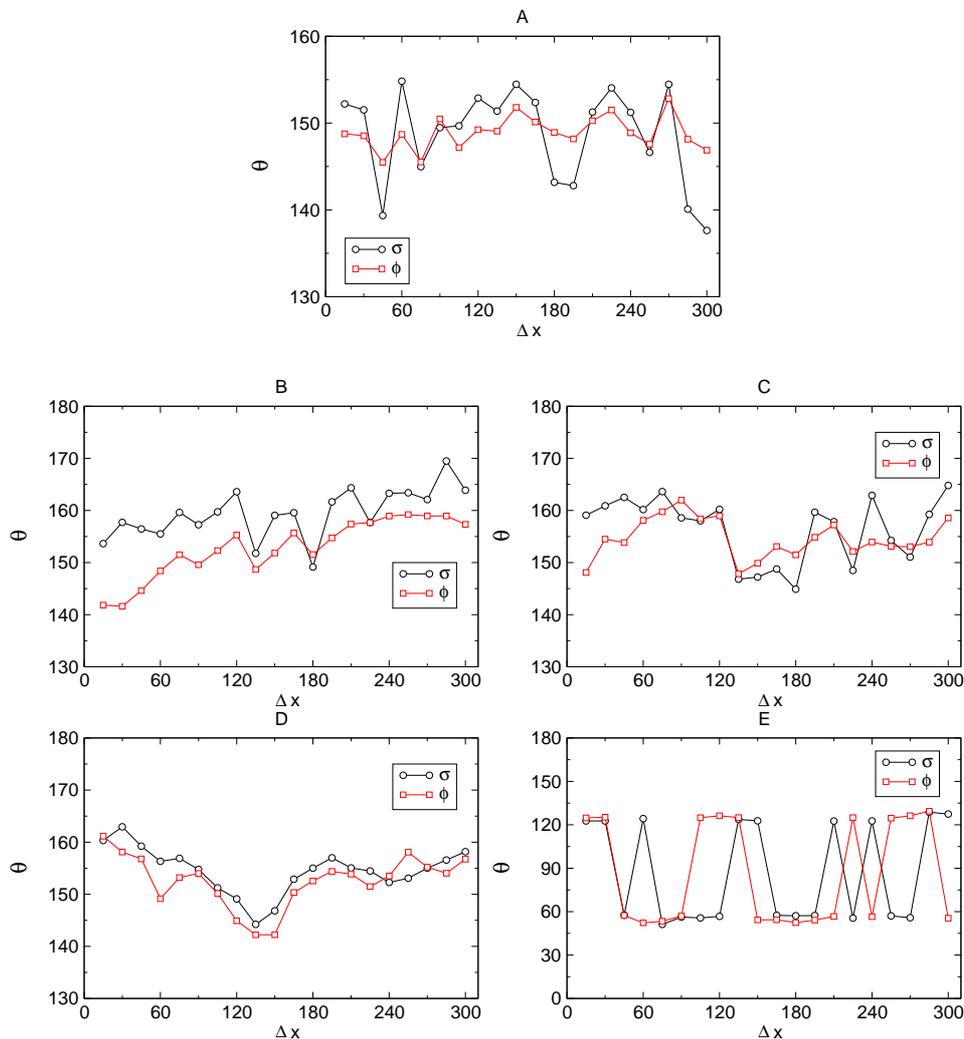


FIG. 7.22 – Evolution de la direction principale des tenseurs de contrainte σ et de fabrique Φ pour les zones A , B , C , D et E de la figure 7.21.

égale à l'une des deux valeurs, la direction principale de la texture est égale à l'autre.

Pour clore cette étude nous regardons l'évolution de la pression à l'intérieur des différentes zones (adimensionnée par $\rho g d$) ainsi que le rapport q/p . Ce dernier reste constant dans la zone E , autour de 0.1 et fluctue faiblement pour les autres zones, autour de la valeur 0.3. Dans les zones A , B et C la pression augmentent faiblement lors du raccourcissement : la valeur de la pression est d'autant plus importante que la zone est proche de la paroi. Ceci s'explique par la butée qui exerce une force sur le fond de l'échantillon plus importante

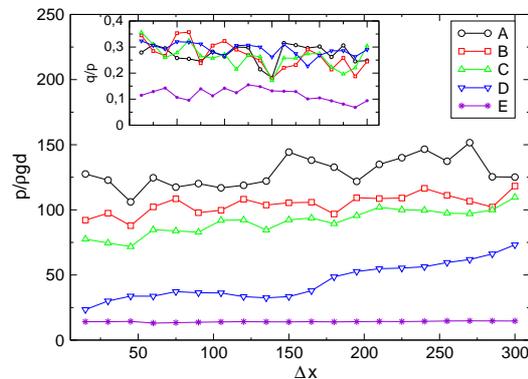


FIG. 7.23 – Evolution de la pression p dans les différentes zones de l'échantillon. En encart évolution du rapport q/p .

près de la paroi. Dans la zone D , la pression est relativement faible dans la première partie de la simulation ($p/\rho gd \approx 40$) et subit une augmentation constante dans la deuxième partie dû à l'arrivée de matière apporté par le mouvement de la paroi. Dans la zone, E $p/\rho gd$ reste constant autour de 20 correspondant au nombre de particules en hauteur. La zone E n'est pas sous l'influence de l'avancée de la butée.

7.4.3 Conclusion

Sur l'ensemble des études de ce chapitre des simulations avec un plus grand nombre de particules ont été réalisées, mais les calculs étant plus longs, moins de paramètres ont pu être testés à ce jour. Le fait de pouvoir simuler des échantillons de tailles conséquentes nous permet d'observer numériquement, des phénomènes proches de ceux observés expérimentalement et dans les structures naturelles. Cette approche n'est pas uniquement visuelle, mais nous permet d'aller regarder l'évolution de grandeurs locales comme le tenseur des contraintes et la texture. Il peut être intéressant de regarder d'autres grandeurs comme la compacité et la coordinence et essayer également de définir une notion de déformation locale pour nos échantillons. Quoi qu'il en soit l'outil de simulation est au service de la communauté géophysicienne qui pourra certainement l'exploiter plus efficacement. Nous pensons cependant que cette première étude de béotien montre les potentiels du logiciel et des post-traitements qui lui ont été associés

Chapitre 8

Aujourd'hui et demain : numérique et granulaire

Notre travail, présenté dans ces quelques pages, s'inscrit dans le cadre de la simulation numérique des milieux granulaires. L'outil numérique est d'un intérêt certain pour l'étude de ces milieux et de nombreuses méthodes sont proposées. Parmi ces méthodes l'approche NSCD dont nous nous servons, a montré à de nombreuses reprises son efficacité par un traitement rigoureux du contact (sans régularisation). Jusqu'à présent cette méthode bénéficiait d'un algorithme de type Gauss-Seidel non linéaire dont la convergence est lente, en particulier lorsque le nombre de particules à modéliser est grand : cette situation se produit lorsque l'on veut observer des phénomènes locaux dans de grandes structures sans connaissance a priori du comportement global du milieu.

Dans le but d'améliorer le temps de calcul, apparaissant comme l'un des principaux handicaps dans les simulations numériques, nous avons développé et mis en place différentes techniques aussi bien séquentielles que parallèles. Présentées dans le chapitre 3, elles nous permettent d'améliorer le temps *CPU* nécessaire aux simulations tout en continuant de bénéficier de la robustesse de la méthode NSCD. Les perturbations numériques engendrées par le traitement parallèle sur l'algorithme NLGS ne modifient pas le comportement mécanique global de nos échantillons, garant de la qualité des solutions. Une étude quantitative nous permet de déterminer, pour un type de simulation donné, la méthode la plus favorable à utiliser en fonction du nombre de processeurs à notre disposition. Différentes architectures de calculateurs ont également été utilisées afin d'évaluer l'influence de celles-ci sur les différentes techniques mises en place.

Outre des développements informatiques, la mise en place d'un solveur de type Gradient Conjugué a été réalisée, solveur auquel le chapitre 4 est dédié. Connue pour être plus rapide que Gauss-Seidel pour des problèmes linéaires (si tant est qu'il soit préconditionné), il s'avère être un bon candidat pour obtenir une convergence plus rapide. Quelques adaptations au schéma général d'un algorithme d'optimisation ont été introduites et jus-

tifiées pour prendre en compte le caractère non associé du contact frottant. Il s'avère que cette propriété ne gêne pas la convergence (excepté pour de forts coefficients de frottement). Cet algorithme possède un caractère global (traitement simultané de l'ensemble des contacts) contrairement à l'algorithme NLGS qui lui présente un caractère local (contact par contact). Ce caractère lui confère quelques avantages dont une convergence rapide soulignée à différentes reprises dans une étude quantitative et qualitative. Nous pointons également les applications pour lesquelles nous pouvons bénéficier pleinement des performances de ce nouvel algorithme. Les produits matrice-vecteur ainsi que les projections pouvant être effectuées composante par composante permettent un traitement parallèle immédiat qui donne des résultats concordants à nos attentes.

Les développements effectués dans un cadre bidimensionnel s'étendent plus ou moins bien aux problèmes tridimensionnels. L'extension du solveur NLGS ainsi que ses différentes implémentations (séquentielles et parallèles) se fait de façon quasi-immédiate en utilisant un algorithme de Newton pour la résolution du problème de contact frottant local. Pour des géométries particulières comme les sphères, une technique plus rapide fournissant des résultats équivalents peut être mise en place. Le passage 2D/3D des techniques d'implémentation ainsi que de la version parallèle ne présente aucune difficulté, mais souligne le problème de l'espace mémoire lors du traitement de gros calculs devenant vite incontournable. Le caractère non polyédrique du cône de Coulomb rend l'extension du CPG plus délicate, en particulier la définition des projections des gradients. Différentes stratégies ont été testées, fournissant des résultats plus ou moins concluants suivant la méthode. Toutefois, la convergence reste meilleure que NLGS mais avec un gain qui est moindre qu'en 2D.

Le choix des deux applications, écoulements granulaires et évolutions quasi-statiques, mettant en jeu un nombre important de particules sur de long processus, a été motivé par une volonté de souligner la diversité du comportement des milieux granulaires, tantôt liquide tantôt solide. Mais malgré leur comportement différents, les deux types d'applications mettent en jeu des milieux denses où la notion de système multi-contact prend tout son sens. Dans une première partie nous avons ainsi pu enrichir les informations sur les écoulements en tambour tournant en mesurant et analysant des données impossibles à obtenir expérimentalement, ceci après avoir vérifié la concordance des profils mesurables expérimentalement et numériquement. Les premiers résultats 3D sont encourageants nous incitant à poursuivre ces investigations. Ces résultats soulignent l'importance de traiter rigoureusement les problèmes de contact locaux pour retrouver un comportement global cohérent. Nous avons pu également entreprendre dans une seconde partie une étude naïve des mouvements tectoniques en analysant différents paramètres et en étudiant deux types de mouvement : les propagations de faille et les raccourcissements. Nos outils semblent permettre une analyse fine et poussée, pouvant se révéler très efficace dans les mains expertes des géophysiciens. Grâce aux développements numériques, les temps de calculs ne se présentent plus comme un handicap insurmontable puisque la combinaison optimisation algorithmique et informatique s'avère relativement efficace (CPG parallèle).

Nous proposons dans cette dernière partie quelques études prospectives entreprises au cours de nos travaux et qu'il nous semble intéressant d'intégrer dans ces quelques pages. Nous présentons en quelques mots les différents contextes, les avancées dans chaque domaine ainsi que les axes qui nous semblent intéressants de développer.

Réseaux faible et fort

Des études numériques sur les milieux granulaires ont permis de mettre en évidence deux réseaux de contact, le réseau fort et le réseau faible, délimités par la force moyenne dans l'échantillon [126]. Ces deux réseaux ont des propriétés différentes. Le réseau fort, essentiellement constitué de contacts adhérents, joue le rôle de "squelette" du milieu, les efforts se propageant essentiellement à travers lui. Le réseau faible est beaucoup plus instable ayant tendance à se comporter comme un fluide coincé dans les cellules formées par le réseau fort. Les contacts du réseau faible sont essentiellement glissants. Dans le chapitre 4, nous avons mis en évidence l'influence du frottement sur la convergence des algorithmes, beaucoup plus forte pour le CPG que pour NLGS. Pour poursuivre cette étude dans le cas d'un réseau bimodal, nous avons tenté d'évaluer les fluctuations dans les différents réseaux en fonction du frottement inter-grains. Les figures 8.1 et 8.2 nous permettent d'observer respectivement l'évolution du nombre de changements de statut au cours des itérations pour les algorithmes NLGS et CPG.

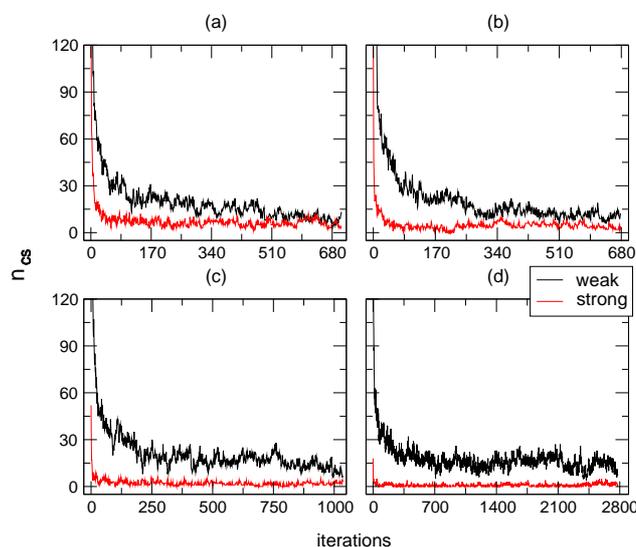


FIG. 8.1 – Changement de statut au cours des itérations NLGS pour différents coefficients de frottement : $\mu = 0.0$ (a), $\mu = 0.2$ (b), $\mu = 0.4$ (c) et $\mu = 0.8$ (d).

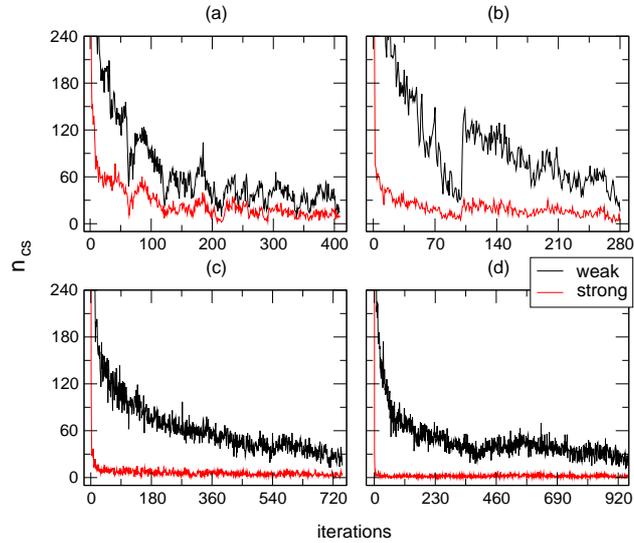


FIG. 8.2 – Changement de statut au cours des itérations CPG pour différents coefficients de frottement : $\mu = 0.0$ (a), $\mu = 0.2$ (b), $\mu = 0.4$ (c) et $\mu = 0.8$ (d).

Le premier constat au regard de ces figures est le faible nombre de changements de statut dans le réseau fort, ceci indépendamment de l'algorithme utilisé. Le nombre de changements reste relativement constant et peu élevé pour l'algorithme NLGS (inférieur à 10). Pour l'algorithme CPG, ce nombre plus important pour de faible valeur - nulle comprise - du frottement, diminue et se stabilise pour des valeurs plus grandes. Dans le réseau faible le comportement est différent. Le nombre de changements de statut est beaucoup plus important, avec une tendance à la décroissance et ceci jusqu'à convergence. Les fluctuations sont beaucoup plus importantes avec CPG que NLGS. Il semble donc que ce soit ces fluctuations sur le réseau faible qui pilote la convergence.

Nous pouvons également regarder en parallèle l'évolution des critères de convergence, des changements de statut dans les différents réseaux et du nombre des différents statuts. La figure 8.3 présente ce parallèle pour les algorithmes NLGS et CPG pour un coefficient de frottement de 0.2.

Les graphes du bas (évolution de n_c) soulignent bien le fait que le réseau fort est essentiellement composé de contacts adhérents. Nous nous apercevons que le réseau faible est composé de contacts adhérents et de contacts glissants à proportion égale. Le phénomène le plus intéressant vient dans l'observation simultanée de l'évolution des critères de convergence et du nombre de changements de statut pour l'algorithme CPG. En effet leur évolution sur le premier tiers est quasi-similaire : une décroissance prononcée sur les deux graphes. Puis un changement brutal dans le réseau faible vient perturber la convergence, faisant apparaître un saut dans l'évolution des critères. Après cette crise les courbes décroissent à

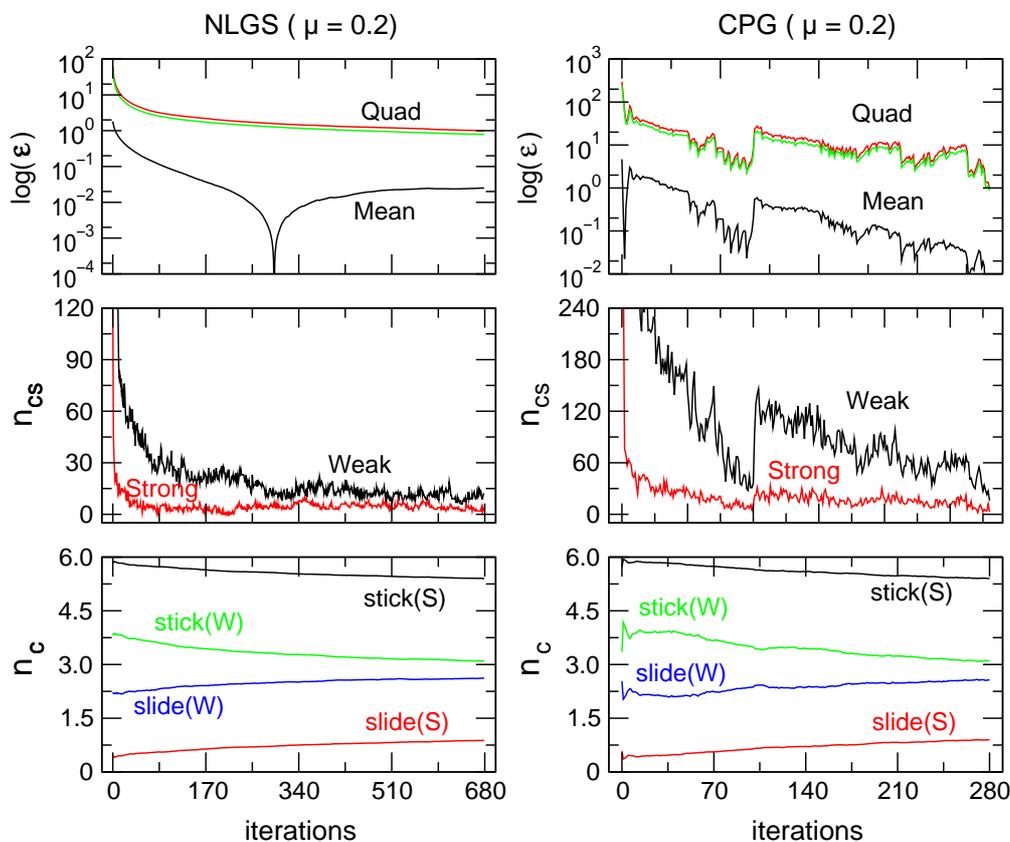


FIG. 8.3 – Parallèle entre les critères de convergence (Mean pour e_m et Quad pour e_Q), les changements de statut dans les différents réseaux (n_{cs}) et le nombre des différents statuts ($n_c \times 1000$).

nouveau jusqu'à convergence. Ainsi le phénomène se manifestant sur la figure 8.3 semble souligner l'importance du réseau faible dans la convergence du CPG. Par quel moyen peut-on alors contrôler ces variations? Plusieurs axes sont envisageables comme par exemple un couplage CPG/NLGS puisque le second semble moins sensible aux variations du réseau faible, ou alors la mise en place d'un algorithme bimodal en résolvant successivement le problème sur le réseau fort puis sur le réseau faible .

Fils et granulat : le Texsol

Le "Texsol" [76, 94], mélange de sable et de fils synthétiques, est un matériau composite utilisé pour construire des ouvrages de soutènement. Dans le début des années 90, de nombreuses recherches théoriques ont été réalisées afin de déterminer le comportement mécanique d'un tel matériau. Avec les progrès des méthodes numériques dédiées

aux éléments discrets, ils nous semblent intéressant de compléter les travaux déjà réalisés par une étude numérique. Grâce à la méthode utilisée, implémentée dans la plateforme logiciel *LMGC90* [45], nous bénéficions d'un outil performant pour attaquer le problème véritablement tridimensionnel que représente le "Texsol".

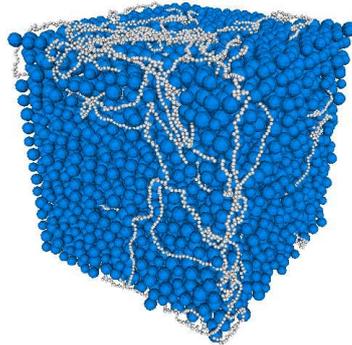


FIG. 8.4 – Echantillon de Texsol.

Le "Texsol" comportant une majeure partie de sable, une approche par éléments discrets nous semble intéressante. Un point délicat est le grand nombre de particules à traiter. Pour donner un ordre de grandeur, pour du sable de $200 \mu m$ de diamètre, il faut plus de 200 000 particules pour modéliser $1 cm^3$. On peut voir sur la figure 8.4 un volume élémentaire (représentatif?) de 20 000 particules (seulement!?). Outre le grand nombre de particules, le point le plus délicat reste la modélisation des fils synthétiques.

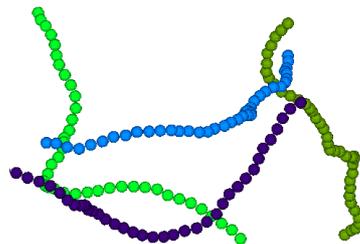


FIG. 8.5 – Des fils vus comme des chaînes de billes.

Une étude préliminaire sur des échantillons 2D a été réalisée pour essayer de déterminer la modélisation la plus probante pour le fil [111]. Les différentes modélisations consistent à voir le fil comme une chaîne de billes reliées par des lois d'interaction à distance (figure

8.5). Des lois élastiques pouvant travailler en traction et/ou compression ainsi que des lois bilatérales ont été testées. Ces dernières apparaissant comme moins sensibles que les lois élastiques qui ont tendance à propager une onde et à rendre instable notre système. La figure 8.6 illustre le comportement des itérations pour un essai biaxial piloté en force effectué sur trois types d'échantillon : le *Texsol 1* utilisant une loi bilatérale pour modéliser le fil, le *Texsol 2* utilisant une loi bilatérale modifiée permettant le rapprochement des billes du fils et le *Sable* pour échantillon témoin.

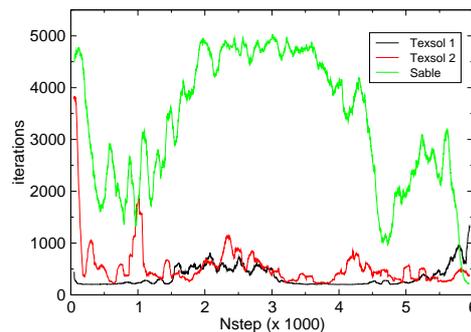


FIG. 8.6 – Evolution des itérations lors d'un essai bi-axial pour trois types d'échantillon.

Il est curieux de voir que la convergence se passe mieux pour les échantillons de Texsol 1 et 2 que pour le sable seul. Ce phénomène est relié au fait que pour le sable, qui subit une déformation beaucoup plus grande que le Texsol, les réarrangements sont beaucoup plus nombreux et donc la stabilisation de l'échantillon nécessite un plus grand nombre d'itérations. Les courbes de la figure 8.7 soulignent cette différence de comportement pour les différents échantillons.

Les fils fournissant une tenue mécanique plus importante, réduisent le nombre de réarrangements à l'intérieur de l'échantillon entraînant également une stabilité dans le nombre d'itérations moyen. La crise au niveau des itérations coïncide avec la diminution de la compacité et l'augmentation de l'énergie cinétique. Le travail préliminaire effectué dans [111] est donc à poursuivre et à étendre au 3D, dimension permettant de prendre correctement en compte l'enchevêtrement des fils. Les techniques informatiques et algorithmiques développées nous permettront de garder des temps de calcul raisonnables et pouvoir ainsi effectuer des études paramétriques complètes.

D'autres perspectives ...

Nous venons de présenter les principaux axes que nous souhaitons explorer par la suite. Ce ne sont pas les seules perspectives que nous avons, aussi bien d'un côté numérique que d'un côté application. Dans le désordre on peut citer :

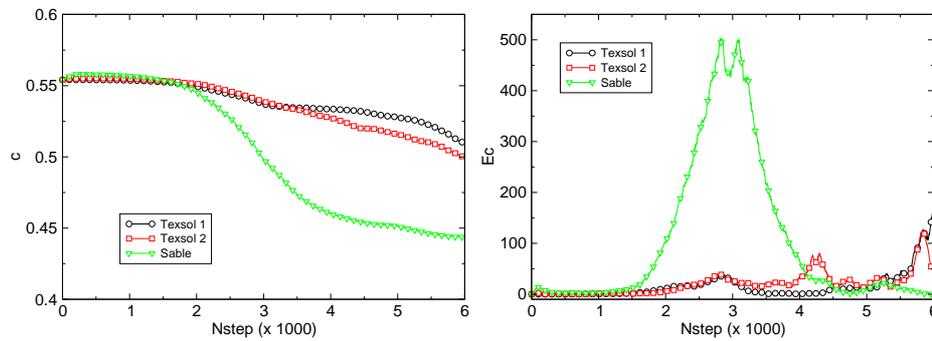


FIG. 8.7 – Evolution de la compacité (gauche) et de l'énergie cinétique (droite) lors de l'essai bi-axial effectué 8.6.

- la parallélisation des autres parties du code, en particulier la détection des contacts, qui doit être abordée pour continuer de bénéficier des efforts réalisés sur le solveur. On peut bénéficier pour ce travail de méthodes générales pré-existantes ou développer des méthodes plus particulières à nos types d'applications.
- une combinaison d'algorithmes pour résoudre nos problèmes de contact frottant. On peut concevoir par exemple un couplage NLGS et CPG où chaque algorithme opère sur l'une des composantes de l'impulsion locale.
- le développement de préconditionneurs plus performants. Ceux-ci peuvent par exemple prendre en compte la taille des particules, leur géométrie ou encore l'appartenance du contact à l'un des deux réseaux, si l'on veut que le préconditionneur prenne en compte l'information mécanique du système.
- l'extension du CPG à des lois plus complexes (cohésion, liaisons bilatérales). Ceci demande une réflexion sur la formulation générale du problème et ainsi redéfinir les différentes projections.
- l'approfondissement des applications mises en place avec notamment une exploitation 3D. Des confrontations directes avec les expériences peuvent être entreprises dans les deux cas et des études paramétriques poussées peuvent aussi être effectuées.

Bien entendu on peut trouver d'autres types d'applications en mécanique des milieux granulaires, mais on peut également considérer l'exploitation de notre algorithme sur d'autres problèmes de contact frottant.

Bibliographie

- [1] 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, Edimburgh, September, 6-8 2000. Civil-Comp Press.
- [2] K. Ach and P. Alart. Numerical simulation of a multi-jointed structure. In *Philosophical transactions of the Royal Society of London*, volume A 359, pages 2557–2573, 2001.
- [3] 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) :203–222, 1993.
- [4] P. Alart. Méthode de Newton généralisée en mécanique du contact. *J. Math. Pures Appl.*, 76 :83–108, 1997.
- [5] P. Alart, M. Barboteu, and M. Renouf. Parallel computational strategies for multi-contact problems : Applications to cellular and granular media. *Int. J. Mult. Scales Comput. Engrg.*, 1(4) :419–430, 2003.
- [6] P. Alart and A. Curnier. A mixed formulation for frictionnal contact problems prone to Newton like solution methods. *Comp. Meth. Appl. Mech. Engrg.*, 92 :353–375, 1991.
- [7] P. Alart and F Lebon. Multigrid method applied to mixed formulation contact problems. In *Contact Mechanics 93*, Southampton, 1993.
- [8] P. Alart, F. Lebon, F. Quittau, and K. Rey. Frictional contact problem in elastostatics : revisiting the uniqueness condition. In M. Jean M. Raous and J.-J. Moreau, editors, *Contact Mechanics*, pages 63–70. Plenum Press-New York, 1995. ISBN 0-306-45065-8.
- [9] M.P. Allen and D.J. Tildesley. *Computer simulation of liquids*. Oxford University Press, 1987.
- [10] B. Andreotti, A. Daerr, and S; Douady. On scaling laws in granular flows down a rough plane. to appear in *Phys. Fluids*.
- [11] A. Aradian, E. Raphael, and P. G. de Gennes. Surface flows of granular materials : a short introduction to some recent models. *C. R. Physique*, 3 :187, 2002.

- [12] I. S. Aranson and L.S. Tsimring. Continuum description of avalanche in granular media. *Phys. Rev. E*, 64 :020301, 2001.
- [13] O. Axelsson. *Iterative Solution Methods*. Cambridge University Press, New York, 1994.
- [14] O. Axelsson and I. Kaporin. Error norm estimation and stopping criteria in preconditioned conjugate gradient iterations. *Numer. Linear Algebra Appl.*, 8 :265–286, 2001.
- [15] D. Barraff. Fast contact force computation for nonpenetrating rigid bodies. In *SIGGRAPH 94*, pages 23–34, Orlando (USA), July 24-29, 1994.
- [16] L.S. Blackford, J. Choi, A. Cleary, E.D. Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and Whaley R.C. *ScaLAPACK User's Guide*. SIAM, Philadelphia,PA, 1997.
- [17] D. Bonamy. *Phénomènes collectifs dans les matériaux granulaires : écoulements de surface et réarrangements internes dans des empilements modèles*. PhD thesis, Université Paris XI, Orsay (France), 2001. Service de physique de l'état condensé-CEA Saclay.
- [18] D. Bonamy, F. Daviaud, and L. Laurent. Experimental study of granular surface flows via a fast camera : a continuous description. *Phys. Fluids*, 14(5) :1666–1673, 2002.
- [19] J.-F. Bonnans, J.-C. Gilbert, C. Lemaréchal, and C.A. Sugastizabal. *Numerical Optimization - Theoretical and Practical Aspect*. Springer, 2003.
- [20] P.A. Boucard and L. Champaney. A suitable computational strategy for the parametric analysis of problems with multiple contact. *Int. J. Numer. Meth. Engng.*, 57 :1259–1281, 2003-12-23.
- [21] J.P. Bouchaud, M. E. Cates, J. R. Prakash, and S. F. Edwards. A model for the dynamics of sandpile surface. *J. Phys. France I*, 4 :1383, 1994.
- [22] P. Breitenkopf and M. Jean. Modélisation parallèle des matériaux granulaires. In *4ème colloque national en calcul des structures*, Giens, 1999.
- [23] B. Brogliato, AA. ten Dam, L. Paoli, F. Génot, and M. Abadie. Numerical simulation of finite dimensional multibody nonsmooth mechanical systems. *Appl. Mech. Rev.*, 55(2) :107–149, 2002.
- [24] H.C.J. Bruneel and I. De Rycke. Quicktrace : a fast algorithm to detect contact. *Int. J. Numer. Meth. Engng.*, 54 :299–316, 2002.
- [25] D.R. Burbidge and J. Braun. Numerical models of the evolution of accretionary wedges and fold-and-thrust belts using the distinct-element method. *Geophys. J. Int.*, 148 :542–561, 2002.
- [26] B. Cambou and M. Jean. *Micromécanique des matériaux granulaires*. Hermès Science-Paris, 2001.
- [27] D. Cambou and Ph. Dubujet. Difficulties and limitation of statistical homogenization in granular materials. In Vermeer et al., editor, *Continuous and Discontinuous*

- Modeling of Cohesive Frictional Materials.*, volume 568, pages 205–214. Springer, 2001.
- [28] N. Cardozo, K. Bhalla, A.T. Zehnder, and R.W. Allmendinger. Mechanical models of fault propagation folds and comparison to the trishear kinematic model. *J. Struc. Geol.*, 25 :1–18, 2003.
- [29] P. Chabrand, F. Dubois, and M. Raous. Various numerical methods for solving unilateral contact problems with friction. *Math. Comput. Modell.*, 28(4-8) :97–108, 1999.
- [30] C. Cholet, G. Saussine, P.E. Gautier, F. Dubois, C. Bohatier, and K. Combe, G. and Sab. Application of discrete element methods to the modelling of ballasted track. In *World Congress on Railway Research (WCRR)*, 2003.
- [31] N. Chrisochoides and D. Nave. Parallel delaunay mesh generation kernel. *Int. J. Numer. Meth. Engng*, 58 :161–176, 2003.
- [32] P.W. Christensen. A semi-smooth Newton method for elasto-plastic contact problems. *Int. J. Solids Struct.*, 39 :2323–2341, 2002.
- [33] J. Christoffersen, M.M. Mehrabadi, and S. Nemat-Nasser. A micro-mechanical description of granular material behavior. *J. Appl. Mech.*, 48(ASME) :339–344, 1981.
- [34] P.R. Cobbold and L. Castro. Fluid pressure and effective stress in sandbox models. *Tectonophysics*, 301 :1–19, 1999.
- [35] P.R. Cobbold, S. Durand, and R. Mourgues. Sandbox modelling of thrust wedges with fluid-assisted detachments. *Tectonophysics*, 334 :245–258, 2001.
- [36] M. Cocu. Existence of solutions of Signorini problems with friction. *Int. J. Engng. Sc.*, 22(5) :567–575, 1984.
- [37] G. Combe and J.-N. Roux. Trajectoires quasi-statiques d’assemblages granulaires. In *Colloque physique et mécanique des matériaux granulaires*, pages 157–162. Presse du Laboratoire central des ponts et chaussées, 2000.
- [38] 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.
- [39] P. A. Cundall and O. D. L. Strack. A discrete numerical model for granular assemblies. *Géotechnique*, 29(1) :47–65, 1979.
- [40] G. de Saxcé and Z.Q. Feng. New inequation and functional for contact with friction. *J. Mech. of Struct. and Mach.*, 19 :301–325, 1991.
- [41] G. Dilintas, P. Laurent-Gengoux, and D. Trystram. A conjugate projected gradient method with preconditioning for unilateral contact problems. *Comp. Struct.*, 29(4) :675–680, 1988.
- [42] C. Dimate, L. Rivera, A. Taboada, B. Delouis, A. Osorio, E. Jimenez, A. Fuenzalida, Cisternas A., and I. Gomez. The 19 january 1995 tauramena (colombia) earthquake : Geometry and stress regime. *Tectonophysics*, 363 :159–180, 2003.

- [43] S. Douady, B. Andreotti, and A. Daerr. On granular surface flow equations. *Eur. Phys. J. B*, 11 :131–142, 1999.
- [44] A. Drescher and G. De Josselin de Jong. Photoelastic verification of a mechanical model for the flow of a granular materials. *J. Mech. Phys. Solids*, 20 :337–351, 1972.
- [45] F. Dubois and 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*, volume 1, pages 111–118. CSMA-AFM-LMS, 2003.
- [46] G. Duvaut. Loi de frottement non locale. *J. Méc. Th. et Appl.*, numéro spécial :73–78, 1982.
- [47] E. Falcon. *Comportement dynamique associés au contact de Hertz : processus collectifs de collision et propagation d’ondes solitaires dans les milieux granulaires*. PhD thesis, ENS - Université Claude Bernard Lyon I, 1997.
- [48] Y.T. Feng, K. Han, and D. R. J. Owen. Filling domains with disks : An advancing front approach. *Int. J. Num. Meth. Engng.*, 56(5) :699–713, 2003.
- [49] E. Finch, S. Hardy, and R. Gawthorpe. Discrete element modelling of contractional fault-propagation folding above rigid basement fault blocks. *J. Struct. Geol.*, 25 :515–528, 2003.
- [50] J. Fortin and P. Coorevits. Selecting contact particles in dynamics granular mechanics systems. In *ACOMEN*, Liège (Belgium), May 28-31 2002. ISBN 2-930332-39-X.
- [51] J. Fortin and G. de Saxé. Modélisation numérique des milieux granulaires par l’approche du bipotentiel. *C. R. A. Sci. Paris, Ser. IIB*, 327 :721–724, 1999.
- [52] J. Fortin and G. de Saxé. Simulation numérique par un modèle discret de l’écoulement et du stockage des matériaux granulaires. In *14eme Congrès Français de Mécanique*, Toulouse, 30 aout-3 septembre 1999. CFM.
- [53] J. Fortin, M. Hjjaj, and G. de Saxcé. An improved discrete element method based on a variational formulation of thr frictional contact law. *Comp. Geotech.*, 29 :609–640, 2002.
- [54] G. Félix. *Écoulements de Matériaux Granulaires en tambour tournant*. PhD thesis, Institut National Polytechnique de Lorraine, Nancy (France), 2001.
- [55] C. Glocker and F. Pfeiffer. *Multibody dynamics with unilateral contacts*. John Wiley and Sons, 1996.
- [56] E. Gondet and P.-F. Lavalée. *Cours OpenMP*. IDRIS, Orsay, septembre 2000.
- [57] S. Hardy. A velocity description of constant-thickness fault-propagation folding. *J. Struct. Geol.*, 19(6) :893–896, 1997.
- [58] S. Hardy and K. McClay. Kinematic modelling of extensional fault-propagation folding. *J. Struct. Geol.*, 21 :695–702, 1999.
- [59] J. Haslinger, R. Kucera, and Z. Dostal. An algorithm for the numerical realization of 3d contact problems with coulomb friction. *J. Comp. Appl. Math.*, 164-165 :387–408, 2004.

- [60] M.W. Heinstein and T.A. Laursen. An algorithm for the matrix-free solution of quasistatic frictional contact problems. *Int. J Numer. Meth. Engrg.*, 44 :1205–1226, 1999.
- [61] J. Herrmann, H., S. Luding, and R. Caferio. Dynamics of granular systems. *Phys. A*, 295 :93–100, 2001.
- [62] J. Herrmann, H. and M. Müller. Simulations of granular media. *World Scientific*, 1999.
- [63] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and Minimization I*. Springer-Berlin, 1993.
- [64] D. Howell and R. P. Behringer. Stress fluctuations in a 2d granular couette experiment : A continuous transition. *Phys. Rev. Lett.*, 82 :5241–5244, 1999.
- [65] M.K. Hubbert. Mechanical basis for certain familiar geological structures. *Bull. Geol. Soc. America*, 62 :335–372, 1951.
- [66] J.T.R. Hugues, I. Levit, and J.M. Winget. An element by element solution algorithm for problems of structural and solid mechanics. *Comput. Methods Appl. Mech. Engrg.*, 36 :241–254, 1983.
- [67] H.M. Jaeger and S.R. Nagel. Granular solids, liquids and gases. *Rev. Mod. Phys.*, 68(4) :1259–1273, 1996.
- [68] N. Jain, M. Ottino, J., and M. Lueptow, R. An experimental study of the flowing granular layer in a rotating tumbler. *Phys. Fluids*, 14(2) :572–582, 2002.
- [69] M. Jean. Computational error or accuracy in numerical simulation of divided materials. note interne.
- [70] M. Jean. The non-smooth contact dynamics method. *Comp. Meth. Appl. Mech. Engrg.*, 177 :235–257, 1999.
- [71] M. Jean and J.J. Moreau. Unilaterality and dry friction in the dynamics of rigid bodies collection. In A. Curnier, editor, *Contact Mechanics International Symposium*, pages 31–48. Presses Polytechniques et Universitaires Romanes, 1992.
- [72] K.L. Johnson, K. Kendall, and Roberts A.D. Surface energy and the contact of elastic solids. *Proc. R. Soc. London. A.*, 324 :301–313, 1971.
- [73] K.M. Johnson and A.M. Johnson. Forced-fold model of trishear-like folds. 2000.
- [74] F. Jourdan, P. Alart, and M. Jean. A Gauss-Seidel like algorithm to solve frictional contact problem. *Comp. Meth. Appl. Mech. Engrg.*, 155 :31–47, 1998.
- [75] T. Kawai. Some considerations on the finite element method. *Int. J. Numer. Meth. Engrg.*, 16 :81–120, 1980.
- [76] M. Khay and J.-P.. Gigan. Mur de soutènement expérimentaux en Texsol : réalisation et interprétation des essais de chargement à la rupture. *Bull. liaison Labo P. et Ch.*, 173 :97–108, 1991.
- [77] Y. Kishino. Disk model analysis of granular media. *Micromechanics of Granular Materials*, pages 143–152, 1988. (eds. M. Satake and J.T. Jenkins), Elsevier.

- [78] Y. Kishino. *Mechanics of granular materials : an introduction*, chapter 3, pages 147–159. Balkema, 1998.
- [79] Y. Kishino, editor. *Powders and Grains*, Sendai-Japan, 2001. Balkema.
- [80] Y. Kishino. Granular flow simulation by granular element method. In *Slow Dynamics in Complex Systems*, November 2003. Sendai, Japan.
- [81] Y. Kishino. The incremental nonlinearity observed in numerical tests of granular media. In *15th ASCE Engineering Mechanics Conference*, New York, Columbia University, June 2-5 2002.
- [82] Y. Kishino, H. Akaizawa, and K. Kaneko. On the plastic flow of granular materials. In Y. Kishino, editor, *Powder and grains*, pages 199–202, 2001.
- [83] A. Klarbring. A mathematical programming approach to three-dimensional contact problems with friction. *Comput. Methods Appl. Mech. Engrg.*, 58 :175–200, 1986.
- [84] T.S. Komatsu, S. Inagasaki, N. Nakagawa, and S. Nasuno. Creep motion in a granular pile exhibiting steady surface flow. *Phys. Rev. Lett.*, 86 :1757–1760, 2001.
- [85] H. Koyi. Mode of internal deformation in sand wedges. *J. Struct. Geol.*, 17(2) :293–300, 1995.
- [86] N. Kukowski, S.E. Lallemand, J. Malavieille, M.-A. Gutscher, and T.J. Reston. Mechanical decoupling and basal duplex formation observed in sandbox experiments with application to the Western Mediterranean Ridge accretionary complex. *Mar. Geol.*, 186 :29–42, 2002.
- [87] P. Ladevèze, A. Nouy, and O. Loiseau. A multiscale computational approach for contact problems. *Comput. Meth. Appl. Mech. Engrg.*, 43 :4869–4891, 2002.
- [88] J. Lanier and M. Jean. Experiments and numerical simulations with 2D disks assembly. *Powd. Tech.*, 109 :206–221, 2000.
- [89] J. Lanier and F. Radjai. *Analyse expérimentale et numérique des variables locales dans les matériaux granulaires.*, chapter 2, pages ...–... Hermès-Paris, 2001.
- [90] T.A. Laursen and J.C. Simo. Algorithmic symmetrization of Coulomb frictional problems using augmented Lagrangians. *Comput. Meth. Appl. Mech. Engrg.*, 108 :133–146, 1993.
- [91] C. Le Saux, F. Cevaer, and R. Motro. Dynamique multi-contact d’un système de solides tridimensionnels rigides. *C. R. Mécanique*, 331 :277–282, 2003.
- [92] C. Le Saux, F. Cevaer, and R. Motro. Contribution to 3d impact problems : collisions between two slender stell bars. *C. R. Mécanique*, 332 :17–22, 2004.
- [93] P. Le Tallec. Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 1 :121–220, 1964.
- [94] E. Leflaive, M. Khay, and J.-C. Blivet. Un nouveau matériau : le Texsol. *Bull. liaison Labo P. et Ch.*, 125 :105–114, 1983.
- [95] J. Lohrmann, N. Kukowski, J. Adam, and O. Oncken. The impact of analogue material properties on the geometry, kinematics, and dynamics of convergent sand wedges. *Journal of Structural Geology*, 25 :1691–1711, 2003.

- [96] S. Luding, E. Clément, A. Blumen, J. Rajchenback, and J. Duran. Onset of convection in molecular dynamics simulations of grains. *Phys. Rev. E*, 50(3) :1762–1765, 1994.
- [97] C. K. K. Lun. Kinetic theory for granular flow of dense slightly inelastic, slightly rough spheres. *J. Fluid Mech.*, 233 :539–559, 1991.
- [98] H.-O. May. The conjugate gradient method for unilateral problems. *Comp. Struct.*, 12(4) :595–598, 1986.
- [99] F. Mehrez. *Modélisation du Contact-Frottement dans la simulation numérique de l'emboutissage des toles*. PhD thesis, Université Pierre et Marie Curie, Paris 6, juin 1991.
- [100] J. Mercier and P. Vergely. *Tectonique*. Dunod, Paris, 1992.
- [101] G.D.R. Midi. On dense granular flows. accepted in Euro. Phys. J. E.
- [102] B. Miller, C. O'Hern, and R. P. Behringer. Stress fluctuations for continuously sheared granular materials. *Phys. Rev. Lett.*, 77 :3110–3113, 1996.
- [103] J.-J. Moreau. Une formulation du contact à frottement sec ; application au calcul numérique. *Comptes Rendus Acad. Sci. Paris, Sér.II*, 302 :779–801, 1986.
- [104] J.-J. Moreau. New computation methods in granular dynamics. In Thornton, editor, *Powder & Grains*, pages 227–232. Balkema Press, 1993.
- [105] J.-J. Moreau. Some numerical methods in multibody dynamics : application to granular materials. *Eur. J. Mech. A Solids*, 13(4-suppl.) :93–114, 1994.
- [106] J.-J. Moreau. Numerical investigation of shear zones in granular materials. In D. eds. Grassberger, P. & Wolf, editor, *Proc. HLRZ-Workshop on friction, Arching, Contact Dynamics*, pages 233–247. World Scientific, Singapore, 1997.
- [107] 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, 1998.
- [108] J.-J. Moreau. Numerical aspects of sweeping process. *Comp. Meth. Appl. Mech. Engrg*, 177 :329–349, 1999.
- [109] J.-J. Moreau. Contact et frottement en dynamique des systèmes de corps rigides. *Rev. Eur. Elem. Finis*, 9 :9–28, 2000.
- [110] J.-J. Moreau. Indétermination liée au frottement sec dans le calcul des granulats. In *Actes du sixième colloque national en calcul des structures*, volume 3, pages 465–472. CSMA-AFM-LMS, 2003.
- [111] O. Mouraille. Efforts à distance : vers une simulation numérique du teksol. Master's thesis, Université Montpellier II, 2004.
- [112] D.M. Mueth, H.M. Jaeger, and Nagel S.R. Force distribution in a granular medium. *Phys. Rev. E*, 57(3), 1998.
- [113] J. Necas, J. Jarusek, and J. Haslinger. On the solution of the variational inequality to the Signorini problem with small friction. *Bolletino U. M. I.*, 5(17.B) :796–811, 1980.

- [114] S. Nemat-Nasser and J. Zhang. Constitutive relations for cohesionless frictional granular materials. *Int. J. Plast.*, 18 :531–547, 2002.
- [115] C. Nougier. *Simulation des interactions outil-sol : application aux outils de traitement des sols*. PhD thesis, Département de Modélisation Physique et Numérique de l’Institut de Physique du Globe de Paris, 1999.
- [116] C. Nougier-Lehon, P. Dubujet, and B. Cambou. Analysis of granular material behaviour from two kinds of numerical modelling. In *15th ASCE Engineering Mechanics Conference*, Columbia University, New York, NY, June 2002.
- [117] M. Oda and K. Iwashita, editors. *Mechanics of granular materials*. Balkema., 1998.
- [118] C.S. O’Hern, S.A. Langer, A.J. Liu, and S.R. Nagel. Force distribution near jamming and glass transitions. *Phys. Rev. Lett.*, 86(1), 2001.
- [119] J.-S. Pang. Newton’s method for B-differentiable equations. *Math. Op. Res.*, 15 :311–341, 1990.
- [120] 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.
- [121] K.S. Persson and D. Sokoutis. Analogue models of orogenic wedges controlled by erosion. *Tectonophysics.*, 356 :323–336, 2002.
- [122] O. Pouliquen. Scaling laws in granular flows down rough inclined planes. *Phys. Fluids*, 11(8) :542–548, 1999.
- [123] I. Preechawuttipong. *Modélisation du comportement mécanique de matériaux granulaires cohésifs*. PhD thesis, LMGC - Université Montpellier II, 2002.
- [124] F. Radjai. *Dynamique des rotations et frottement collectif dans les systèmes granulaires*. PhD thesis, Université PARIS XI ORSAY, 1995.
- [125] F. Radjai, M. Jean, J.-J. Moreau, and S. Roux. Force distributions in dense two-dimensional granular systems. *Phys. Rev. Lett.*, 77(2) :264–277, 1996.
- [126] F. Radjai, D. E. Wolf, M. Jean, and J.-J. Moreau. Bimodal character of stress transmission in granular packings. *Phys. Rev. Lett.*, 80(1) :61–64, 1998.
- [127] J. Rajchenbach. Granular flows. *Adv. Phys.*, 49(2) :229–256, 2000.
- [128] M. Raous and S. Barbarin. Conjugate gradient for frictional contact. In Edt. A. Curnier, editor, *Proc. Contact Mechanics Int. Symp.*, pages 423–432. PPUR, 1992.
- [129] M. Renouf. Méthode de dynamique du contact et calcul parallèle. Master’s thesis, Université Montpellier II, 2001.
- [130] M. Renouf, Dubois F., 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 :375–382, 2004.
- [131] M. Renouf and Y. Kishino. A short comparison between the granular element method and the non smooth contact dynamics method. *JSPS summer program*, pages 136–142, 2003.

- [132] G.H. Ristow. Simulating granular flow with molecular dynamics. *J. Phys. I France*, 2 :649–662, 1992.
- [133] G.H. Ristow. Dynamics of granular materials in a rotating drum. *Europhys. Lett.*, 34(4) :263–268, 1996.
- [134] J.B. Rosen. The gradient projection method for nonlinear programming. part 1 - linear constraints. *J. Soc. Indust. Appl. Math.*, 8(1) :181–217, March 1960.
- [135] J.B. Rosen. The gradient projection method for nonlinear programming. part 2 - non linear constraints. *J. Soc. Indust. Appl. Math.*, 9(4) :514–532, Dec. 1961.
- [136] L. Rothenburg and A.P.S. Selvadurai. A micromechanical definition of the cauchy stress tensor for particulate media. In Selvadurai (ed.), editor, *Procs. Int. Symp. on the Mechanical Behavior of Structural Media*, volume Ottawa, Part B, pages 469–486, 1981.
- [137] W. Sassi and J.L. Faure. Role of faults and layer interfaces on the spatial variation of stress regimes in basins : inferences from numerical modelling. *Tectonophysics*, 266 :101–119, 1997.
- [138] M. Satake. *Mechanics of granular materials : an introduction*, chapter 1, pages 1–85. Balkema, 1998.
- [139] G. Saussine. *Contribution à la modélisation de granulats tridimensionnels : application au ballast*. PhD thesis, Université Montpellier II, Sciences et Technologie du Languedoc, 2004.
- [140] G. Saussine, C. Cholet, P.E. Gautier, F. Dubois, C. Bohatier, and J.J. Moreau. Modelling ballast under cyclic loading using discret element method. In *Proc. Cyclic Behaviour of Soils and Liquefaction Phenomena*, April 2004.
- [141] S. B. Savage. Analysis of slow high concentration flows of granular materials. *J. Fluid Mech.*, 377 :1–26, 1998.
- [142] J. Schäfer and D.E. Wolf. Bistability in simulated granular flow along corrugated walls. *Phys. Rev. E*, 51(6) :6154–6157, 1995.
- [143] L. E. Silbert, D. Ertas, G. S. Grest, T. C. Halsey, D. Levine, and S. J. Plimpton. Granular flow down an inclined plane : Bagnold scaling and rheology. *Phys. Rev. E*, 64 :051302(14), 2001.
- [144] L. Staron. *Étude Numérique des Mécanismes de Déstabilisation des Pentes Granulaires*. PhD thesis, Département de Modélisation Physique et Numérique de l’Institut de Physique du Globe de Paris, 2002.
- [145] N. Taberlet, P. Richard, A. Valance, W. Losert, and J.-M. Pasini. Superstable granular heap in a thin channel. *Phys. Rev. Lett.*, 91(26) :264301, 2003.
- [146] S. Tetsuya. Annual report of the earth simulator center. Technical Report ISSN 1348-5830, Japan Marine Science and Technology Center, June 2003. <http://www.es.jamstec.go.jp/esc/images/annualreport2002/index.htm>.
- [147] H. Troadec. *Texture locale et plasticité des matériaux granulaires*. PhD thesis, Université Montpellier II, 2003.

-
- [148] H. Troadec, F. Radjai, S. Roux, and J.C. Charmet. Model for granular texture with steric exclusion. *Phys. Rev. E.*, 66 :041305(4), 2002.
- [149] P. Vialon, M. Ruhland, and J. Grolier. *Eléments de tectonique analytique*. Masson, Paris, 1976.
- [150] O.R. Walton. Numerical simulation of inclined chute flows of monodisperse, inelastic frictional spheres. *Mech. Mat.*, 16 :239–247, 1993.
- [151] R. Walton, O. and L. Braun, R. Simulation of rotary-drum and repose tests for frictional spheres and rigid sphere clusters. Technical report, workshop on Flow of Particules and Fluids, Ithaca, N.Y., Sept. 29 - Oct. 1 1993.
- [152] J. Weber. Recherches concernant les contraintes intergranulaires dans les milieux pulvérulents. *Bull. liaison P. et Ch.*, 20 :3–1 to 3–20, jul.-aug. 1966.
- [153] D.E. Womble, S.S. Dosanjh, B. Hendrickson, Heroux M.A., Plimpton S.J., Tomkins J.L., and D.S. Greenberg. Massively parallel computing : A sandia perspective. *Parallel Computing*, 25 :1853–1876, 1999.
- [154] D. M. Wood. *Soil Behaviour and Critical State Soil Mechanics*. Cambridge University Press, 1990.
- [155] P. Wriggers. Finite element algorithms for contact problems. *Arch. Comput. Meth. Engrg.*, 2 :1–49, 1995.
- [156] K. Yamane, M. Nakagawa, S. A. Altobelli, T. Tanaka, and Y. Tsuji. Steady particulate flows in a horizontal rotating cylinder. *Phys. Fluids*, 10 :1419–1427, 1998.
- [157] R. Yemmas. *Simulations numériques des matériaux granulaires*. PhD thesis, Université Montpellier II, 1993.
- [158] A. Zucchini. A parallel preconditioned conjugate gradient solution method for finite element problems with coarse-fine mesh formulation. *Comput. Struct.*, 78 :781–787, 2000.

Ce travail est dédié à la simulation numérique des milieux granulaires denses dont les interactions sont régies par des lois de contact frottant. L'approche utilisée est la méthode *Non Smooth Contact Dynamics* qui traite le contact sans régularisation numérique et utilise un algorithme de type Gauss-Seidel. Pour diminuer le temps nécessaire aux simulations, et mener des études paramétriques fines, nous avons procédé à plusieurs types d'optimisation numérique portant, dans un premier temps, sur la mise en oeuvre de différents types d'implémentation exploitant ou non le calcul parallèle (optimisation informatique) puis, dans un second temps, sur le développement d'un algorithme de type Gradient Conjugué (optimisation algorithmique). Ces différentes stratégies se révèlent très efficaces nous permettant de simuler plus rapidement des problèmes bi et tridimensionnels. Deux applications pointant sur la diversité de comportement des milieux granulaires (liquide/solide) ont été abordées : une étude numérique sur les écoulements en tambour tournant venant compléter des résultats expérimentaux et une étude de mouvements tectoniques par éléments discrets permettant de porter un nouveau regard sur les expériences analogiques en "boîtes à sable" effectuées en géophysique.

Numerical Optimisation and Parallel Computing dedicated to the simulation of bi and tridimensional particulate media

This work is devoted to the numerical simulation of dense granular media which interact by frictional contact. We use the Non Smooth Contact Dynamics approach which deals rigorously with contact problems and uses a classical Gauss-Seidel like algorithm. To keep reasonable time of computation, we proceed to different kinds of optimisation. In a first part we develop sequential and parallel implementations (informatic optimisation) and in a second one we introduce a new algorithm based on a conjugate gradient method (algorithmic optimisation). These strategies are efficient and allow to simulate faster granular systems with different and sometimes simultaneous behaviours from solid to liquid. Simulations of flows in rotating drums complete the recent experimental investigations. A numerical sandbox modelling is proposed for replacing analogic sandbox experiments used in the study of tectonic movements.

Disciplines : Mécanique

Mots clés : Contact Frottant, Milieux Granulaires numériques denses, 2D/3D, NSCD, Calcul Parallèle, Gradient Conjugué, Tambour tournant, Boite à Sable

L.M.G.C. UMR-5508 Université Montpellier II, Place Eugène Bataillon, 34095 Montpellier.