



HAL
open science

How to overcome indetermination and interpenetration in granular systems via nonsmooth contact dynamics. An exploratory investigation

Pierre Alart

► **To cite this version:**

Pierre Alart. How to overcome indetermination and interpenetration in granular systems via nonsmooth contact dynamics. An exploratory investigation. *Computer Methods in Applied Mechanics and Engineering*, 2014, 270, pp.37-56. 10.1016/j.cma.2013.11.020 . hal-02075883

HAL Id: hal-02075883

<https://hal.science/hal-02075883v1>

Submitted on 21 Mar 2019

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.

How to overcome indetermination and interpenetration in granular systems via nonsmooth contact dynamics. An exploratory investigation

Pierre Alart *

*Laboratoire de Mécanique et Génie Civil (LMGC), University Montpellier 2/CNRS UMR5508 CC048, Place E. Bataillon, F-34095 Montpellier Cedex 5, France
Laboratoire de Micromécanique et d'Intégrité des Structures (MIST), IRSN, CNRS, Université Montpellier 2, F-34095 Montpellier Cedex 5, France*

A B S T R A C T

The present paper is an exploratory investigation to overcome difficulties encountered in the simulation of dense granulates when a NSCD approach is used. Some simple examples permit identification of impulse indetermination as the main cause of the failure or the slowdown of the numerical algorithm. We propose to deal directly with the space of impulses with null resulting momentum. A prediction–correction scheme is detailed and enriched to restrict at best the interpenetration. Finally sophisticated tools are developed to preserve the sparsity of the matrices and consequently to ensure an efficient implementation of an adapted nonsmooth solver.

Keywords:

Nonsmooth contact dynamics
Multicontact systems
Granular mechanics

1. Introduction

The nonsmooth contact dynamics (NSCD) method, developed by Moreau and Jean [1,2], is used to deal with large collections of packed bodies to simulate the behavior of granular materials. The Nonlinear Gauss–Seidel (NLGS) algorithm [3] is the generic solver applied to the NSCD formulation. This combination allows simulation of the behavior of a collection of (especially rigid) bodies involving different and mixed regimes: static, slow dynamics (solid), fast dynamics (fluid). The rotative drum is a typical example where a static regime prevails at the bottom of the drum while avalanches of grains occur on the top [4,5]. The specificity of the NSCD approach is the possibility to solve numerous simultaneous contacts in one time step; we then speak about multicontact problems. Then a large time increment may be used contrary to the Molecular Dynamics approach [6] that requires very small time steps, i.e. increasingly smaller time steps since the system is large and dense.

However the NSCD–NLGS method has two drawbacks. The first one is related to the velocity formulation for the non-interpenetration introduced by Moreau [1]. This formulation is exact for time continuous modeling, but may generate parasitic interpenetrations in a time-stepping integration scheme. The second drawback is the slow convergence of the NLGS algorithm. For dense granulates submitted to a quasi-static regime the convergence may even fail as illustrated in the following. We can relate this problem to the indetermination generated by self-equilibrated force networks in strongly confined granulates [7]. When a dense granulate is submitted with imposed velocities on its boundary, the two drawbacks are added leading in some cases to accumulation of the interpenetration in some localized regions. For instance the simulation of

* Address: Laboratoire de Mécanique et Génie Civil (LMGC), University Montpellier 2/CNRS UMR5508 CC048, Place E. Bataillon, F-34095 Montpellier Cedex 5, France. Tel.: +33 4 67 14 39 89; fax: +33 4 67 14 39 23.

E-mail address: pierre.alart@univ-montp2.fr

a sandbox experiment, useful in geomechanics for fold formation in sedimental soil layers [8], leads to such situations in the bottom corner close to the pushing wall.

Starting from an analysis of these above mentioned difficulties, the objective of this paper is to propose an enriched algorithm to accelerate the convergence or at least to control this convergence and the quality of the solution. Since the problem admits a plurality of solutions, specially for the frictional contact case, we cannot define an error measure with respect to a reference solution. As the resolution in a forthcoming strategy could be incomplete at each time step, the solutions will be compared in term of global behavior and of violation of the interpenetration. The proposed approach strives to respect at best the nonsmooth contact dynamics framework. However some regularization techniques are introduced in Sections 3 and 4 but restricted to the prediction stage. The large size of the time step permitted by the NSCD approach is preserved and could be even increased.

The present paper is an exploratory investigation that examines certain ways to overcome some numerical difficulties in granular dynamics. Therefore the numerical tests are performed with MAPLE and MATLAB on small examples because the implementation on a general discrete element software program such as LMGC90 requires major efforts that may profoundly modify the structure of the solver (<http://www.lmgc.univ-montp2.fr/LMGC90/>). The main goal is to improve numerical tools for simulating the behavior of granular systems without assuming its regime (solid, fluid, gas). However the present study is a stage in the investigation of the stresses in dense granular systems. We would like to distinguish between contacts and impacts in the internal impulse network: on the one hand a global network of contact forces with possible self-balanced contributions, on the other the local impacts. An attempt to deal with this splitting may be found in [25]. Such an approach may be useful for developing an homogenization strategy or a domain decomposition method [9,10].

Section 2 recalls the basis of the NSCD integration scheme before illustrating the drawbacks via very simple examples. Section 3 is devoted to braving the impulse indetermination by handling the null space of the Delassus operator. A prediction–correction algorithm is investigated in terms of convergence acceleration and the evolution of the interpenetration. Such an algorithm is an incomplete resolution. The initial equations are not exactly verified at each time step, as in explicit integration schemes. It is a risky challenge for dealing with nonsmooth relationships. Some numerical tests allow to appreciate the advantages and drawbacks of such a strategy. A new prediction stage is proposed in Section 4 to overcome the interpenetration drift generated by the previous algorithm. Finally, Section 5 adapts a technical framework developed in [11] for preserving the sparsity of the matrices and consequently for assuring an efficient implementation. The standard NLGS solver first has to be modified to deal with a sparse representation of the null space basis.

2. Nonsmooth contact dynamics

The NSCD method is based on a velocity-impulse formulation of the multicontact dynamics that allows velocity jumps and nonsmooth contact laws.

2.1. Components

The equations are directly presented according to a time-stepping integration scheme and with imposed velocities for some boundary bodies. The variables are summarized in Table 1. The multicontact system is restricted to a collection of rigid bodies. The possible configurations of the system are parametrized through generalized coordinates, say x , from which derive the generalized velocities V . The dimension of x is the number of degrees of freedom. According to classical kinematic analysis the relative velocity v_α at some contact α is given by a relation $v_\alpha = H_\alpha^t V$, where H is a connectivity mapping. In order to deal with granular medias confined by prescribed velocity conditions, the generalized velocities V are split into free and given components, V_f and V_g . The mapping H is also split into H_f and H_g . Based on duality considerations (conservation of the power expressed with local variables or generalized variables), the representative R_α of the local reaction impulse r_α for the generalized variable system satisfies the relation, $R_\alpha = H_\alpha r_\alpha$.

Table 1
General notations.

$[t^i, t^f]$	Time interval ($dt = t^f - t^i$, time step)
x^i (resp. x for x^f)	Generalized coordinates at time t^i (resp. t^f)
V^i (resp. V for V^f)	Generalized velocities at time t^i (resp. t^f)
V_f^i (resp. V_g^i)	Free (resp. given) velocities at time t^i
R^d	Given impulses over the time step
R	Contact impulses over the time step
g (for $g(x)$)	Normal contact gaps at contacts
v (for v^f)	Relative velocities at contacts
r	Contact forces or impulses at contacts
H (for $H(x)$)	Contact to grain linear mapping
H^T (for $H^T(x)$)	Grain to contact linear mapping
M (for $M_f(x)$)	Inertia matrix

Some variables depend implicitly on the generalized coordinates, but this dependency is omitted for convenience. Some variables are defined at the final instant t^f , but the exponent is suppressed for underlining that these variables are the unknowns. The fundamental equations are split into three classes and written in the following according to the global variables related to the bodies or to the contacts. For more details refer to [1,2].

Dynamics equation. The discretized equation of dynamics invokes the initial and final generalized velocities in the left hand side and the given impulses R^d and the contact impulses R in the right hand side. The contact impulses on a given body of made of the covariant components of the impulses at all the contacts concerning the body via the admissibility equations, i.e. the integral over $[t^i, t^f]$ of the contact forces.

$$M(V_f - V_f^i) = R + R^d. \quad (1)$$

Admissibility equations. The connectivity mappings H and H^T links the generalized velocities to the relative velocities, and by duality the contact impulses between covariant and contravariant components.

$$\begin{aligned} v &= H^T V = H_f^T V_f + H_g^T V_g, \\ R &= H_f r. \end{aligned} \quad (2)$$

Contact behavior (formally). All the contact laws are formally represented by a relation between the local relative velocity v_α and the contact impulse r_α . Some parameters may be implied in this relation, like the gap, the cumulated sliding, the wear...

$$\mathcal{R}(v, r) = 0 \iff \mathcal{R}(v_\alpha, r_\alpha) = 0, \quad \alpha = 1, n_{\text{contact}}, \quad (3)$$

The reduced dynamics condenses the linear equations (1) and (2).

$$H_f^T M^{-1} H_f r - v = -H_f^T V_f^i - H_g^T V_g^f - H_f^T M^{-1} R^d := -v^d. \quad (4)$$

Synthetically, the so-called reduced dynamics are expressed by introducing the *Delassus operator* $W (= H_f^T M^{-1} H_f)$,

$$W r - v = -v^d. \quad (5)$$

2.2. The generic NSCD integration scheme

The interaction conditions are formally written in formula (3) as a set of relations between local variables v_α and r_α . Among the interactions, the unilateral contact law has to be rewritten in term of velocity because the natural formulation uses a complementarity law between the normal gap and the normal contact force. The *viability lemma* in [1] justifies the formulation of a velocity-impulse Signorini law as an algorithm applied to the normal components of the relative velocity and the contact impulse for all contacts α :

$$\begin{aligned} \text{if } \tilde{g}_\alpha \geq 0, & \quad \text{then } r_{\alpha,n} = 0, \\ \text{if } \tilde{g}_\alpha < 0, & \quad \text{then solve } 0 \leq v_{\alpha,n} \perp r_{\alpha,n} \geq 0. \end{aligned} \quad (6)$$

The notation $0 \leq v_{\alpha,n} \perp r_{\alpha,n} \geq 0$ is a synthetic formulation of complementarity conditions generally expressed as $v_{\alpha,n} \geq 0, r_{\alpha,n} \geq 0, v_{\alpha,n} r_{\alpha,n} = 0$. The notation \tilde{g}_α refers to a prediction of the gap and not to an exact value. This prediction may use different strategies.

- Adopt the value reached at the end of the previous step: $\tilde{g}_\alpha = g_\alpha(x^i) = g_\alpha^i$.
- Predict a value at mid-point: $\tilde{g}_\alpha = g_\alpha(x^i + \frac{dt}{2} V^i)$.
- Define the connectivity mapping H .

Associating a velocity/impulse formulation with a gap/force formulation has been justified from an algorithmic viewpoint in [12,13] to stabilize energy-consistent methods for elastodynamics in the finite element context. For instance in [13], an adapted continuation Newton method is performed which is decomposed in two steps. In a preliminary step, the Newton method solves the equation with a gap/force formulation of the Signorini condition. In a second step, the Newton iterations are continued by replacing the gap/force formulation by a velocity/impulse formulation. Similar ideas have been developed in [14] in the framework of nonsmooth dynamics based on the Gear–Gupta–Leimkuhler method [15]. These approaches could be investigated for the granular dynamics, but they are unwieldy from the implementation viewpoint because, for the granular dynamics, the number of contact unknowns is much larger than the number of degrees of freedom.

The NSCD integration scheme is presented in Algorithm 1 with a standard prediction of the gap ($\tilde{g} = g^i$), but another choice may be made. Tasks deriving from the contact detection are mentioned in Algorithm 1 because these items are useful in the forthcoming algorithms.

Algorithm 1. NSCD integration scheme

for i step = 1, . . . , n step **do**

Contact detection and

 Compute the predicted gap ($\tilde{g} = g^i$) Define the potential contacts α ($\tilde{g}_\alpha \leq 0$) Define the associated mappings H, H_f, H_g Initialize the configuration (x^i, V^i, r^i) at time t^i Compute the Delassus operator W and the right hand-side v^d

Solve the following system with the NLGS algorithm (Algorithm 2)

$$\begin{cases} Wr - v = -v^d \\ \mathcal{R}(v, r) = 0 \end{cases} \quad (7)$$

Update the configuration ($r^f = r$)

$$\begin{cases} V_f^f = V_f^i + M^{-1}(H_f r^f + R^d) \\ x^f = x^i + dt V^f \end{cases} \quad (8)$$

end for

The solver, based on a nonlinear Gauss–Seidel method [3], is detailed in Algorithm 2 before being substantially modified for an efficient implementation in Section 5.

Algorithm 2. NLGS solver

for $k = 1, \dots, convergence$ **do** **for** $\alpha = 1, \dots, n_{contact}$ **do** {Loop over the contacts} Determine (r_α^k, v_α^k) such that

$$\begin{cases} W_{\alpha\alpha} r_\alpha^k - v_\alpha^k = -v_\alpha^d - \sum_{\beta < \alpha} W_{\alpha\beta} r_\beta^k - \sum_{\alpha < \beta} W_{\alpha\beta} r_\beta^{k-1} \\ \mathcal{R}(r_\alpha, v_\alpha) = 0 \end{cases} \quad (9)$$

end for**end for**

For all cases, the exact interpenetration cannot be ensured at each time step. The main variable in the system (7) is the contact impulse r and the updating of the position of the bodies has to be performed via the contact impulse. An artificial correction of the position of the bodies would perturb strongly the global behavior. Moreover, if the convergence of the NLGS algorithm is not reached after many time steps, cumulated interpenetration may occur because the computed contact impulses are not large enough to repulse the contacting bodies. Such a situation is specially critical when the W matrix is singular, with a null space in which the algorithm cannot converge. Thus the indetermination of the contact network is a crucial point for cumulated interpenetrations and convergence failure.

The friction law is naturally formulated between the tangential relative velocity and the tangential contact impulse. The numerical treatment with the NLGS algorithm is technical but does not generate divergence of the algorithm. However a strong friction coefficient leads to wedging effects which may amplify the accumulation of interpenetrations.

2.3. Some small instructive examples

The behavior of this numerical scheme may be illustrated with some simple examples involving a confined system by some non null prescribed velocities on the boundary. The simulations are performed with MAPLE sheets using classical routines of the Linear Algebra package. The first one is a very simple example with two disks, submitted to gravity, progressively wedged between two walls. The right wall is guided with a constant horizontal velocity V_{imp} . The radius of the disks is slightly different leading to an almost pathological situation when wedging. The process is performed with 100 time steps and sets a maximal number of iterations (150). For all the forthcoming simulations, since the contact relations (3) are exactly solved, the convergence test requires only that the relative error on the residuum be less than 10^{-3} ,

$$Err = \frac{\|Wr^k - v^k + v^d\|}{\|v^d\|} < 10^{-3}. \quad (10)$$

The first simulation uses a frictionless contact law for all five contacts. In the second simulation different friction coefficients are postulated to the contacts according to the numbering given in Fig. 1: $\mu_1 = 1, \mu_2 = 0.01, \mu_3 = 0.01, \mu_4 = 1, \mu_5 = 0.01$. A first version of the NLGS algorithm does not use an initialization of the contact impulses (or $r^0 = 0$); it is called *without memory*. The second version, called *with memory*, initializes the impulses with the values obtained for the previous time step.

The evolution of the number of NLGS iterations is given in Fig. 2 for the three cases. In the same figure the evolution of a mean interpenetration violation is represented. This mean violation is computed with the following formula, where n_c is the number of contacts,

$$Violation = \sum_x \frac{\max(0, -\tilde{g}_x)}{n_c}. \quad (11)$$

For the frictionless case, when wedging occurs, the maximal number of iterations is reached between time steps 42 and 44 and the violation increases noticeably. Without fixing the maximal number of iterations the time step 42 requires 37,500 iterations! But the wedging is released at the next step. In Fig. 3 the normal impulses are represented at different times. During wedging, their intensity increases until removal of the wedge leading to an increase in the kinetic energy of the first disk (cf. the position of the disk at iteration 45).

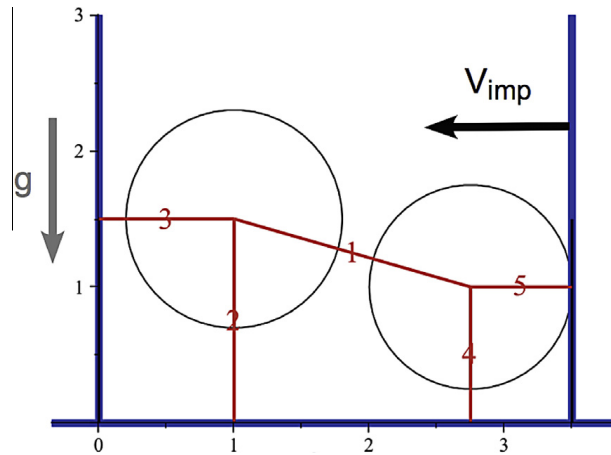


Fig. 1. A two disk example under gravity in a box with an imposed velocity on the right wall – initial configuration – radius equal to 0.8 cm for the left disk and 0.75 cm for the right disk.

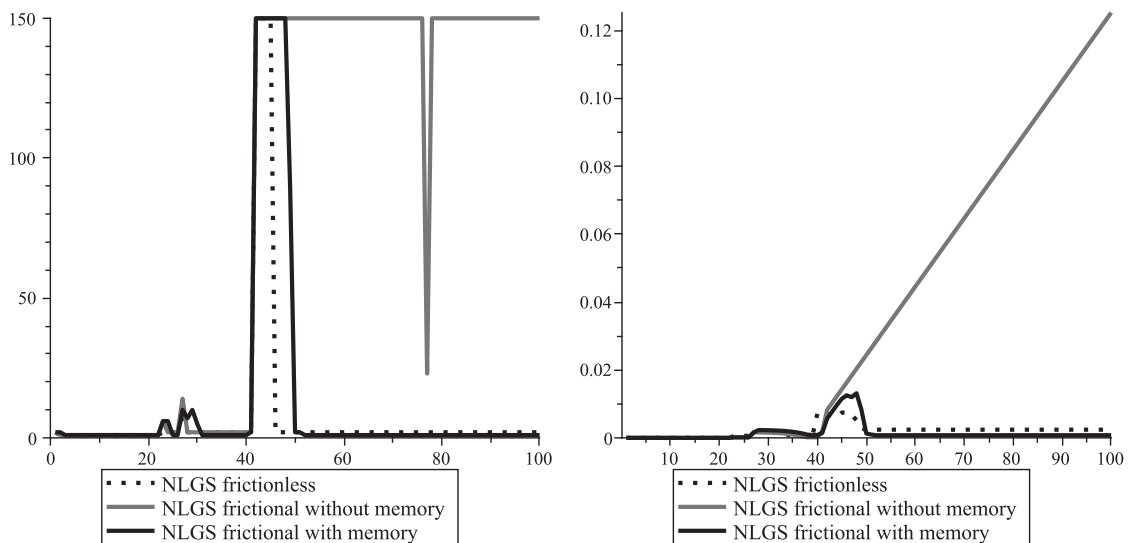


Fig. 2. Evolution of the number of iterations (on the left) – evolution of the violation (on the right). The first figure gives the number of iterations at each time step (for 100 time steps). The number of iterations is bounded to 150. The second figure gives the interpenetration in proportion of the mean radius of the disks. The forthcoming figures use the same representation.

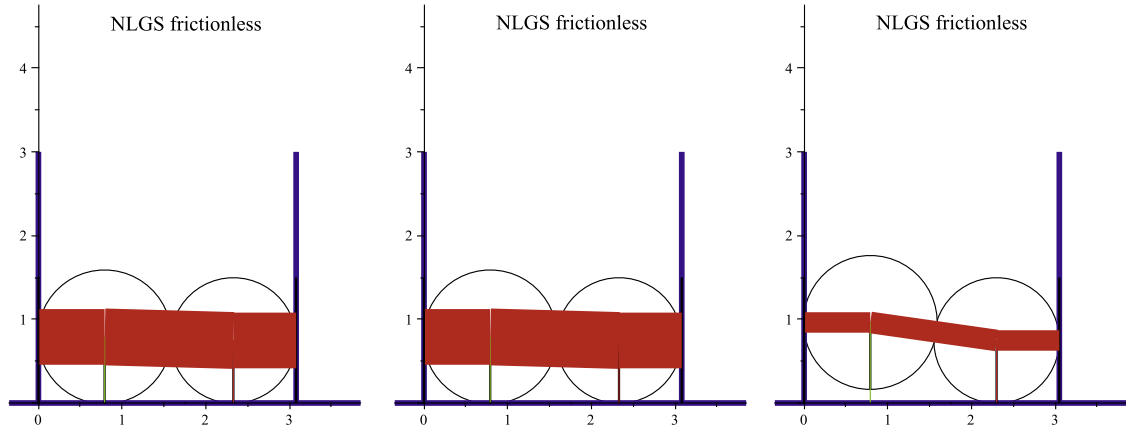


Fig. 3. Frictionless case – time steps 42, 43, 45 – normal impulses.

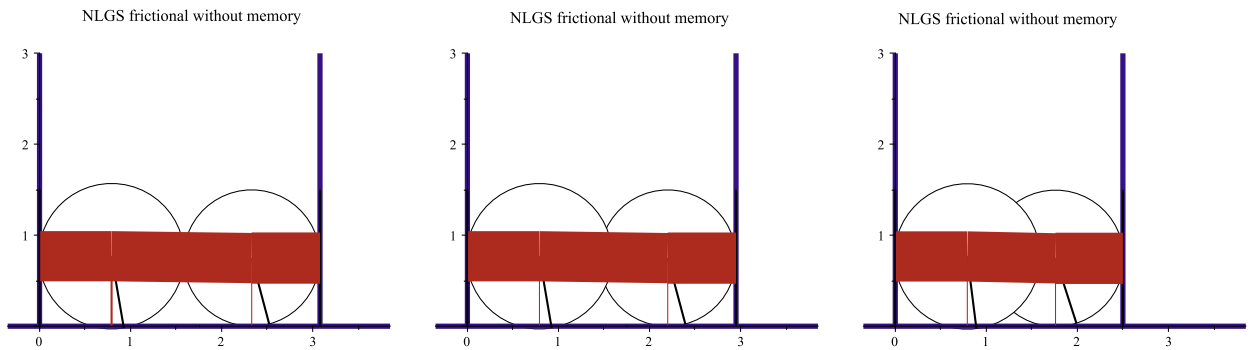


Fig. 4. Frictional case *without memory* – time steps 42, 55, 100 – normal impulses.

For the frictional case, the wedging leads to nonconvergence of the NLGS algorithm until the end of the process when the *without memory* version is performed (cf. Fig. 2). Consequently the normal impulses do not increase during the second phase but the interpenetration becomes nonadmissible as illustrated in Figs. 2 and 4.

Using the *with memory* version, one recovers a behavior closer to the frictionless case with a longer contact phase between the two disks and an increase in the normal impulses until removal of the wedge and a larger artificial energy restitution (cf. Figs. 2 and 5).

The second example involves 5 disks submitted to similar constraints as represented in Fig. 6. Ten contacts are taken into account during the process without allowing new ones. Only a frictional case is considered with the friction coefficients: $\mu_1 = 0.1, \mu_2 = 1, \mu_3 = 0.1, \mu_4 = 1, \mu_5 = 1, \mu_6 = 1, \mu_7 = 10, \mu_8 = 0.1, \mu_9 = 0.1, \mu_{10} = 10$. This distribution implies the continuous rotation of some disks: anticlockwise for disks 2 and 4, clockwise for disk 5 (cf. Fig. 8). As for the previous example the *with memory* version improves the convergence for a comparable violation (Fig. 7). The configurations at steps 14 and 25 in Fig. 8 illustrate the influence of the different friction coefficients. This final configuration will enable comparison of solutions obtained with the forthcoming algorithms.

These examples involve a velocity-driven loading because this loading is the main source of convergence degradation. It is not always possible to replace a velocity-driven loading by a force-driven loading, specially when the total reaction force of the granular system on an obstacle is not a monotonously increasing. Such a situation occurs for example on the simulation of sandbox experiments in geomechanics [8].

3. Braving the indetermination

3.1. Dealing with the null space of the Delassus operator

The bad (slow and sometimes failing) convergence of the generic algorithm may be attributed both to the nonsmoothness of the frictional contact law and to the singularity of the W matrix. These features lead to the indeterminacy of the impulse network, that has often been considered as a defect in the Coulomb friction theory. For instance in [17] concerning the Painlevé paradox [16], the authors examine the problem of inconsistencies, i.e. configurations for which no continuous solution exists, as well as indeterminacies, i.e. configurations which lead to non-uniqueness of solutions.

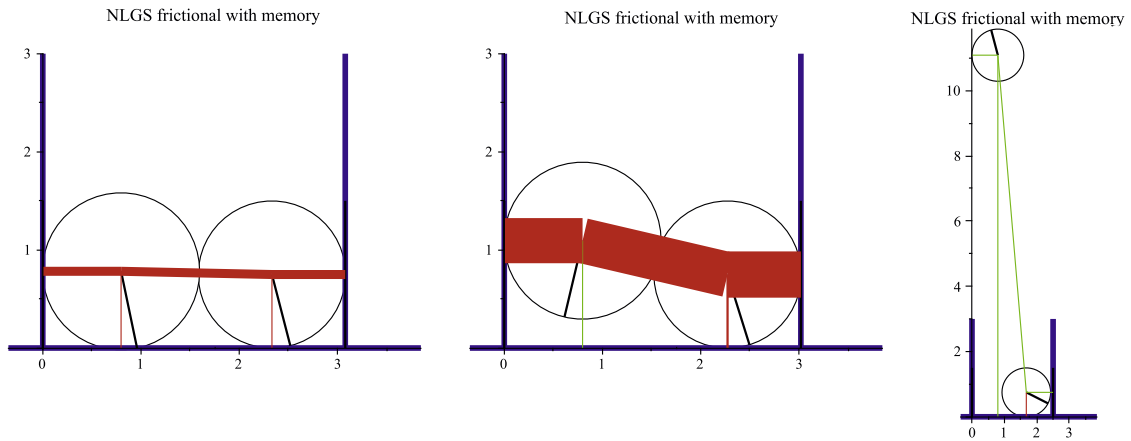


Fig. 5. Frictional case with memory – time steps 42, 48, 100 – normal impulses.

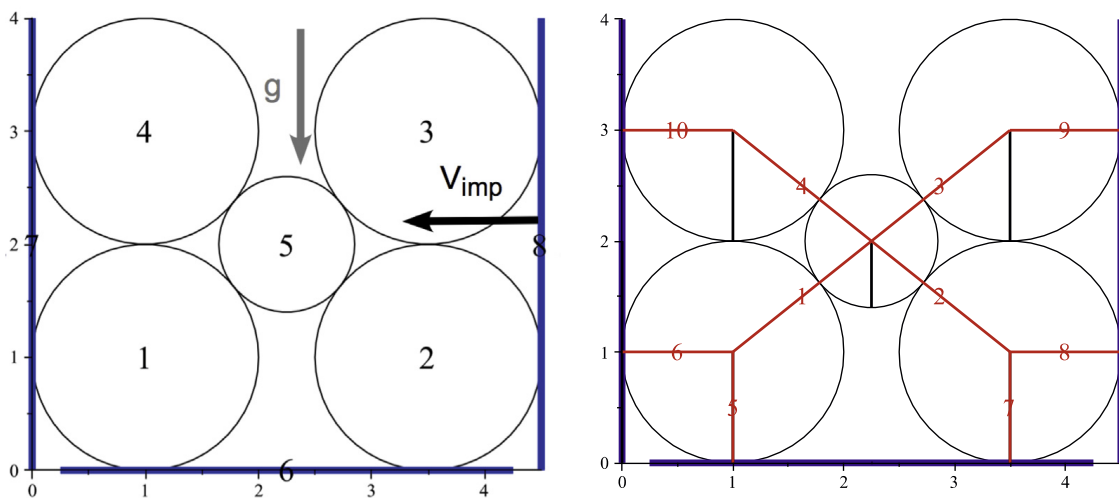


Fig. 6. A five disk example (initial configuration and solicitations) – body numbering and contact numbering.

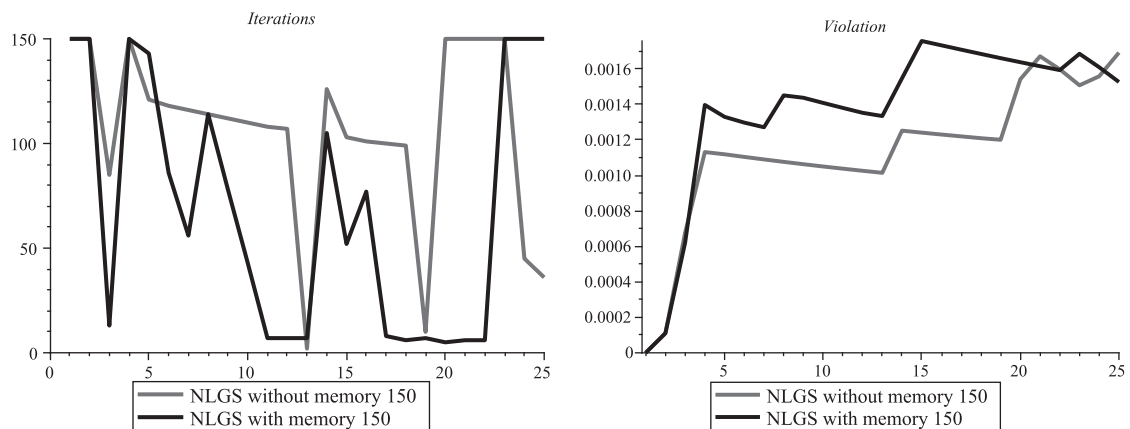
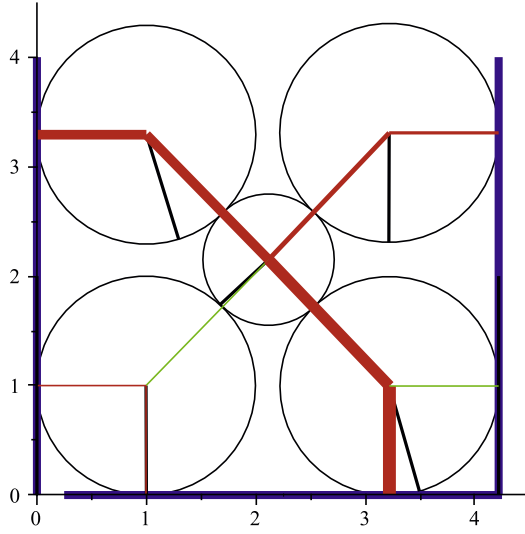


Fig. 7. Evolution of the number of iterations – evolution of the violation (5 disks, 25 time steps).

An attempt to represent the multiplicity of solutions in granular systems in terms of force network or acceleration distribution using a statistical approach is proposed in [18,7]. We focus here our attention on the null space of W or H which represents the self-equilibrated force networks that are particularly numerous in dense granulates.

Our first way consists of dealing explicitly with this null space in the numerical scheme. The problem initially summarized in Eqs. (3) and (5) is reformulated by introducing a basis N of the null space of H ,

NLGS with memory 150



NLGS with memory 150

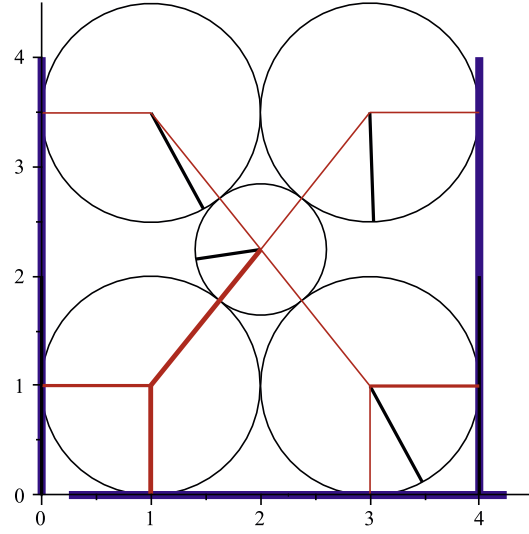


Fig. 8. Configurations – time steps 14 and 25 (final) – normal impulses.

$$\begin{cases} (W + NN^T)r - v - N\lambda = -v^d, & \text{(a)} \\ \mathcal{R}(r, v) = 0, & \text{(b)} \\ N^T(v - v^d) = 0, & \text{(c)} \end{cases} \quad (12)$$

The Delassus operator is regularized in the first equation and the last equation forces to the right hand side of the reduced dynamics to be orthogonal to the null space, as a mathematical constraint. The inverse of the regularized Delassus matrix defines a pseudo-inverse matrix of W . Eqs. (12a) and (12c) are linear whereas the relations (12b) are nonlinear and non-smooth. This system may be solved by introducing a Lagrange multiplier. An Uzawa type algorithm then involves two stages at each iteration: an incomplete solution of a regularized problem (12a) and (12b) by a single iteration of the NLGS method, and then an updating of the multiplier according to the constraint (12c), as summarized in the Algorithm 3.

Algorithm 3. NLGS solver with a multiplier dealing with the null space (NLKerGS-I)

for $k = 1, \dots, \text{convergence}$ **do**

STAGE 1: Perform a single NLGS iteration,

for $\alpha = 1, \dots, n_{\text{contact}}$ **do** {Loop over the contacts}

Determine (r_α^k, v_α^k) such that

$$\begin{cases} \widetilde{W}_{\alpha\alpha} r_\alpha^k - v_\alpha^k = -v_\alpha^d + (N\lambda^{k-1})_\alpha - \sum_{\beta < \alpha} \widetilde{W}_{\alpha\beta} r_\beta^k - \sum_{\alpha < \beta} \widetilde{W}_{\alpha\beta} r_\beta^{k-1} \\ \mathcal{R}(r_\alpha, v_\alpha) = 0 \end{cases} \quad (13)$$

end for

STAGE 2: Update the multiplier,

$\lambda^k = \lambda^{k-1} + \rho N^T(v^k - v^d)$ (version 1) or $\lambda^k = N^T r^k$ (version 2)

end for

The regularized $W + NN^T$ matrix is denoted \widetilde{W} and a convergence parameter ρ is introduced. For this algorithm ρ must be chosen at best. An analysis of a single example with one contact proves that the best spectral radius is close to 1 for all ρ . We propose to improve the previous algorithm by defining an enriched second stage. This consists of determining the pair $(v^{k+1/2}, \lambda^k)$ satisfying the first and third equations of (12),

$$\begin{cases} v^{k+1/2} + N\lambda^k = (W + NN^T)r^k + v_\alpha^d, \\ N^T v^{k+1/2} = N^T v^d, \end{cases} \quad (14)$$

By eliminating $v^{k+1/2}$ a possible solution of the resulting equation is $\lambda^k = N^T r^k$ as reported in Algorithm 3 in a second version. An advantage of this algorithm is that it does not require the choice of a new numerical parameter.

The study of the performance of these two algorithms (postponed in the last subsection) proves that their convergence behavior is similar. Moreover, the convergence rate is identical to that of the generic algorithm (but the computational cost is much greater). A new improvement attempt may be to initialize at best the multiplier. A first approach consists of solving a predictive linearized problem. We propose to restrict this approach to the unilateral contact because the contact may be predicted by the relations (6) in an explicit way, whereas the frictional status is not easily reachable. Moreover, it is too restrictive to postulate an adherent contact status for all contacts and some numerical tests, not presented here, prove it. Then, once the contact is predicted, the complementarity condition in (6) is approximated by a linear relation where η is a small parameter,

$$\begin{aligned} \text{if } g_\alpha^i \geq 0, \quad & \text{then } \tilde{r}_{\alpha,n} = 0 \\ \text{if } g_\alpha^i < 0, \quad & \text{then } \tilde{r}_{\alpha,n} = -\frac{dt}{\eta} \tilde{v}_{\alpha,n} \quad (\eta \rightarrow 0) \end{aligned} \quad (15)$$

From a computational viewpoint, a sequence of computations is performed starting from the definition of a diagonal matrix D representing the predicted active contacts (see Algorithm 4). The linear system to solve involves only the free velocities V_f like unknowns, but the predicted impulse uses all the velocities.

Algorithm 4. Linear predictive stage and multiplier initialization

Define D matrix: $D_{\alpha\alpha,n} = \begin{cases} 0 & \text{if } g_\alpha^i \geq 0 \\ 1 & \text{if } g_\alpha^i < 0 \end{cases}$
Solve the linear system: $(M + \frac{dt}{\eta} H_f D H_f^T) \tilde{V}_f = R^d + M V_f^i - \frac{dt}{\eta} H_f D H_g^T V_g^f$
Define the predicted impulse: $\tilde{r} = -\frac{dt}{\eta} D H^T \tilde{V} (:= r^{lin})$
Initialize the multiplier: $\lambda^0 = N^T \tilde{r}$

3.2. A single prediction/correction algorithm

As the numerical study shows that the convergence is not improved with some initialization, we propose to suppress the iterative process between the two stages. The thus obtained Algorithm 5 comprises a predictive stage and a corrective one using the regularized Delassus operator and an enriched right hand side. This strategy leads to an incomplete resolution. Eq. (12c) is not automatically verified. This is then a scientific wager to introduce a partial explicit strategy in an implicit integration scheme for a non smooth problem. It may be first justified by underlining that Eq. (12c) is verified if the problem does not invoke prescribed velocities or invokes prescribed null velocities, $V_g = 0$. For a general case, such a strategy has to be validated by comparing the solutions in term of final configurations and interpenetration.

Algorithm 5. A single prediction/correction algorithm using the null space (NLKerGS-P)

Prediction: $\tilde{r} = 0$ or r^{lin} (Algorithm 4) or r^{unil} or r^{elas} (Algorithm 6)
Correction: perform NLGS to find (r, v) such that

$$\begin{cases} (W + NN^T)r - v = -v^d + NN^T \tilde{r} \\ \mathcal{R}(r, v) = 0 \end{cases} \quad (16)$$

A limit predictive process may be defined by solving the frictionless unilateral contact problem. Indeed, note that the behavior of the generic algorithm applied to a frictionless contact problem is much better than applied to a frictional problem. The impulse derived from this strategy is then denoted r^{unil} for the forthcoming numerical study. It is performed by calling the NLGS solver with a zero friction coefficient. The zero initialization of the impulses is considered in order to evaluate the advantages of the other initializations. A last prediction is denoted r^{elas} and is specified in a following section.

3.3. Numerical experiments

This section summarizes the behavior of the different algorithms presented above applied to the 5 disk example.

The behavior of Algorithm 3 is given with dashed lines in Fig. 9 for the evolution of the number of iterations and in Fig. 10 for the violation: *NLKerGS-I-r0* for the zero initialization, *NLKerGS-I-rlin* for the linear initialization (Algorithm 4), *NLKerGS-I-runil* for the unilateral initialization. The number of iterations is similar to that obtained with the generic solver without memory and is greater than with memory. The violation level is similar for all initializations and comparable to the generic algorithm.

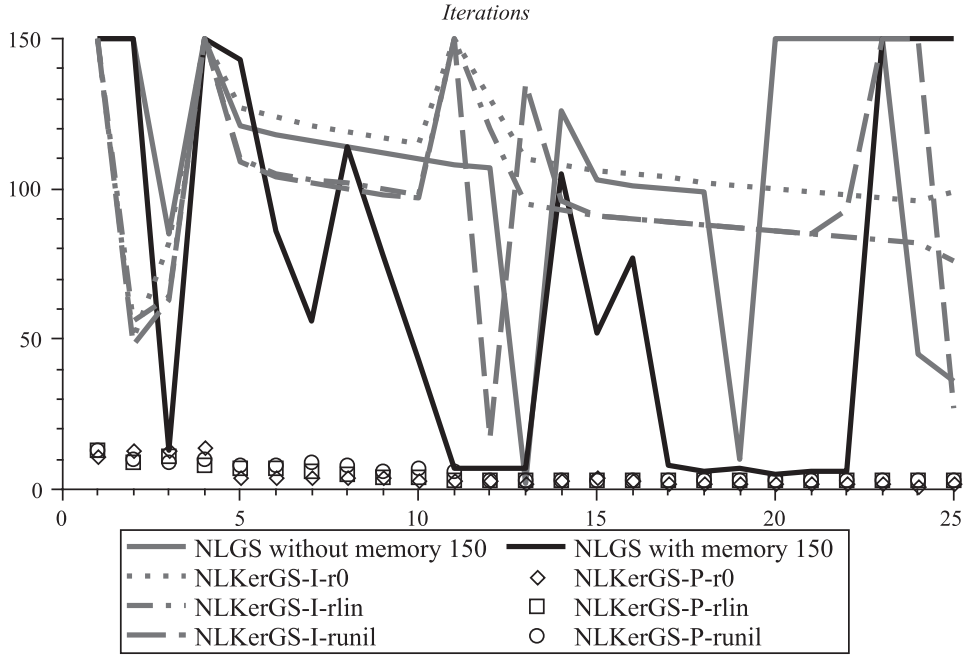


Fig. 9. Evolution of the number of iterations for the different algorithms (5 disks, 25 time steps).

The behavior of the prediction/correction Algorithm 5 is given with points in Fig. 9 for the evolution of the number of iterations and in Fig. c for the violation: *NLKerGS-P-r0* for the zero prediction, *NLKerGS-P-rlin* for the linear prediction (Algorithm 4), *NLKerGS-P-runil* for the unilateral prediction. The number of iterations decreases drastically for all predictions and is due to the regularized Delassus operator. On the other hand the interpenetration evolves differently for the three predictions. With a zero prediction, the violation quickly becomes inadmissible as confirmed in Fig. 11 where the final configurations are plotted. The final configurations obtained with the two other predictions are similar to the configuration derived from the generic solver plotted in Fig. 8. The interpenetration does not exceed 1 percent of the disk radius, as evaluated in Fig. 10, and rotations of the grains are very close. We can even observe more regular and symmetric impulse networks throughout the process. However the drift of the interpenetration for the two predictions is not negligible as illustrated in Fig. 10; it could become inadmissible for more complex and longer simulations. This drift has to be corrected before proposing an efficient strategy. The η parameter would be controlled with respect to the time increment and the mean mass of a grain. However two different values have been tested: $\eta = 0.0001$ for which the drift is equivalent to the unilateral prediction, $\eta = 0.001$ for which the drift is larger (not represented in Fig. 10). Consequently, it is not necessary to use a too small η coefficient that does not improve the evolution of the interpenetration.

4. Braving the interpenetration

4.1. An elastic prediction/nonsmooth correction algorithm

The previous study proves that the use of the velocity formulation of the unilateral contact to predict an impulse network leads to an interpenetration drift over time. Hence the regularized dynamics combined with a velocity formulation cannot supply an impulse network, or a force network, which holds the steric exclusion over the whole granular system. In other words, the correction stage dealing with a regularized Delassus operator cannot really correct the interpenetration drift generated by the linear prediction stage based on a velocity formulation of unilateral contact.

An alternative for the prediction stage starts from a classical Signorini law using the gap as the main kinematic variable: $0 \leq g_{\alpha,n} \perp r_{\alpha,n} \geq 0$. These complementarity conditions have to be approximated by a regularized relation which involves an elastic type parameter. Consequently we replace the linear prediction based on the velocity formulation by a linear elastic prediction as generally considered in quasistatic systems [19]. Then the contact law for the predictive stage is now written as follows, where $\langle \cdot \rangle^-$ denotes the non positive part,

$$\tilde{r}_{\alpha,n} = -\frac{dt}{\varepsilon} \langle \tilde{g}_{\alpha} \rangle^- = -\frac{dt}{\varepsilon} \langle g_{\alpha}^i + dt \dot{v}_{\alpha,n} \rangle^-. \quad (17)$$

An explicit prediction of this nonsmooth relation gives the following approximation,

$$\tilde{r}_{\alpha,n} \approx -\frac{dt}{\varepsilon} D_{\alpha\alpha,n} \tilde{g}_{\alpha} \quad \text{with} \quad D_{\alpha\alpha,n} = \begin{cases} 0, & \text{if } g_{\alpha}^i \geq 0, \\ 1, & \text{if } g_{\alpha}^i < 0. \end{cases} \quad (18)$$

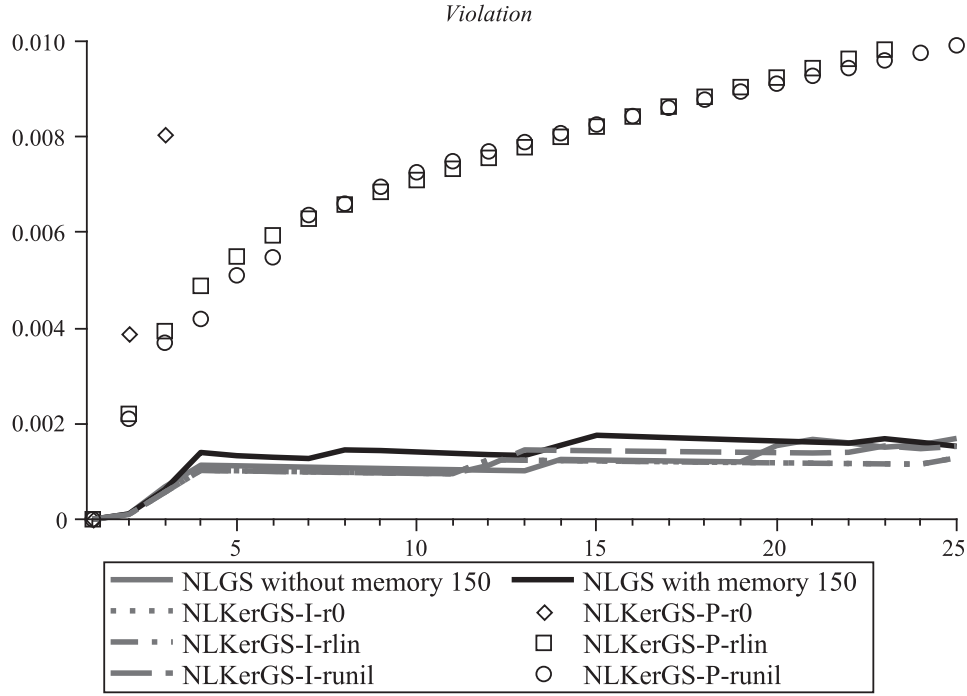


Fig. 10. Evolution of the interpenetration for the different algorithms (5 disks, 25 time steps).

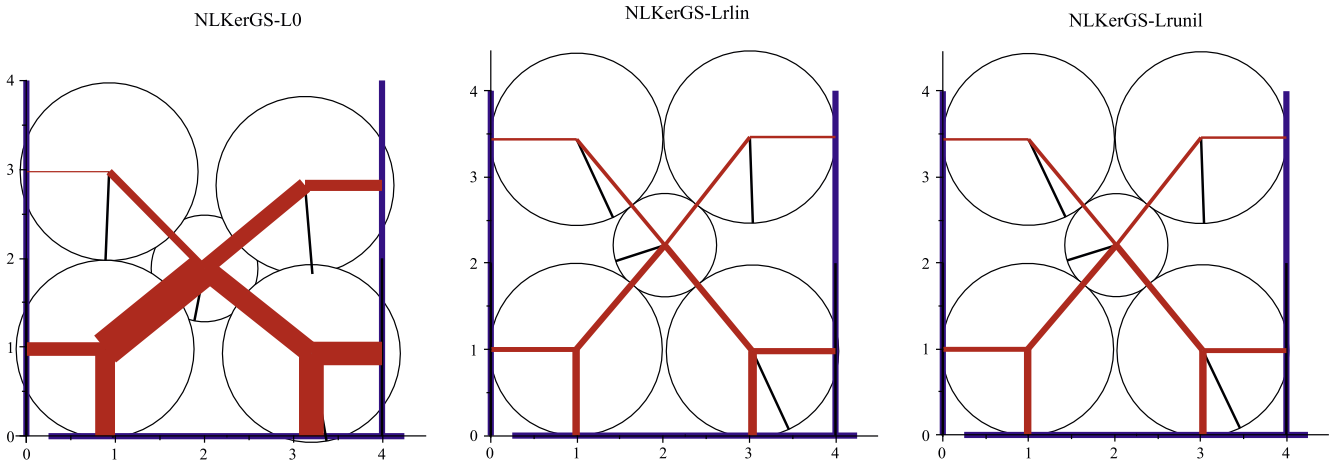


Fig. 11. Final configurations for the 3 predictions: zero, linear, unilateral.

In substituting (18) in the dynamics equation and using the admissibility equations, we obtain a regular linear system that is the main part of the elastic predictive stage detailed in Algorithm 6. This elastic prediction is then inserted in Algorithm 5.

Algorithm 6. Elastic predictive stage

Define D matrix: $D_{\alpha\alpha,n} = \begin{cases} 0 & \text{if } g_{\alpha}^i \geq 0 \\ 1 & \text{if } g_{\alpha}^i < 0 \end{cases}$

Solve the linear system: $(M + \frac{dt^2}{\epsilon} H_f D H_f^T) \tilde{V}_f = R^d + M V_f^i - \frac{dt}{\epsilon} H_f D (g^i + dt H_g^T V_g^f)$

Define the predicted impulsion: $\tilde{r} = -\frac{dt}{\epsilon} D (g^i + dt H^T \tilde{V}) (= r^{elas})$

4.2. Numerical tests

The prediction/correction algorithm is tested on the 5 disk example with different ϵ values (Fig. 12). The number of iterations is equivalent to the linear prediction. However the evolution is less regular because the micro-rebounds due to the

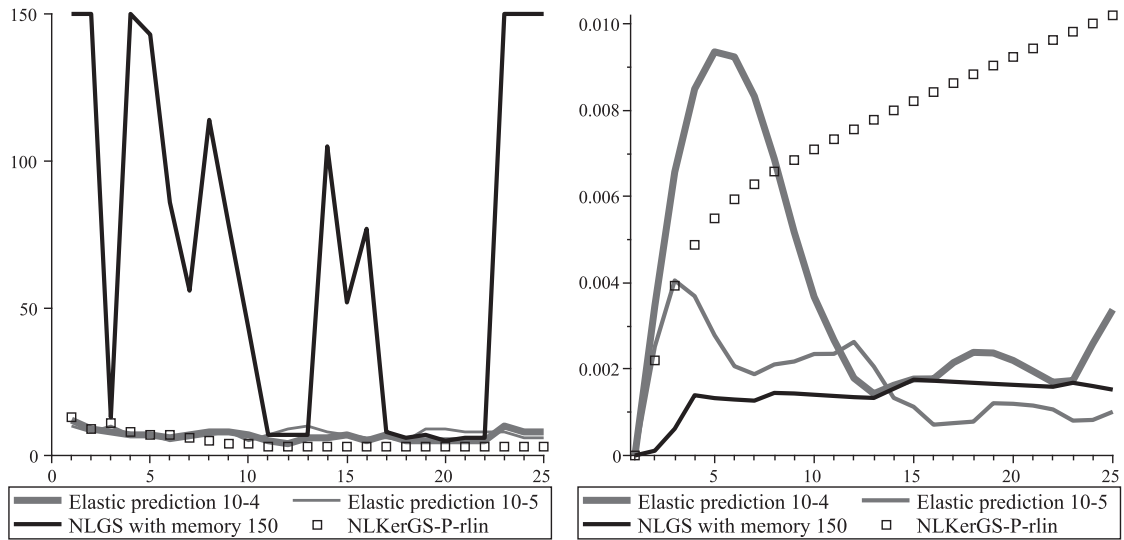


Fig. 12. Evolution of the number of iterations – evolution of the violation with the elastic prediction (5 disks, 25 time steps).

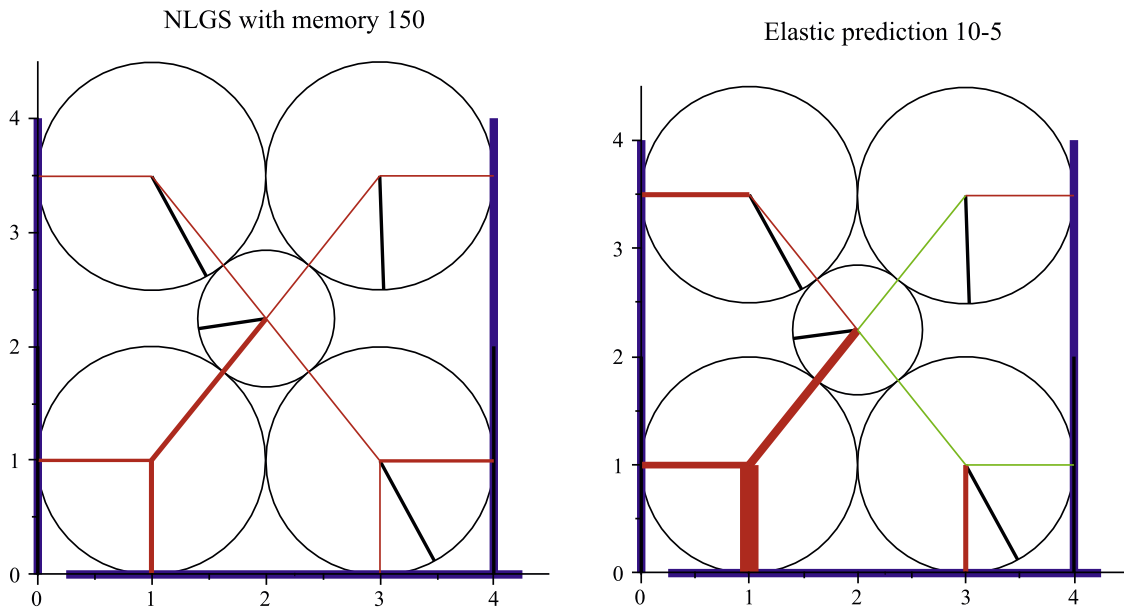


Fig. 13. Final configurations: generic algorithm and elastic prediction.

elastic prediction lead to frequent changes in the predicted contact statuses. A movie plotting the impulse network shows that the evolution of this network is more erratic with the smallest ε value. As expected, the elastic prediction corrects the drift of the violation all the more so since the elastic parameter ($\frac{1}{\varepsilon}$) is greater. For $\varepsilon = 10^{-5}$ the violation is equivalent to the generic algorithm and the final configuration is very similar (Fig. 13) even if the impulse network is significantly different.

4.3. Elastic parameter and dynamic effects

To study the influence of the elastic parameter on the solution, we consider a 30 disk example. The initial configuration is provided by the LMG90 software program to maximize the compactness and is plotted in Fig. 14. The potential contacts are listed in the figure and will not be updated throughout the simulation because a contact detection process is not performed on our MAPLE sheet. Then some parasitic interpenetrations could occur in some cases. Moreover, and contrary to the previous examples, the roundoff errors imply an initial interpenetration over the sample (see Fig. 15b).

Fig. 15 summarizes the numerical behavior of the different schemes. The maximum number of iterations for the generic algorithm is set at 300. This level is reached for the first six time steps what may be related to a wedging phase at the bottom of the sample. The elastic prediction–correction algorithm converges with 5 iterations on average for all ε values whereas the

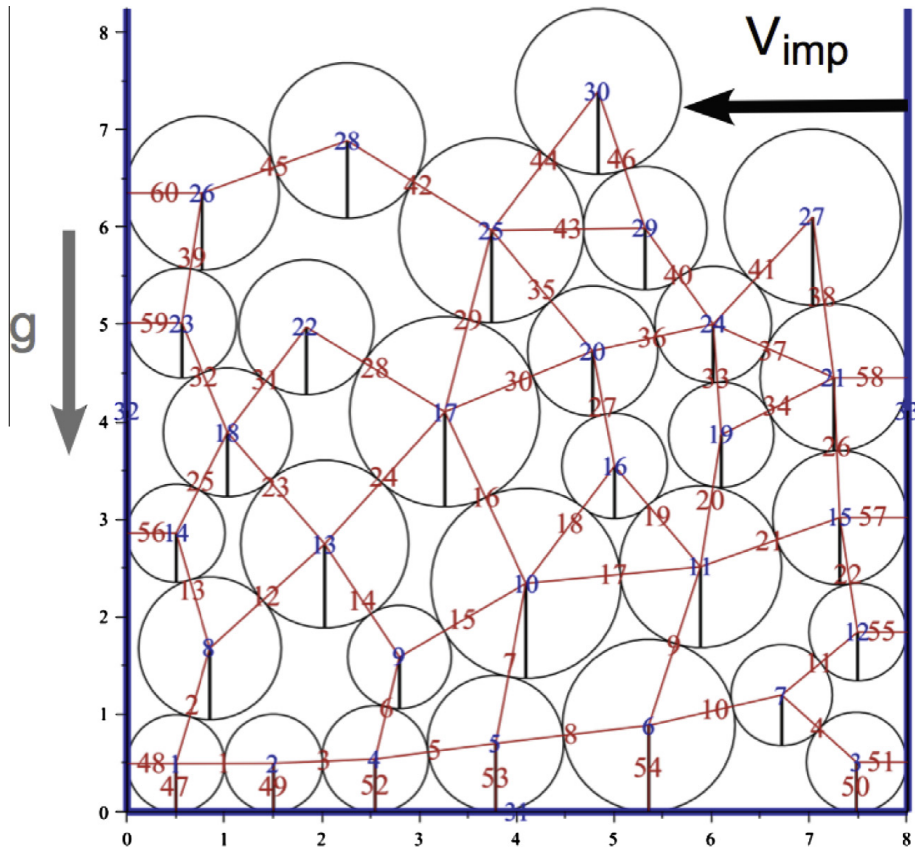


Fig. 14. A 30 disks example (initial configuration) – body, contact numbering and solicitations. The friction coefficient is equal to 0.2 between all the bodies.

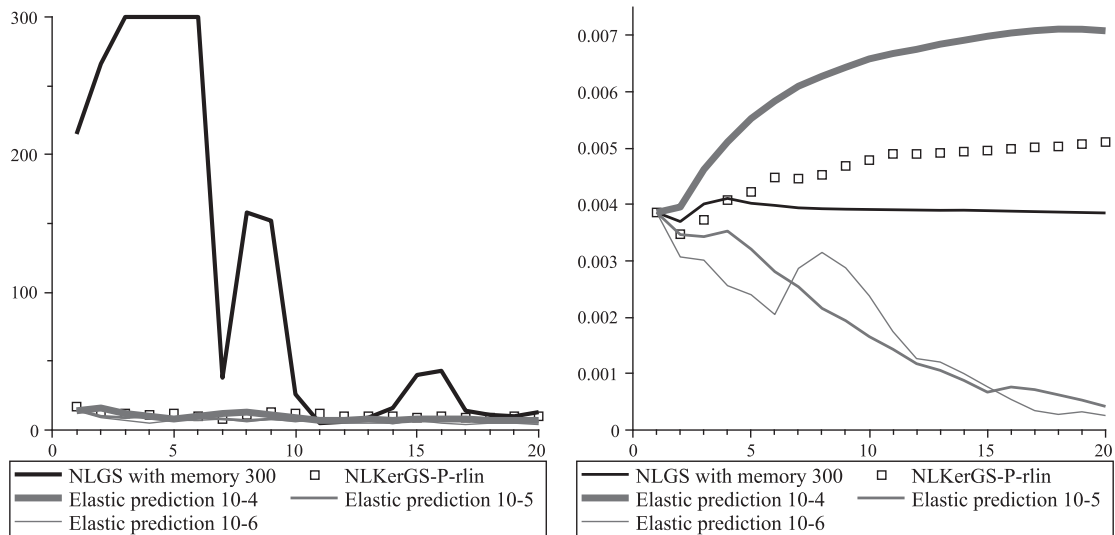


Fig. 15. Evolution of the number of iterations (the maximal number is fixed to 300) – evolution of the violation (30 disks, 20 time steps).

linear prediction–correction method requires 15 iterations on average for convergence. This increase in the required iterations seems to result from the maximum number of active contacts inherent to the linear prediction that cumulates the interpenetration throughout the simulation as observed in Fig. 15b. On the other hand, the elastic prediction with high elastic coefficients ($\varepsilon = 10^{-5}$ or 10^{-6}) partially reduces the initial interpenetration whereas a small value ($\varepsilon = 10^{-4}$) amplifies the interpenetration drift in comparison with the linear prediction.

Movies may underline the vibrational effects generated by a high elastic coefficient. In this paper only the final configurations are available in Figs. 16 and 17. The drift in the interpenetration is not observable on the linear prediction solution (Fig. 16). For a small elastic coefficient ($\varepsilon = 10^{-4}$), the wedging at the bottom of the box is not yet resorbed after 20 time steps

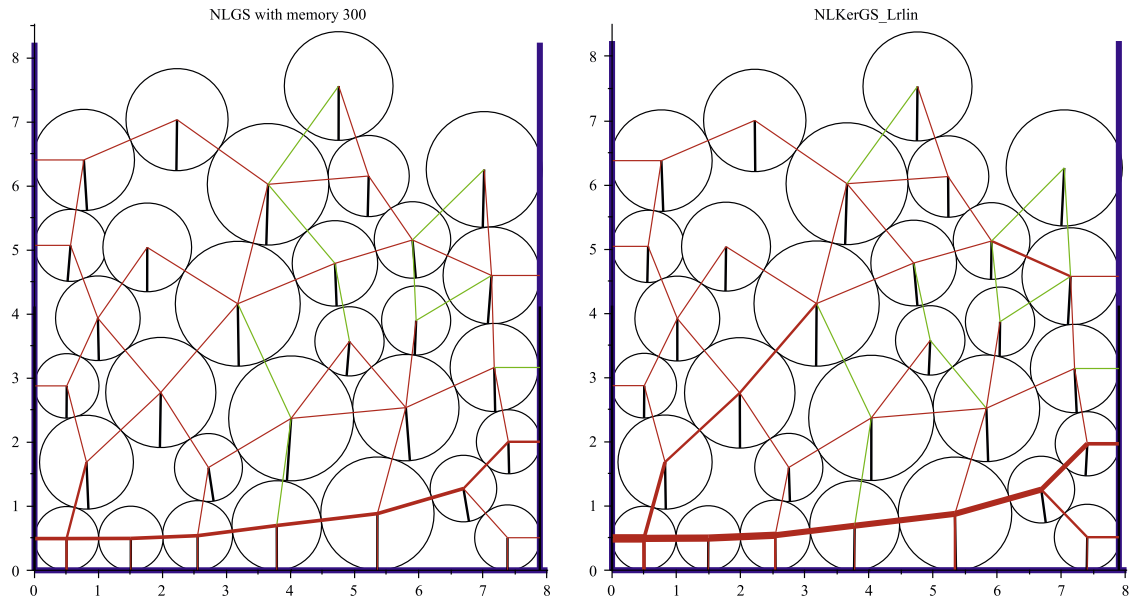


Fig. 16. Final configurations for the generic solver and the linear prediction.

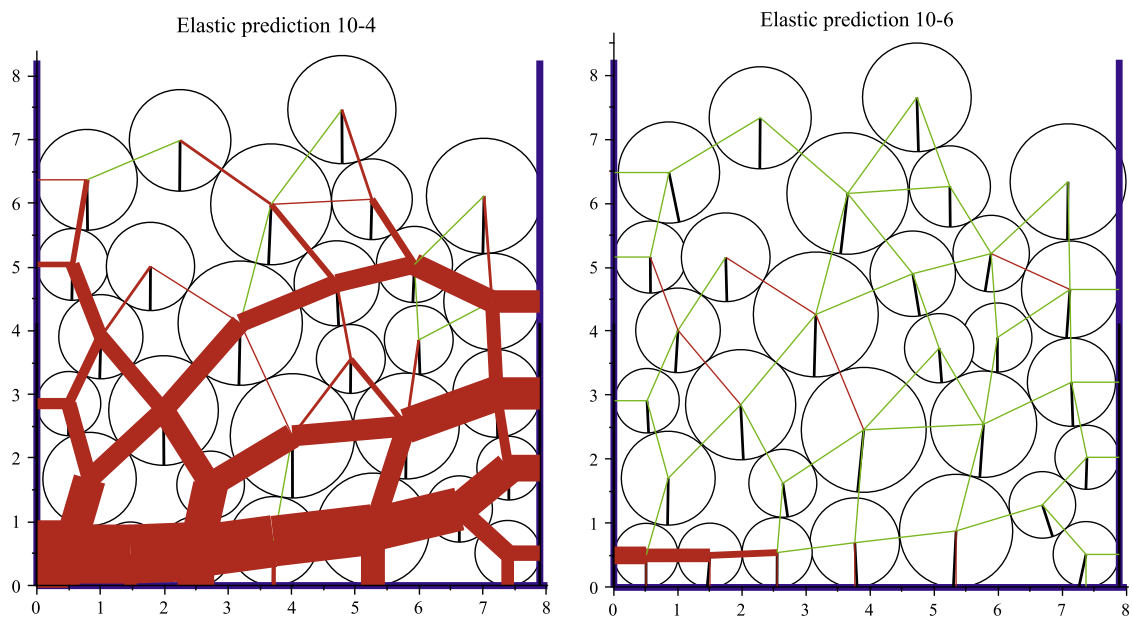


Fig. 17. Final configurations for two elastic coefficients.

(Fig. 17a). A large elastic coefficient implies micro rebounds between grains and an artificial dynamic behavior of the sample (Fig. 17b). The thickness of the impulses is not coherent from a figure to an other; the maximal thickness is calibrated over a process of one simulation.

This elastic prediction is a stopgap solution because we lost so the interest to use a velocity formulation of contact conditions. Such a velocity formulation is used for modeling inelastic shocks which occur primarily in dense granulates. However an elastic contribution is often introduced for controlling the interpenetration, even if this breaks the principles of the non smooth dynamics approach. The choice of elastic parameter is a delicate question depending on the local mechanical properties such as the contacting grains mass or the sollicitation frequency; this discussion is detailed by Jean in [20]. In some applications, it is useful to evaluate a posteriori the quasistatic or dynamic character of the problem by introducing an inertia parameter [4]. The behavior of a railway ballast submitted to a local tamping process or a train passing underlines different areas with a dynamic or static regime [21]. The present study provides only a qualitative evaluation of the dynamic effects in an academic problem. As mentioned above the present elastic prediction is not satisfactory from a conceptual point of view. A practical and compromise way could be to combine the linear and elastic predictions but the proportion of the two predictions depends strongly on the considered problem.

5. Efficient implementations

The previous strategy has one main drawback for dealing with large-scale granular systems that involve many thousands of bodies and contacts: the regularized Delassus operator is a full matrix because of the regularized part $Z = NN^T$. The marked reduction in the number of NLGS iterations can improve the CPU computing time only if the NLGS solver runs with a sparse matrix or at least with a sparse representation of the null space N . Before detailing the sparse representation of the null space, the NLGS solver has to be modified without explicitly computing the Z matrix.

5.1. A modified NLGS solver

As the Z matrix is never stored, only the results of the matrix–vector products Nx and $N^T x$ are assumed to be available for all x vectors. The following algorithm is a combination of a Gauss–Seidel method for the W part of \widetilde{W} and a Jacobi method for the Z part. The diagonal part of Z may be inserted in two different ways: we can compute the punctual diagonal or the block diagonal, for which each block refers to a α contact. This last version is proposed in Algorithm 7 for a two-dimensional model where $e_\alpha = [e_{\alpha,n}, e_{\alpha,t}] \in \mathbb{R}^{2n_{\text{contact}} \times 2}$.

Algorithm 7. NLGS-Jac solver without storing the Z matrix

Perform a procedure for computing Nx and $N^T x$
 Compute $N^T e_\alpha$, $\alpha = 1, \dots, n_{\text{contact}}$
 Compute the (block) diagonal part of $Z = NN^T$: $Z_{\alpha\alpha} = e_\alpha^T NN^T e_\alpha$, $\alpha = 1, \dots, n_{\text{contact}}$
 Compute the extra term in the right hand side: $\tilde{z} = Z\tilde{r}$ ($y = N^T \tilde{r}$, next $\tilde{z} = Ny$)
for $k = 1, \dots, \text{convergence}$ **do**
 Compute $z^{k-1} = Zr^{k-1}$ ($y = N^T r^{k-1}$, next $z^{k-1} = Ny$)
 for $\alpha = 1, \dots, n_{\text{contact}}$ **do** {Loop over the contacts}
 Determine (r_α^k, v_α^k) such that

$$\begin{cases} \widetilde{W}_{\alpha\alpha} r_\alpha^k - v_\alpha^k = -v_\alpha^d + \tilde{z}_\alpha - \sum_{\beta < \alpha} W_{\alpha\beta} r_\beta^k - \sum_{\alpha < \beta} W_{\alpha\beta} r_\beta^{k-1} - z_\alpha^{k-1} + Z_{\alpha\alpha} r_\alpha^{k-1} \\ \mathcal{R}(r_\alpha, v_\alpha) = 0 \end{cases} \quad (19)$$

end for
end for

The computation of $N^T e_\alpha$, for all α may be a consuming phase if the representation of the N null space requires a full matrix, but such a procedure may be performed using parallel processes. Before optimizing this stage, the performance of the modified NLGS solver has to be quickly analyzed in comparison with the NLGS solver (Algorithm 2) applied to the \widetilde{W} matrix.

5.2. Weighting the regularized part

The regularization of W is generally defined according to a weighting factor ω such that $\widetilde{W} = W + \omega Z$. As the nondiagonal part of Z is not taken into account in the modified NLGS solver, the contribution of ωZ should not to be too large in comparison with W . This topic would require a more extensive study, but in this exploratory investigation we focus our attention on a simple remark on an illustrative example.

The following developments are performed first with MAPLE, and then with MATLAB to simulate larger granular systems on a single time step. We consider the second step of the 30 disk example. This is yet a small example with 30 bodies and 60 contacts; that means that $H \in \mathbb{R}^{90 \times 120}$ and $W \in \mathbb{R}^{120 \times 120}$. The Euclidean norms of the matrices are respectively, $\|W\| = 16.9$ and $\|Z\| = 1.0$ (the null space basis is normalized). Table 2 gives the number of iterations with the standard NLGS solver and with the modified solver for different ω values. We finally add the gap between the 2 results to check that the two algorithms converge toward the same solution; the solution is not a priori unique.

Such behavior is confirmed on a larger example with 500 disks in Table 3. This is still a small sample (see Fig. 18) but the size of the matrices becomes large enough for full storage: $H \in \mathbb{R}^{1500 \times 2044}$ and $W \in \mathbb{R}^{2044 \times 2044}$. The Euclidean norm of the

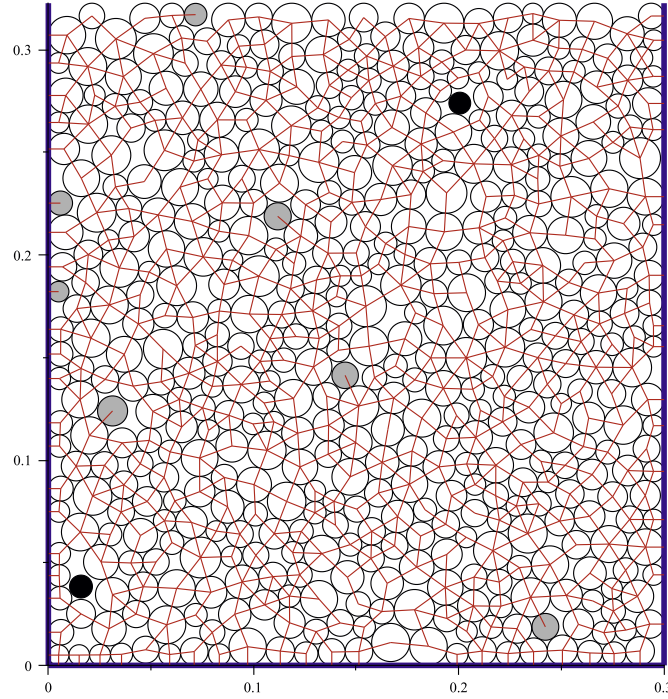
Table 2
 Convergence behavior of the modified NLGS solver applied to the 30 disk sample.

ω	1	5	10
NLGS iterations	10	8	10
NLGS-Jac iterations	10	50	∞
Gap	$\approx 10^{-4}$	$\approx 10^{-4}$	≈ 0.7

Table 3

Convergence behavior of the modified NLGS solver applied to the 500 disk sample.

ω	0.1	0.5	1	2	3	4	5
NLGS iterations	90	29	20	15	13	11	11
NLGS-Jac iterations	90	31	23	18	17	20	∞
Gap	$\approx 10^{-4}$	$\approx 10^{-4}$	$\approx 10^{-4}$	$\approx 10^{-4}$	$\approx 10^{-4}$	$\approx 10^{-4}$	$\approx 10^{-2}$

**Fig. 18.** A 500 disk example – “floating” grains: one contact (gray), no contact (black).**Table 4**

Number of non zero elements for the sparse representation – 30 disk example.

Matrices	H	W	\widetilde{W}	R_1	L	U	Total (R_1, L, U)
Non zeros	502	1300	14,400	872	324	965	2161

Table 5

Number of non-zero elements for the sparse representation – 500 disk example.

Matrices	H	W	\widetilde{W}	R_1	L	U	Total (R_1, L, U)
Non zeros	9758 ^a	27,858	4,177,936	32,583	12,487	43,254	88,324

^a H is previously reduced.

matrix, $\|W\| = 20.4$. We can clearly observe the regularization effect due to the kernel contribution, because the number of NLGS iterations decreases as ω increases. But the modified solver does not converge for large ω values even if the number of iterations does not increase too much for small ω .

5.3. Sparse computing

5.3.1. Sparse representation of the nullspace basis

In [11] S. Le Borne proposes an efficient block algorithm that computes a sparse representation of the null space basis in almost optimal complexity. The aim is to construct a null space basis of the discrete divergence operator in the finite element context. In the discrete element context, since we use a velocity-impulse formulation that favors impulses as the main variables, we are interested in the null space of the H operator (in fact this is the H_f operator, but for simplicity we keep H in the following). The objective is to compute, or rather represent, $N = \text{Ker}H$ in a way that permits an efficient matrix-vector product Nx .

The approach is based on the QR factorization of a full rank matrix that we note H^T in the following. Indeed, let $H^T \in \mathbb{R}^{n \times m}$ ($n > m$) be a sparse full rank matrix, i.e. $\text{rank}(H^T) = m$, and let

$$H^T = QR = \begin{pmatrix} Y & N \end{pmatrix} \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \quad (20)$$

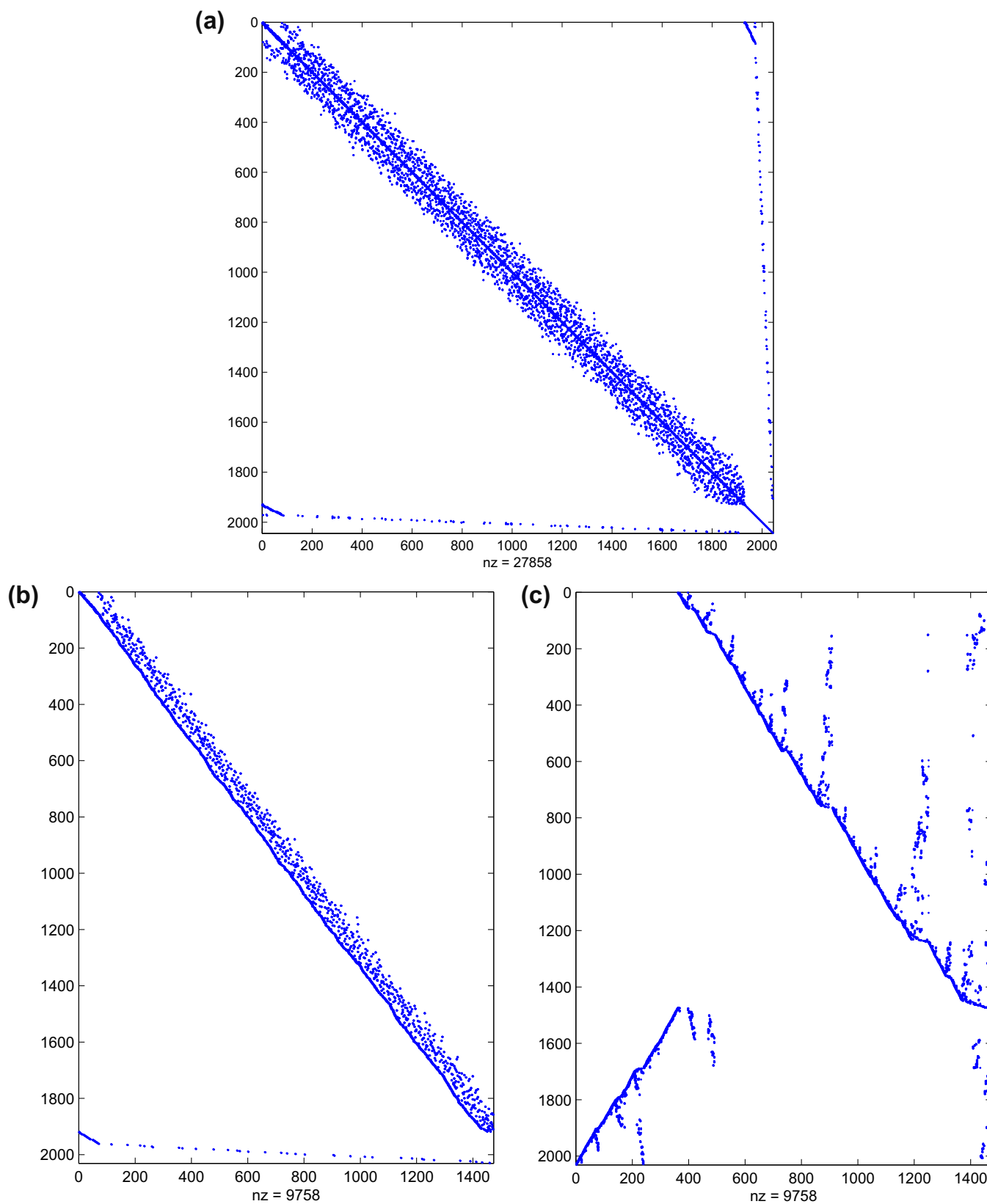


Fig. 19. Profiles of W, H^T of full rank, H^T after reordering.

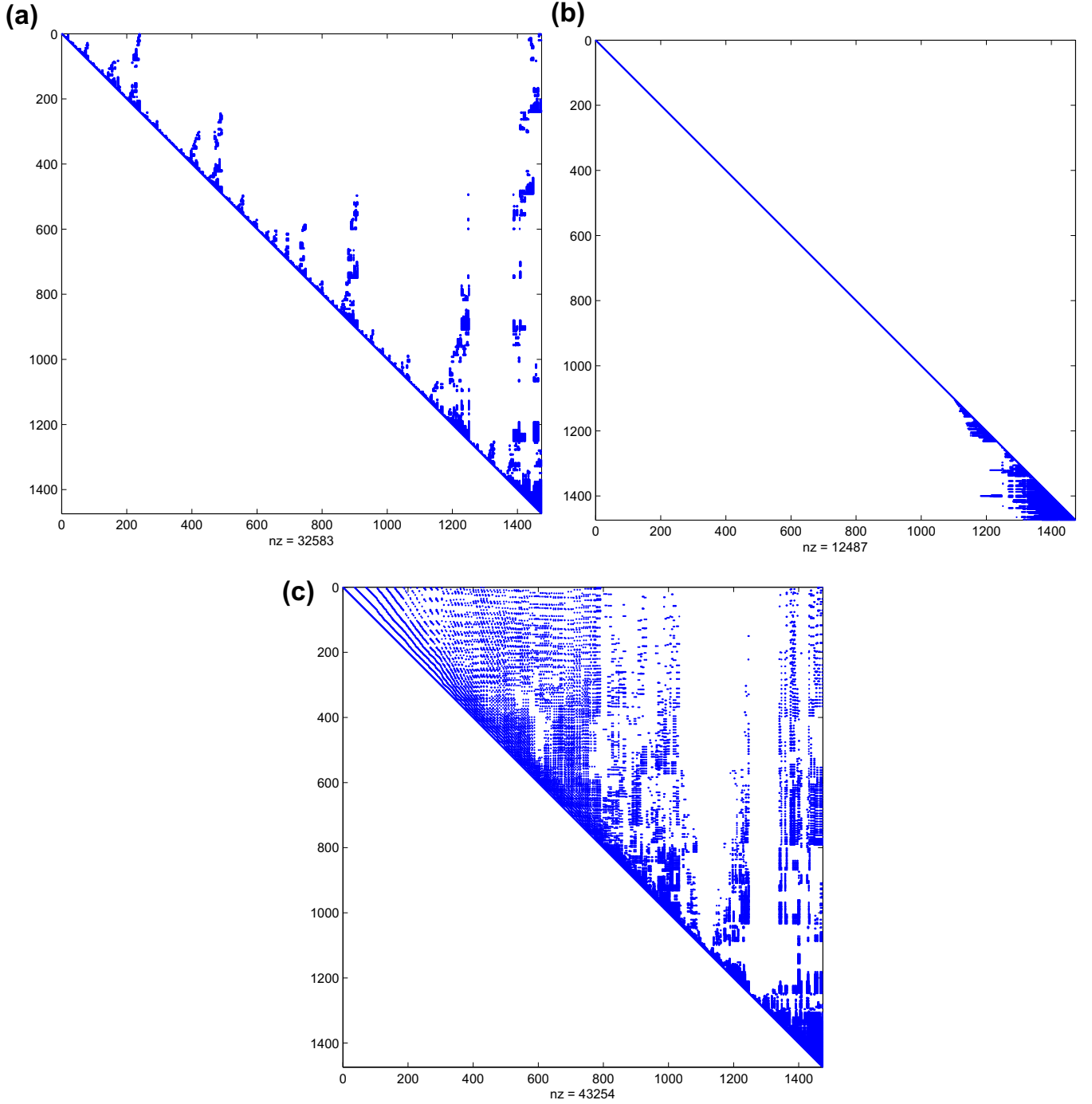


Fig. 20. Profiles of R, L and U .

be its full QR factorization as it is obtained through a sequence of m Householder transformations. The first m columns of Q , denoted by Y , span the range of H^T and are uniquely determined, whereas the last $n - m$ columns of Q , denoted by N , form a basis of the nullspace of H . Since we assume H^T has full rank, the product HH^T is positive definite and we can compute its Choleski factorization $HH^T = R_1^T R_1$ ($R_1 \in \mathbb{R}^{m \times m}$). The main result of [11] is summarized in the following theorem.

Theorem. [11] Let $H^T = \begin{pmatrix} H_1^T \\ H_2^T \end{pmatrix} \in \mathbb{R}^{n \times m}$ with $H_1^T \in \mathbb{R}^{m \times m}$, $H_2^T \in \mathbb{R}^{(n-m) \times m}$ and $\text{rank}(H^T) = m$. Let $R_1 \in \mathbb{R}^{m \times m}$ denote the upper triangular Choleski factor of HH^T , and let L, U denote the LU factors of $H_1^T - R_1$, assuming they exist (i.e. $H_1^T - R_1 = LU$). Then a null basis of H is represented by,

$$N = \begin{pmatrix} 0 \\ I_2 \end{pmatrix} + \begin{pmatrix} H_1^T R_1^{-1} - I_1 \\ H_2^T R_1^{-1} \end{pmatrix} L^{-T} U^{-T} H_2^T. \quad (21)$$

Table 6

Iterations and CPU times – 500 disk example.

Method or phase	NLGS (full)	NLGS-Jac (full)	Compute Z (sparse)	Compute DZ (sparse)	NLGS Z (sparse)	Total (sparse)
Iterations	20	23			23	
CPU time (s)	25.7	33.2	0.01	1	7	8.01

The sparse and implicit representation of the null space basis consists of the sparse matrices R_1, L, U . This result is obtained after ordering techniques that preserve at best the sparsity of the initial matrix H^T . A *nested dissection ordering* [22] is performed for the columns of H^T starting from the matrix graph of HH^T . A specific row ordering is performed in order to simplify the LU factorization of $H_1^T - R_1$.

The storage gain may be easily illustrated on the 30 disk example for which the H matrix has full rank. Table 4 gives the number of non-zero elements of the different matrices. The number of non-zero elements for \tilde{W} is equal to 120^2 (full matrix). The sparse representation (R_1, L, U) requires only 2161 non-zero elements to compare to 1300 for the W matrix. An evaluation of the CPU times is not relevant because the example is too small.

The more restrictive assumption for applying this procedure is the full rank of H^T ; this restriction is overcome in the next section.

5.3.2. Full rank and floating grains

The procedure providing a sparse representation of the null space runs only if the H matrix has full rank. For example, it fails on the 500 disks sample for which $\text{rank}(H^T) = 1487$ ($H \in \mathbb{R}^{1500 \times 2044}$). Indeed, $m = 1500$, which corresponds to 500 bodies (3 degrees of freedom per disk). The difference between 1500 and 1487 represents the local degrees of freedom of “floating” disks, more precisely 7 disks (gray in Fig. 18) having a single contact and 1 degree of freedom and 2 disks (black in Fig. 18) having no contact and 3 degrees of freedom, then $7 \times 1 + 2 \times 3 = 13$ degrees of freedom. Such the grains and the associated contacts do not contribute to the null space of H^T . A possible strategy consists of eliminating in H^T the degrees of freedom of the floating grains and the “single” contacts. Then H^T is reduced to a full rank matrix belonging to $\mathbb{R}^{1473 \times 2030}$.

Table 5 gives the non-zero elements in the different matrices we have to store. A full storage of the \tilde{W} matrix would require around 4 millions non-null terms whereas storage of the sparse matrices R_1, L and U does not reach 90,000 terms for around 28,000 with the W matrix. The profiles of the different matrices are provided in Figs. 19 and 20.

We can finally evaluate the CPU time gain obtained by the sparse computing strategy in comparison with a full storage. Table 6 provides the number of iterations for the standard NLGS and NLGS-Jac solvers (as previously mentioned in Table 3) and the number of iterations required by the NLGS-Z algorithm based on the NLGS-Jac scheme and the sparse representation of the null space; this number of iterations is equal to 23. The cost of the sparse computing strategy has to include the two preliminary phases, the computation of the sparse representation of Z (very quick) and the computation of the diagonal part of Z . In conclusion the total time is four times shorter than for the full storage method.

6. Conclusions, prospects and speculations

The present exploratory investigation shows the links between the force indetermination in the granular packings and the behavior of the NLGS algorithm. It proposes a new avenue of research consisting of dealing directly with the null space of the Delassus operator. A prediction–correction scheme was found to be a fast numerical algorithm. On the other hand, a drift in the interpenetration between grains occurs during the process simulation. This interpenetration may be controlled by injecting an elastic contribution in the prediction stage. The proportion of this elastic contribution remains an open question, certainly depending on the problem type. The present elastic contribution in the prediction stage is only a stopgap solution. A first challenge is to find a best solution than an elastic contribution for controlling the drift of the interpenetration. A deformable particle model is an alternative which may be explored with a method of floating frame of reference [23]. The method of a floating frame of reference (a brief overview was given in [24]) is a special type of a general family of so called corotational methods. A more prospective way would be to separate, in the impulse network, a dynamic contribution with eventual shocks from a quasi-static contribution belonging to the null space and slowly evolving. In this case the temporal integration scheme has to be revised by including a temporal multiscale approach. Such a splitting of impulsive and non-impulsive episodes is studied in [25]. A more attainable objective would be to implement the present strategy in a general discrete element software program to evaluate its performance on large-scale granular systems. However this may be too costly in term of computational time. This strategy could be then restricted to some time steps where the convergence fails.

References

- [1] J.J. Moreau, Numerical aspects of sweeping process, *Comput. Methods Appl. Mech. Eng.* 177 (1999) 329–349.
- [2] M. Jean, The non-smooth contact dynamics method, *Comput. Methods Appl. Mech. Eng.* 177 (1999) 235–257.
- [3] F. Jourdan, P. Alart, M. Jean, A Gauss–Seidel like algorithm to solve frictional contact problems, *Comput. Methods Appl. Mech. Eng.* 155 (1–2) (1998) 31–47.
- [4] G.D.R. MiDi, On dense granular flows, *Eur. Phys. J. E: Soft Matter Biol. Phys.* 14 (4) (2004) 341–365.

- [5] M. Renouf, D. Bonamy, F. Dubois, P. Alart, Numerical simulation of two-dimensional steady granular flows in rotating drum: on surface flow rheology, *Phys. Fluids* 17 (10) (2005) 103303.
- [6] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1) (1979) 47–65.
- [7] J.J. Moreau, Facing the plurality of solutions in non smooth mechanics, in: *Nonsmooth/Nonconvex Mechanics with Applications in Engineering*, Editions Ziti, Thessaloniki, 2006, pp. 3–12 (Proc. Internat. Conference in Memoriam of P.D. Panagiotopoulos).
- [8] M. Renouf, F. Dubois, P. Alart, Numerical investigations of fault propagation and forced-fold using a non smooth discrete element method, *Eur. J. Comput. Mech.* 15 (5) (2006) 549–570.
- [9] P. Alart, D. Iceta, D. Dureisseix, A nonlinear domain decomposition formulation with application to granular dynamics, *Comput. Methods Appl. Mech. Eng.* 205–208 (2012) 59–67.
- [10] T. Koziara, N. Bićanić, A distributed memory parallel multibody contact dynamics code, *Int. J. Numer. Methods Eng.* 87 (1–5) (2011) 437–456.
- [11] S. Le Borne, Block computation and representation of a sparse nullspace basis of a rectangular matrix, *Linear Algebra Appl.* 428 (11–12) (2008) 2455–2467.
- [12] G. Love, T. Laursen, Improved implicit integrators for transient impact problems – dynamic frictional dissipation within an admissible conserving framework, *Comput. Methods Appl. Mech. Eng.* 192 (19) (2003) 2223–2248.
- [13] Y. Ayyad, M. Barboteu, Formulation and analysis of two energy-consistent methods for nonlinear elastodynamic frictional contact problems, *J. Comput. Appl. Math.* 228 (1) (2009) 254–269.
- [14] V. Acary, Projected event-capturing time-stepping schemes for nonsmooth mechanical systems with unilateral contact and coulombs friction, *Comput. Methods Appl. Mech. Eng.* 256 (2013) 224–250.
- [15] C. Gear, B. Leimkuhler, G. Gupta, Automatic integration of Euler–Lagrange equations with constraints, *J. Comput. Appl. Math.* 12–13 (1985) 77–90.
- [16] L. Lecornu, Sur la loi de coulomb, *C. R. Acad. Sci. Paris* 140 (1905) 847–848.
- [17] F. Génot, B. Brogliato, New results on Painlevé paradoxes, *Eur. J. Mech. A/Solids* 18 (4) (1999) 653–677.
- [18] T. Unger, J. Kertész, D.E. Wolf, Force indeterminacy in the jammed state of hard disks, *Phys. Rev. Lett.* 94 (2005) 178001.
- [19] J.-N. Roux, G. Combe, Quasistatic rheology and the origins of strain, *C. R. Phys.* 3 (2) (2002) 131–140.
- [20] M. Jean, *Numerical Simulation of Granular Materials*, ISTE, 2010. pp. 149–315.
- [21] E. Azéma, F. Radjaï, R. Peyroux, F. Dubois, G. Saussine, Vibrational dynamics of confined granular materials, *Phys. Rev. E* 74 (2006) 031302.
- [22] J. Stern, S. Vavasis, Nested dissection for sparse nullspace bases, *SIAM J. Matrix Anal. Appl.* 14 (3) (1993) 766–775.
- [23] A. Lozovski, L. Daridon, F. Dubois, S. Pagano, The method of a floating frame of reference for non-smooth contact dynamics, *Tech. rep.*, <<http://hal.archives-ouvertes.fr/hal-00782620>>.
- [24] A.A. Shabana, O.A. Bauchau, G.M. Hulbert, Integration of large deformation finite element and multibody system algorithms, *Trans. ASME-V – J. Comput. Nonlinear Dyn.* 2 (4) (2007) 351–359.
- [25] T. Schindler, V. Acary, Time stepping schemes for nonsmooth dynamics based on discontinuous Galerkin methods: definition and outlook, *Math. Comput. Simul.*, 2012, <http://dx.doi.org/10.1016/j.matcom.2012.04.012>, URL: <<http://www.sciencedirect.com/science/article/pii/S0378475412001231>>.