



HAL
open science

Towards an augmented domain decomposition method for nonsmooth contact dynamics models

Vincent Visseq, Pierre Alart, David Dureisseix

► **To cite this version:**

Vincent Visseq, Pierre Alart, David Dureisseix. Towards an augmented domain decomposition method for nonsmooth contact dynamics models. *Computational Particle Mechanics*, 2014, 1 (1), pp.15-26. 10.1007/s40571-014-0005-8 . hal-00998116v1

HAL Id: hal-00998116

<https://hal.science/hal-00998116v1>

Submitted on 23 Mar 2016 (v1), last revised 14 Jun 2017 (v2)

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.

Towards an augmented domain decomposition method for nonsmooth contact dynamics models

Vincent Visseq · Pierre Alart · David Dureisseix

Abstract This paper explores the numerical performances of algorithms enriched by an augmented interface problem in a domain decomposition method dedicated to nonsmooth dynamic systems. Starting from simulations on a single time step, different algorithms are tested on moderate size samples. The analysis of the results leads to an incomplete resolution strategy for solving a time-evolution problem.

Keywords Discrete elements · Multibody · Parallel computing · Augmented algorithm · Distributed memory · Granular material · LMGC90

Mathematics Subject Classification 65M55 · 70F35 · 70F40 · 70E55

1 Introduction

We are concerned herein in dense discrete systems with a potentially large number of bodies and of nonsmooth interac-

tions between them (mainly frictional contact), for which granular materials are the main application. The difficulties for experiments to gather data at the microscale level (the scale of the grains), and for the comprehension of the involved phenomena, lead to extensive use of numerical simulations as virtual tests. Simulation for models describing individually all the bodies (modeled for instance as rigid bodies with large displacements and rotations, with a dynamic evolution of movements and of the interaction network between bodies) leads to costly models with a large volume of results. As a first step, the examples in this article will deal with medium sized 2D granular packings.

We proposed in [24,25] a domain decomposition strategy for granular media based on previous works [2,21] dedicated to nonsmooth discrete systems. Such a domain decomposition is illustrated in Fig. 1. We studied in [24] two domain decomposition methods (DDM) as a support for distributed memory parallelization with message passing. We analyzed the parallel performances for large scale problems on a supercomputing architecture according to the literature [6,16,23]. As underlined in [3] the efficiency of the Domain Decomposition methods in the context of multiprocessor computations is well established from theoretical and practical standpoints when dealing with a linear system derived from a discretization of a continuous problem [10,17]. The scalability may be proved theoretically when a coarse problem is added to the preconditioner of a conjugate gradient algorithm applied to the interface problem [18]. Enrichment of the coarse problem turns out to be mandatory to improve the convergence for different situations such as 3D [12], plate bending [19] or dynamic problems [11]. We proposed in [3] a first extension of such an enrichment for nonsmooth dynamical problems with an analytical study on very small examples.

To assess the local dynamical problem, the reformulation of Newton–Euler equations in terms of measurable differen-

V. Visseq
Department of Computer Science (DIKU), University
of Copenhagen, 2100 Copenhagen, Denmark

P. Alart (✉)
Laboratoire de Mécanique et Génie Civil (LMGC),
CNRS UMR 5508, Université Montpellier 2,
34095 Montpellier Cedex 5, France
e-mail: Pierre.Alart@univ-montp2.fr

P. Alart
Laboratoire de Micromécanique et d'Intégrité des Structures
(MIST), IRSN, CNRS, Université Montpellier 2,
Montpellier Cedex 5, France

D. Dureisseix
Laboratoire de Mécanique des Contacts et des Structures
(LaMCoS), INSA-Lyon, CNRS UMR 5259, Université de Lyon,
69621 Villeurbanne, France

tial inclusions (so that discontinuities are taken into account when two bodies collide), the time integration of those equations over the time slab $[t^i, t^f]$ leads to a velocity–impulse based formulation of the dynamics of the rigid body collection [20]:

$$MV - R = R^d, \quad (1)$$

where the prescribed right-hand side is $R^d = R^D + MV^i$. V , or V^f , is the assembly of the velocity of the grains defined at the final instant t^f ; it contains the translational degrees of freedom (dof), and the rotational ones, in the inertia eigenbasis frame of each grain; the exponent is suppressed for underlining that these variables are unknowns, as R in the left-hand side. R is the resultant impulse on the grains due to interactions with other grains and R^D are the external prescribed impulses. V^i denotes known quantities of the previous time step or at the initial instant t^i . The matrix M contains both the mass (for the translational dof) and the inertia (for the rotational dof).

Unilateral contact and friction laws between particles are naturally expressed in contact frames. Mapping between particles dof and contact dof is achieved in a matricial way as $v = H^T V$ and $R = Hr$, with v the assembly of the relative contact velocities and r the assembly of the contact impulses. H is a predicted compatibility operator computed at the initial instant or at a middle instant. The dynamic equations (1) are then condensed to contact dof as,

$$\begin{cases} Wr - v = -v^d \\ \mathcal{R}(v, r) = 0 \end{cases}, \quad (2)$$

with $W = H^T M^{-1} H$, $v^d = H^T M^{-1} R^d$ and $\mathcal{R}(v, r) = 0$ as the formal notation of contact laws. The difficulty to solve the problem (2) is at least two-folds: on one hand, the number of unknowns (number of interaction quantities r and v) may be large and the Delassus operator W is not well conditioned (it is a priori non invertible). On the other hand, the constitutive relations are nonsmooth (e.g. they are non linear and not differentiable). To address the nonsmoothness issue, the NonSmooth Contact Dynamics (NSCD) method with a nonlinear Gauss-Seidel (NLGS) solver [14, 15, 20] are used. To address the large size of the problem, parallel computing can be used [16, 23], and in this article, we rely on a substructuring approach [3, 13, 24]. Nevertheless, since we focus herein on dedicated strategies, we only consider moderate size samples, and a small number of subdomains for the proposed examples.

Section 2 is dedicated to the domain decomposition in the context of granular media and to the formulation of the generic solver. Section 3 presents the augmented interface problem for enriching one of the two stages of the algorithm. Several algorithmic strategies are developed and tested in Sect. 4. Finally a fully multiscale resolution is proposed in

Sect. 5. For an evolution granular problem, this amounts to an incomplete resolution at each time step according to a separation of the scales, macro scale for the interface, micro scale inside the subdomains. After the conclusions in the last section, some technical aspects on the parallel solver are developed in ‘‘Appendix’’.

2 Domain decomposition

2.1 Domain partitioning

Domain partitioning of a discrete element collection is, at each time step or at a user-defined frequency, a partitioning of an interaction graph. The interaction graph consists in nodes associated to grains and edges associated to interactions.

The proposed DDM assumes a partitioning similar to [25] and dual to the partitioning proposed in [13]: one distributes the interactions among the subdomains. A convenient way is to distribute the middle points between the centers of mass of interacting particles over the subdomains using a regular cartesian grid with as many cells as subdomains (n_{sd} denotes the number of subdomains). It has been named as the ‘box method’ [6]. With such a choice, if a grain (or a particle) indexed with i supports interactions with m_i neighboring subdomains, m_i is called its *multiplicity number*. If $m_i > 1$ the particle \mathcal{S}_i belongs both to the subdomain and to the interface between subdomains (cf. Fig. 1). Therefore, a boolean matrix B_E selecting kinematic degrees of freedom of grains belonging to subdomain E allows to define the grain velocities in this subdomain as,

$$V_E = B_E V \quad (3)$$

With this definition of the mapping matrix, one can check that the diagonal matrix of the grain multiplicities is $\sum_{E=1}^{n_{sd}} B_E^T B_E$.

2.2 FETI-like domain decomposition: NSCDD algorithm

The present DDM considers a non overlapping partition of the sample.

For consistency with the rigid model of the grains, the masses and moments of inertia are distributed among the neighboring subdomains according to their multiplicity number. More precisely the distribution of masses and inertia is an algebraic partitioning and not a geometrical partitioning. This leads to a partition of unity over the inertia parameters, as,

$$\tilde{M}_E = B_E D M B_E^T, \quad (4)$$

with,

$$D_{kl} = \begin{cases} 0 & \text{if } k \neq l \\ 1/m_i & \text{if } k = l \end{cases} \quad (5)$$

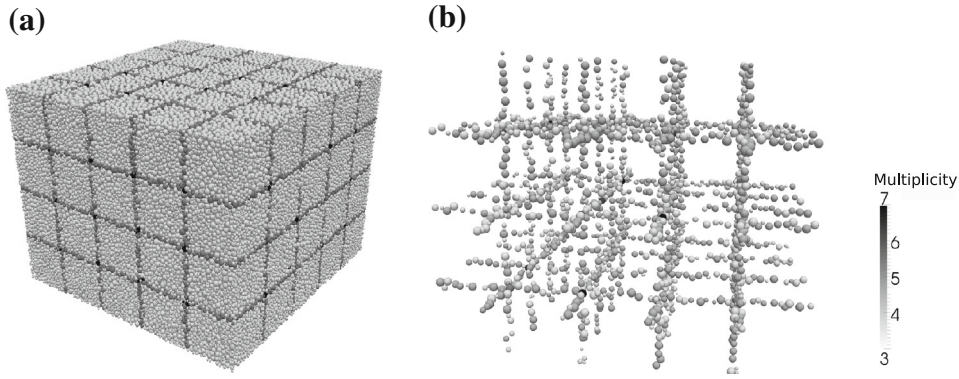


Fig. 1 Domain partitioning of a 200,000 sphere sample (100 subdomains). **a** and underlying structure of corner grains. **b** For convenience **a** gray scale is restricted to [1,4] and **b** gray scale is restricted to [3,7]; $m = 1$: inner grain; $m = 2$: face grain; $m > 2$: corner grain

for entries k related to the grain \mathcal{S}_i . The partition of unity property reads: $M = \sum_{E=1}^{n_{sd}} B_E^T \tilde{M}_E B_E$.

This topic is investigated in details in [25]. In each subdomain E , the problem is identical to the global one (with the subscript E), provided that a term arising from the inter-grain interface is added. It can be built from the interconnecting condition (on the velocity jumps of boundary grains) that has been added to ‘glue’ neighboring subdomains, where $A_{\Gamma E}$ is a signed boolean matrix with a finite rotation, to map the grain velocities V_E to the global coordinate basis into which the null velocity jump on the grain interface is expressed,

$$\sum_{E=1}^{n_{sd}} A_{\Gamma E} V_E = 0 \quad (6)$$

Γ denotes the global interface of all the interface grains. Formally the previous summation is performed on all the subdomains; in a practical way, for a given grain interface, only the neighboring subdomains have to be considered. We then obtain a FETI-like formulation [5,7,9] for the reference problem using a multiplier field F_Γ and the notation $\hat{A}_{\Gamma E}^T = H_E^T \tilde{M}_E^{-1} A_{\Gamma E}^T$, $\tilde{W}_E = H_E^T \tilde{M}_E^{-1} H_E$,

$$\left. \begin{aligned} \tilde{W}_E r_E - v_E - \hat{A}_{\Gamma E}^T F_\Gamma &= -v_E^d \\ \mathcal{R}(v_E, r_E) &= 0 \end{aligned} \right\} E = 1, \dots, n_{sd} \quad (7)$$

$$\sum_{E=1}^{n_{sd}} A_{\Gamma E} V_E = 0$$

The reduced problem on (r_E, v_E, F_Γ) , with the notations $\hat{f} = \sum_E A_{\Gamma E} \tilde{M}_E^{-1} R_E^d$, $v_E^d = H_E^T \tilde{M}_E^{-1} R_E^d$ and $X = \sum_E A_{\Gamma E} \tilde{M}_E^{-1} A_{\Gamma E}^T$ and a partial condensation of the problem, reads,

$$\left. \begin{aligned} \tilde{W}_E r_E - v_E - \hat{A}_{\Gamma E}^T F_\Gamma &= -v_E^d \\ \mathcal{R}(v_E, r_E) &= 0 \end{aligned} \right\} E = 1, \dots, n_{sd} \quad (8)$$

$$X F_\Gamma - \sum_{E=1}^{n_{sd}} \hat{A}_{\Gamma E} r_E = \hat{f}$$

One easily reformulates the interface equation as an incremental problem [25]: if F_Γ is associated to a velocity field V with the dynamical equations, and if this last field is not continuous at the interface, the correction of the impulse field F_Γ is ΔF_Γ such that

$$X \Delta F_\Gamma = \sum_{E=1}^{n_{sd}} A_{\Gamma E} V_E = [V]_{|\Gamma} \quad (9)$$

the last term being the residual on the interface, i.e. the velocity jump $[V]_{|\Gamma}$. As for many domain decomposition approaches, the goal is to be able to localize the same typical problem that is under consideration on each subdomain independently, while designing a suited coupling recovery algorithm between subdomains, i.e. on the interface.

Here, the formulation described in Algorithm 1 has been implemented into the LMGC90 platform [8] for time-evolution problems. At each new time step of the incremental solving procedure, the mapping H and the contact graph have to be updated within a contact detection phase. Eventually, the domain could also be repartitioned according to the new contact graph.

3 Augmented interface problem

The NSCDD method exhibits good parallel efficiency for dense granular dynamics problems [24]. We exemplified in [25] that the global behavior and the micromechanical structure of large-scale dense granular systems under biaxial loading are not disturbed by NSCDD substructuring. Moreover, extensibility is recovered when the number of particle dof is large compared to the number of interface dof. Nevertheless the number of iterations increases with the number of particles (both for sequential and parallel algorithms). This phenomenon is related to the nonsmoothness of the considered interactions. Indeed, because of the nonsmooth relations between the internal dof of a subdomain, no condensation

Algorithm 1 NonSmooth Contact Domain Decomposition (NSCDD)

for $i = 1, \dots, N$ **do**
 Contact detection (eventually parallelized) and possible new decomposition of the domain
 Initialize unknowns at time t_i : (r_E, v_E, F_Γ)

while (convergence criterion not satisfied) **do**
 Stage 1: In parallel for $E = 1, \dots, n_{sd}$
 Disassemble interface impulses F_Γ into local impulses F_E
 Compute the velocity $\tilde{v}_E^d = v_E^d - H_E^T \tilde{M}_E^{-1} F_E$
 Compute $(\tilde{r}_E, \tilde{v}_E)$ with n_{GS} non-linear Gauss-Seidel (NLGS) iterations on:

$$\begin{cases} \tilde{W}_E \tilde{r}_E - \tilde{v}_E = -\tilde{v}_E^d \\ \mathcal{R}(\tilde{v}_E, \tilde{r}_E) = 0 \end{cases}$$

 Update $(r_E, v_E) \leftarrow (\tilde{r}_E, \tilde{v}_E)$
 Compute R_E and correct the velocity on interface grains to get $A_{\Gamma E} \tilde{V}_E$
 Stage 2: In sequential, but may be parallelized,
 Compute ΔF_Γ as: $X \Delta F_\Gamma = \sum_{E=1}^{n_{sd}} A_{\Gamma E} \tilde{V}_E$ and update interface impulses F_Γ

end while

Update grain positions in parallel
end for

process on the interface is allowed. The NSCDD method defines a quasi-diagonal linear interface problem without coupling the different interfaces of a subdomain.

To tackle this phenomenon, the developments reported in [3] proposed to introduce a numerical tangent search direction for contact unknowns $(\tilde{r}_E, \tilde{v}_E)$, once an iterate (r_E, v_E) is obtained from the nonsmooth reduced dynamics, satisfying the dynamic equations and verifying

$$\begin{aligned} \tilde{R}_E &= G_E \tilde{r}_E \text{ et } \tilde{v}_E = G_E^T \tilde{V}_E, \\ (\tilde{r}_E - r_E) + \ell_E (\tilde{v}_E - v_E) &= 0, \end{aligned} \quad (10)$$

with ℓ_E a *numerical scalar parameter* of the method (homogeneous to a mass) and G_E a *compatibility operator* that can be different from the predicted operator H_E . Note that ℓ_E could be defined as a diagonal matrix with tuned coefficients for each contact. This ‘tangent’ numerical search direction is not a physical one, due to the nonsmooth nature of the frictional contact behavior. Choosing this search direction is therefore not trivial and different possibilities can be tested.

Herein, we study the properties of the augmented, or enriched, interface problem with respect to the chosen compatibility operator for the tangent search direction. A generic compatibility operator is $G_E = H_E$, i.e. the compatibility operator used for solving the nonsmooth dynamics inside the subdomains given by the contact detection phase. The asymptotic study done in [3] shows that—at least without friction—the optimal compatibility operator is the restriction of H_E to the only active ($r > 0$) contacts, but this optimal operator is a priori unknown.

For enrichment of the NSCDD method, $(\tilde{r}_E, \tilde{v}_E)$ must ensure compatibility of velocity across the interface. The substructured dynamics for these quantities reads

$$\tilde{M}_E \tilde{V}_E - H_E \tilde{r}_E = R_E^d - A_{\Gamma E}^T F_\Gamma. \quad (11)$$

Substituting \tilde{r}_E from Eq. (10), we get

$$\tilde{M}_{\ell, E} \tilde{V}_E - G_E (r_E + \ell_E v_E) = R_E^d - A_{\Gamma E}^T F_\Gamma, \quad (12)$$

with $\tilde{M}_{\ell, E} = \tilde{M}_E + \ell_E K_E$ and $K_E = G_E G_E^T$; this last matrix contains information of the contact network thanks to connectivity matrices G_E and G_E^T . As for the generic NSCDD method, dynamic equations are condensed on the interface and the continuity equation $\sum_E A_{\Gamma E} \tilde{V}_E = 0$ allows to express the enriched interface equation for F_Γ :

$$X_\ell F_\Gamma = \sum_{E=1}^{n_{sd}} A_{\Gamma E} \tilde{M}_{\ell, E}^{-1} \left[R_E^d + G_E (r_E + \ell_E v_E) \right], \quad (13)$$

with the *enriched interface operator*,

$$X_\ell = \sum_{E=1}^{n_{sd}} A_{\Gamma E} \tilde{M}_{\ell, E}^{-1} A_{\Gamma E}^T. \quad (14)$$

Discussion The NSCDD enrichment leads to a coupled interface problem. Solving the NSCDD enriched interface problem is time consuming as it requires to solve a global linear problem on the whole domain viewed as a lattice structure with the same connectivity as the contact graph. Due to the distribution of the database per subdomain, and to avoid a costly direct solve, we choose to design a parallel conjugate gradient algorithm close to the one used in classical distributed parallel approach (cf. Appendix).

4 Algorithmic strategies for the enriched NSCDD

To study the properties of the augmented interface problem, at a first hand, different solving algorithms are compared on a single time step, for cases without friction ($\mu = 0$) and with a dry friction coefficient $\mu = 0.3$. Additionally, two compatibility operators are considered for the enriched interface problem:

- G_E such that $G_E r_E = H_E r_E$; therefore, only active contacts are taken into account in building $K_E = G_E G_E^T$,
- G'_E such that $G'^T_E V_E = 0$; with taking into account only normal components of active contacts in building $K'_E = G'_E G'^T_E$.

To exhibit an eventual convergence rate acceleration for the enriched interface problem depending on parameter ℓ_E , two augmented algorithms are compared. The first one, named as ‘Fully enriched algorithm’—FEA, consists of a

direct extension of the generic Algorithm 1 for which the now enriched interface problem is solved after a single NLGS iteration ($n_{GS} = 1$) and the convergence is tested on the interface and inside the subdomains. This is summarized in Algorithm 2.

Algorithm 2 NSCDD ‘Fully enriched algorithm’—FEA

```

for  $k_1 = 1, \dots, It_{max1}$  do
  Stage 1: In parallel for  $E = 1, \dots, n_{sd}$ 
    NLGS iteration
  Stage 2: On the interface
    Solving augmented interface problem
  if (Convergence within the body and on the interface) then
     $It_1 = k_1$  and terminate for loop
  end if
end for

```

The second studied algorithm, named as ‘Relaxed enriched algorithm’—REA in the following, consists at a first step in iterating on the contacts (stage 1) then on the interface (stage 2) until convergence is reached on the interface. The convergence test is thus restricted to the interface, so the convergence criterium is relaxed. The second step consists in iterating only on contacts (independently for each subdomain), until convergence within the body. Thus this second step refines the solution at the micro scale. This is summarized on Algorithm 3. It allows to focus on the convergence rate of the interface problem depending on the enrichment parameter ℓ_E , more precisely the dimensionless parameter $\ell = \frac{\ell_E}{m_E}$, where m_E is a reference mass.

For both algorithms, the compatibility operator G_E or G'_E is redefined after each NLGS stage, to update the contact status.

Algorithm 3 NSCDD ‘Relaxed enriched algorithm’—REA

```

for  $k_2 = 1, \dots, It_{max2}$  do
  Stage 1: In parallel for  $E = 1, \dots, n_{sd}$ 
    NLGS iteration
  Stage 2: On the interface
    Solving augmented interface problem
  if (Convergence on the interface) then
     $It_2 = k_2$  and terminate for loop
  end if
end for
for  $k_3 = 1, \dots, It_{max3}$  do
  Stage 1: In parallel for  $E = 1, \dots, n_{sd}$ 
    NLGS iteration
  if (Convergence within the body) then
     $It_3 = k_3$  and terminate for loop
  end if
end for

```

4.1 Granular test case

In order to illustrate the study on a single time step, a limited size example is proposed.

Setup phase The sample is constituted with 730 disks previously packed in a rigid box whose walls are clusters of disks. The final state is obtained after a vertical gravity load g is prescribed until the sample is stabilized.

Prescribed loading The considered simulation consists in prescribing a rotation of the sample. This rotation is modeled with a rotated gravity vector as in Fig. 2: $g' = [\sin(\theta), -\cos(\theta)]^T \times \|g\|$.

Discussion Results on the number of iterations at convergence are collected in Fig. 3 for a domain decomposition with two subdomains using a partitioning grid (1×2). The number of iterations It_2 for the frictionless case (Fig. 3a, b):

- is constant for $\ell < 1$ (even with $\ell = 0$, i.e. a standard NSCDD interface problem), with $It_2 < It_1$,
- decreases for $1 < \ell < 100$,
- diverges for $\ell > 500$ (not depicted).

For the frictional case, similar results are obtained, though with a less decreasing trend for $1 < \ell < 100$. For all cases, $It_2 + It_3 < It_1$: algorithms FEA and REA are not equivalent with respect to the number of iterations. But globally the gain in terms of iteration number is too weak for compensating the cost of the enriched interface resolution.

For $\ell < 10$, the compatibility operator G'_E leads to similar results as for $K_E = G_E G_E^T$, but the number of iterations is more stable for larger values of ℓ . In the following, G'_E is therefore selected.

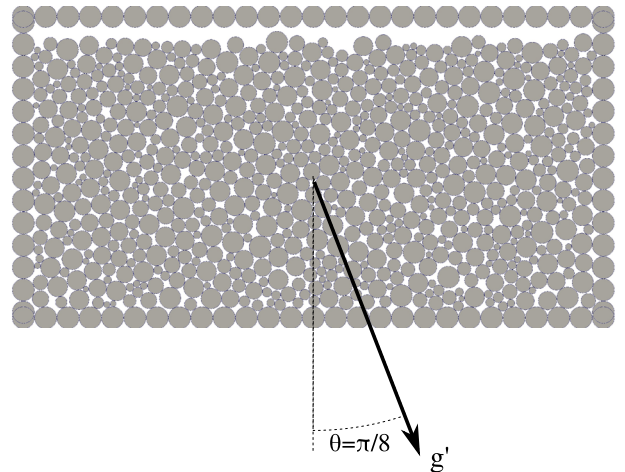


Fig. 2 Example of 730 disks with inclined gravity, one time step; geometry and prescribed loading

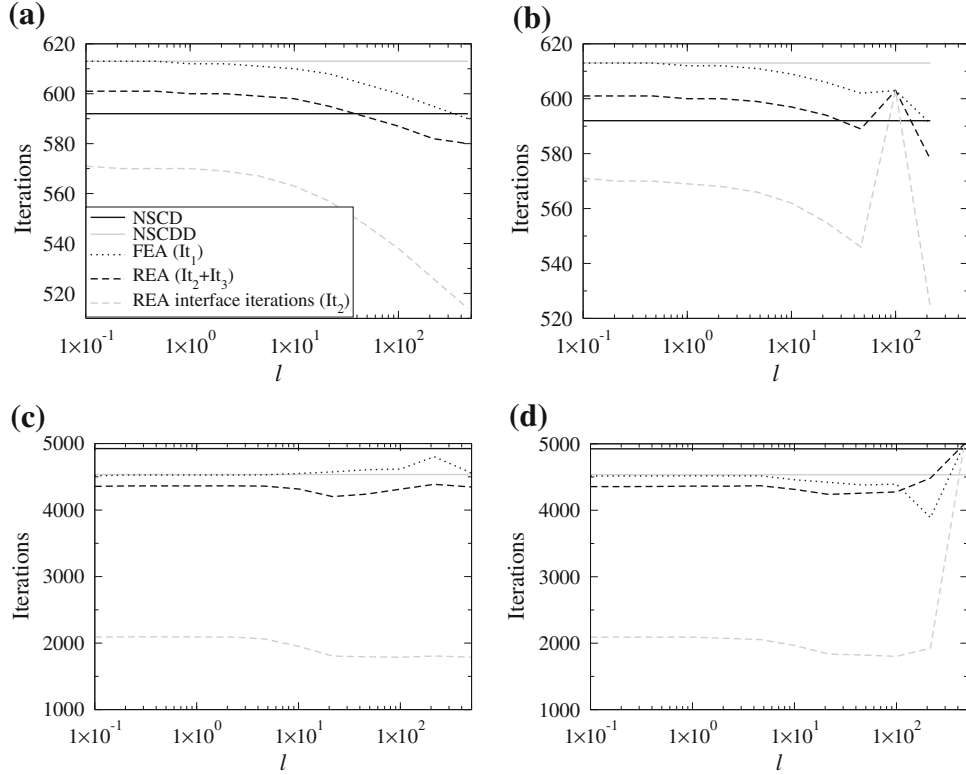


Fig. 3 Example of 730 disks with inclined gravity, one time step: number of iterations to converge as a function of ℓ . **a** frictionless contact ($\mu = 0$) and $K'_E = G'_E G_E^T$. **b** frictionless contact ($\mu = 0$) and

$K_E = G_E G_E^T$. **c** contact with friction ($\mu = 0.3$) and $K'_E = G'_E G_E^T$. **d** contact with friction ($\mu = 0.3$) and $K_E = G_E G_E^T$.

5 A fully multiscale resolution

5.1 Test on a full process (with or without friction)

Consider now the simulation of the behavior of the same granular test bed along a full time evolution process, for which the gravity vector $g(\theta)$ incrementally rotates from $\pi/8$ up to $-\pi/8$, as depicted in Fig. 4.

Figures 5 and 6 show that the two augmented algorithms do not allow a significant reduction of the number of iterations needed to converge, when compared to the reference algorithms NSCD (without substructuring) and NSCDD (without enrichment). Moreover, mean and maximal interpenetrations (a measure of the quality of the numerical solution produced) are larger (though still small when compared to the mean disk radius that was selected to be 1).

This trend is similar for cases without and with friction, Figs. 5 and 6, for a substructuring in two subdomains with a partitioning grid (1×2). Due to the additional cost of the augmented interface problem, the algorithms FEA and REA are inefficient for such a granular evolution process problem. Such a numerical behavior may be explained by the rigid nature of the particles and the non smoothness of the interactions. In other words a large-scale nonsmooth problem with exact steric exclusions cannot be correctly enough predicted

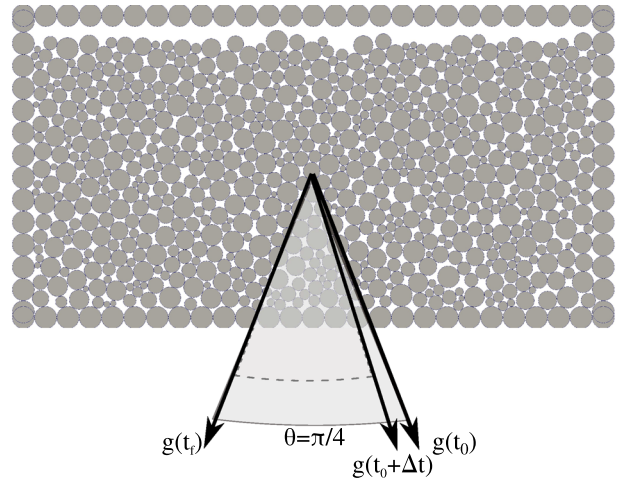


Fig. 4 Test with 730 disks under rotating gravity: evolution process during with the gravity vector incrementally rotates up to $\pi/4$, with 50 time steps

by a linear problem because the local non smooth corrections strongly perturb the global dynamics relayed by the interfaces. Except if we accept to solve coarsely the global interface problem at each time step, before correcting, once and for all, the non smooth local interactions. Hence the motivation of the following section.

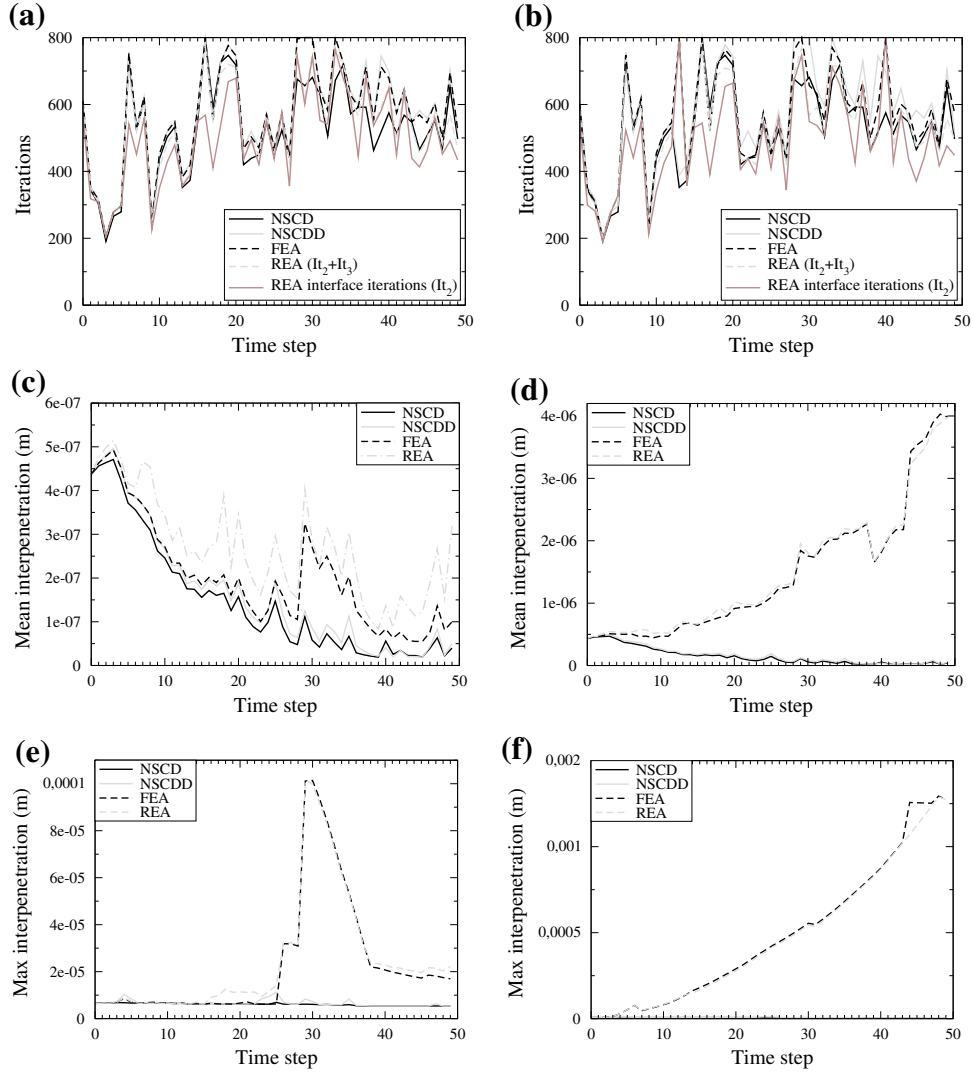


Fig. 5 Test with 730 disks and a rotating gravity, without friction ($\mu = 0$), and $K'_E = G'_E G_E^T$. Number of iterations to convergence as a function of time step for $\ell = 10$ (a) and $\ell = 100$ (b). Mean inter-

penetration as a function of time step for $\ell = 10$ (c) and $\ell = 100$ (d). Maximal interpenetration as a function of time step for $\ell = 10$ (e) and $\ell = 100$ (f)

5.2 Incomplete resolution

The proposal in this section is to combine (i) an explicit resolution of the (linear) interface problem at the subdomain scale (macro scale), based on the active contact network as stated at the beginning of the time step, with (ii) an implicit resolution of nonsmooth problems within each subdomain, for each contact (micro scale).

This strategy relies on the assumption that interface forces traducing the global behavior of the media evolve slower than local impulses ruled by nonsmooth dynamics. The works in [22] on bimodality of the contact network exemplify that the strong network is ruled by normal impulses in the contacts hardly involving tangential sliding.

It is then possible to choose a different compatibility operator G_E for determining $(\tilde{r}_E, \tilde{v}_E)$ for the different stages of

the augmented algorithms; a first selection for this operator is to choose G_E such that $G_E^T V_E = 0$, by selecting normal components of active contacts. The Algorithm 4, named as ‘Incomplete enriched algorithm’—IEA, is a proposal for the implementation of such a scheme, with an update stage of active contacts at the beginning of each time step, using a single NLGS iteration. In other words the IEA algorithm consists in restricting the first step of the REA algorithm to a single iteration ($It_{max2} = 1$).

5.3 Slow dynamic test

In order to test the algorithm IEA, the same problem of granular sample with 730 disks and rotation of gravity vector is reused. This test indeed belongs to the category of problems where the contact network is relatively persistent though the

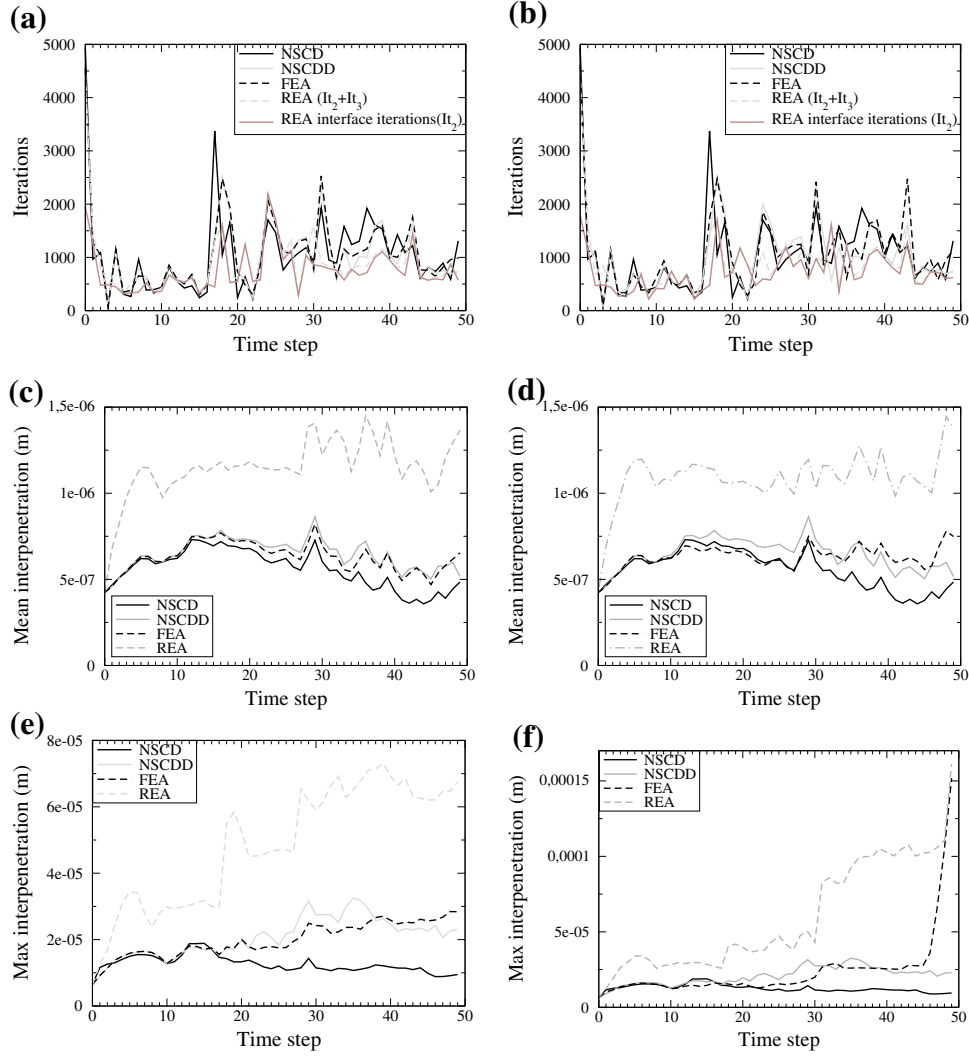


Fig. 6 Test with 730 disks and a rotating gravity, with friction ($\mu = 0.3$), and $K'_E = G'_E G'^T_E$. Number of iterations to convergence as a function of time step for $\ell = 10$ (a) and $\ell = 100$ (b). Mean interpenetration as a function of time step for $\ell = 10$ (c) and $\ell = 100$ (d). Maximal interpenetration as a function of time step for $\ell = 10$ (e) and $\ell = 100$ (f)

contact force distribution notably evolves. Therefore it suits the assumptions favorable to the incomplete solve strategy previously described. This incomplete solve requires also to assess the quality of the obtained solution, by checking a quality control indicator. This indicator is the mean or maximal interpenetration.

The obtained iteration numbers at convergence, i.e. It_4 in Algorithm 4, as well as the interpenetrations are depicted in Fig. 7, for the same partitioning grid for the subdomains (1×2), and for a friction coefficient $\mu = 0.3$.

Figure 7a compares the number of iterations for algorithm IEA with respect to the references (algorithms NSCD and NSCDD). A moderate reduction is obtained, with a non monotonous dependence on parameter ℓ . For readability reasons, only cases $\ell = 0, 10, 100$ are depicted.

Concerning mean and maximal interpenetrations, Fig. 7b, c depict a series of curves corresponding to $\ell \in [0, 10^5]$. These interpenetrations largely decrease with ℓ . For $\ell \in [0, 10^3]$ they significantly evolve with an increasing trend as the time steps are progressing, whereas for $\ell \in [10^4, 10^5]$ they are stabilizing after a reduced number of time steps. Nevertheless, these interpenetrations are larger than their counterparts for the algorithms NSCD and NSCDD (Fig. 6) but remain acceptable with respect to the mean radius of grains that was selected to 1.

Algorithm IEA is somehow a multiscale approach; concerning space using a domain decomposition method providing the subdomain scale and the grain scale, and also concerning time evolution using different time integrations depending on the spatial scale. The linear interface problem couples

etration as a function of time step for $\ell = 10$ (c) and $\ell = 100$ (d). Maximal interpenetration as a function of time step for $\ell = 10$ (e) and $\ell = 100$ (f)

Algorithm 4 NSCDD ‘Incomplete enriched algorithm’—IEA

```

for  $i = 1, \dots, N$  do {Loop on time steps}
  Eventual new subdomain decomposition
  Contact detection, setting of external impulses and prescribed
  velocities

  Explicit interface problem
  In parallel for  $E = 1, \dots, n_{sd}$ 
    Update the contact status
  On the interface
    Solving the augmented interface problem

  NSCD solves per subdomain
  for  $k_4 = 1, It_{max4}$  do
    In parallel for  $E = 1, \dots, n_{sd}$ 
      NLGS iteration
    if (Convergence within the body) then
       $It_4 = k_4$  and terminate for loop
    end if
  end for

  Compute velocities and position of the grains
end for
  
```

the whole set of subdomains, but is solved in an explicit manner, while nonsmooth problems per subdomains are iteratively solved to capture the local configuration changes. The correction along time steps of the interpenetration is allowed

with this semi-implicit strategy, though only an incomplete solve is performed at each time step (Fig. 8).

5.4 Dynamic flow test

We now consider a granular flow with an horizontal main velocity, and with a periodic boundary condition in the same direction. Thus the sample is sloped with an angle θ equal to $\pi/6$. Two subdomains are defined as previously, Fig. 10. This problem exhibits a large modification of the contact graph along the evolution process and a convection from the right to the left direction.

The evolution of the mean interpenetration in Fig. 9 is similar to the previous test with a decrease then a stabilization with respect to the ℓ factor. This is particularly true for the end of the process whereas the classification of the curves is not obvious in a first period. The maximal interpenetration evolves quite differently and sometimes up to values that are less acceptable than for the previous test. More precisely, for $\ell = 10^3$, the maximal interpenetration stabilizes to an excessive value whereas, for $\ell = 10^5$, the maximal interpenetration evolves with large variations but tends to decrease until to an acceptable value. Such a behavior is related to the evolution of the contact network with the flow. As an illustration in Fig. 10 the interpenetrations are generated nearby the interface because of the incomplete resolution, then they

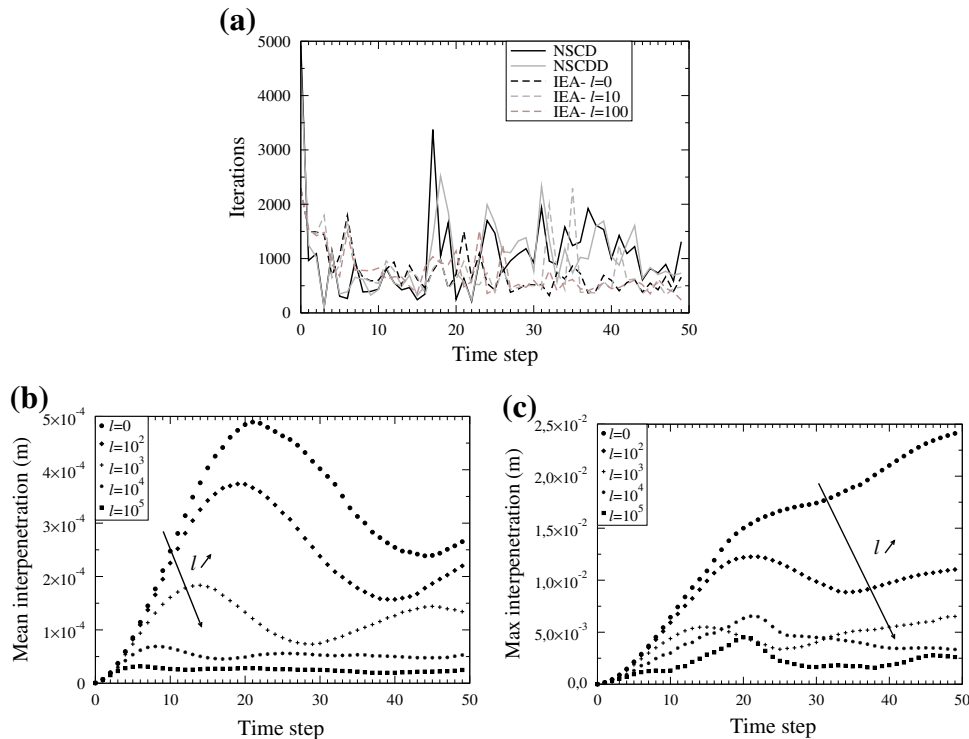


Fig. 7 Test with 730 disks, a rotating gravity and a friction coefficient $\mu = 0.3$; number of iterations to reach convergence (a), mean interpenetration (b) and maximal interpenetration (c) according to $\ell \in [0, 10^5]$

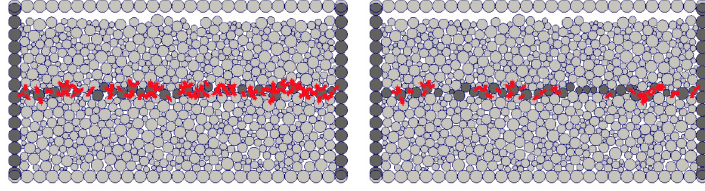


Fig. 8 Test of 730 disks under rotating gravity: **a** first and **b** final time step. The multiplicity is: 1 for a *gray particle* and 2 for a *dark particle*. Red (dark) segments indicate disk overlaps δ_α satisfying $\delta_n \geq 0.9 \max \delta_n$. (Color figure online)

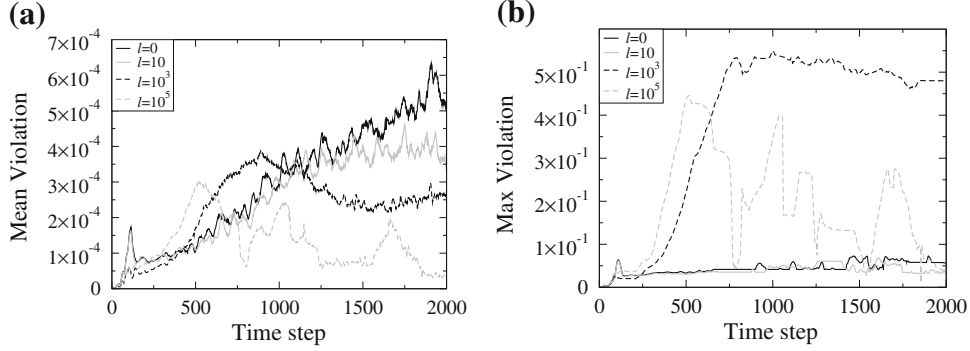


Fig. 9 Test of a 750 disk granular flow: evolution of **a** mean interpenetration and **b** maximal interpenetration according to $\ell \in [0, 10^5]$

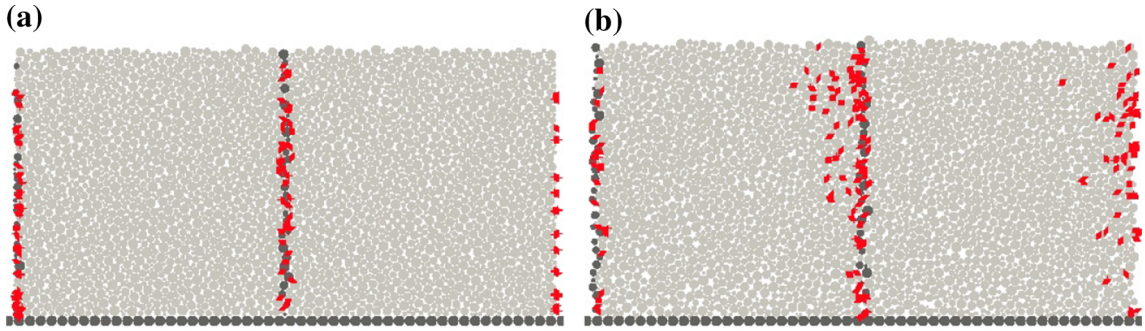


Fig. 10 Test of a 750 disks granular flow with $\ell = 10^3$: **a** beginning and **b** final time step (200). The multiplicity is: 1 for a *gray particle* and 2 for a *dark particle*. Red (dark) segments indicate disks overlap δ_α satisfying $\delta_n \geq 0.9 \max \delta_n$. (Color figure online)

migrate inside the subdomains. These interpenetrations hold as long as the contacts persist and are erased as soon as the contacts release. This process explains the high level and the oscillations of the maximal interpenetrations in a highly dynamic test.

For this test case, we moreover use a parallelization of the interface problem, as described in the following section.

5.5 Parallel resolution of the interface problem

The augmented interface problem (13) is a global problem involving the whole force network of the sample. Its structure is similar to the one of a linear elastic problem in quasi-static evolution. To solve efficiently this global problem, an iterative approach is an appealing alternative. Since matrix

X_ℓ is symmetric and definite positive, and since the data are distributed among the subdomains, the conjugate gradient algorithm is a suited choice.

Algorithm 5 proposes a detailed version of the augmented version of Algorithm 2 for one time step. This resolution involves two embedded domain decomposition methods:

- global iterations of NSCDD approach,
- parallel conjugate gradient on the augmented interface gluing problem.

The augmented interface problem cannot be solved with an incremental formulation as in the generic algorithm (Algorithm 1). The specific implementation of this parallel conjugate gradient for the granular interface problem on the

Algorithm 5 NSCDD—augmented version with two embedded DDM

while (convergence criteria in the body and on the interface are not satisfied) **do**

Stage 1: In parallel for $E = 1, \dots, n_{sd}$

Evaluation of $\tilde{v}_E^d = v_E^d - H_E^T \tilde{M}_E^{-1} F_E$

Compute (r_E, v_E) with n_{GS} NLGS iterations on:

$$\begin{cases} \tilde{W}_E r_E - v_E = -\tilde{v}_E^d \\ \mathcal{R}(v_E, r_E) = 0 \end{cases}$$

Stage 2: On the interface

while (convergence criteria of the conjugate gradient are not satisfied) **do**

Parallel iterates of the conjugate gradient algorithm on F_Γ :

$$X_\ell F_\Gamma = \sum_{E=1}^{n_{sd}} A_{\Gamma E} M_{\ell, E}^{-1} [R_E^d + H_E (r_E + \ell v_E)]$$

end while

end while

LMGC90 platform is recalled in Algorithm 6 as an “Appendix”. The overall stages are standard ones and leads to message passing exchanges between subdomains for the distributed memory parallelization paradigm underlying the sample available substructuring. Note that the matrix-vector products at the subdomain level are performed with a linear system solving on each subdomain independently. The local solving of the linear systems of the form,

$$\tilde{M}_{\ell, E} x = b, \quad (15)$$

are performed using the sparsity properties of matrix $\tilde{M}_{\ell, E}$, with the MUMPS library [4].

This strategy is efficient from a computational cost point of view, as indicated in Table 1, and additional optimizations can be performed on the implementation of the parallel conjugate gradient (decentralized communications, preconditioning, etc.)

6 Conclusions

Domain decomposition methods are usually very well suited to implementations on distributed memory architectures, since the data locality is ensured with the geometrical domain substructuring, and is mapped to the local memories of the different processors. Therefore, favorite message passing

Table 1 CPU time; sample of 730 disks with rotating gravity and $\mu = 0.3$

Algorithm	CPU (s)
NSCD	117
NSCDD	63
EA3— $\ell = 0$	46
EA3— $\ell = 10^6$	51

libraries such as MPI are useful for this kind of implementation. The OpenMP paradigm is more suited to shared memory parallelization, with minimal intrusivity in the parallelized code. Nevertheless, an organization with data locality such as domain decomposition usually exhibits better performances on this kind of architecture as well (though the efficient use of parallel architecture lead usually to a smaller number of processors than for the previous approach). Load balancing is an issue for each kind of parallelization strategies, and recent advances in this study are available, see [23] for instance. With the use of coarse space (or augmented algorithms), the parallel part of these algorithms are decreasing, due to the advent of a global coarse problem on the whole physical domain (though it may also be parallelized, as done in this article).

This first attempt to enrich a domain decomposition strategy coupled with the contact dynamics underlines the difficulty to improve the convergence of a nonsmooth solver with an enriched linear predictor. Indeed the convergence rate should be significantly increased for compensating for the cost of the solution of the augmented interface problem. Such a goal cannot be reached with a complete resolution at all the scales and at each time step as proved in Sect. 4.1. With the present approach, the gain is not in the scalability performance that the algorithm enrichment may produce, but on the possibility to add a dedicated computational strategy based on a multiscale sequential strategy (using the coarse problem as a macroscopic scale): the incomplete resolution strategy. This strategy leads to admissible solutions if the contact network is stable enough to limit the interpenetration errors. This topic remains an open question for dynamic flow problems, specially if the granular medium is confined, restricting the contact releases.

The present approach has first to be tested on large-scale 3D examples with several subdomains as presented in [24]. But the main improvement concerns the correction of the interpenetration during the process. The velocity formulation of the unilateral contact law used in the standard NSCD approach [20] leads to local interpenetrations which may be large if we use an averaged criterion and they are not corrected in the following time steps because no elastic restoring force is introduced. Without changing the contact law we propose to investigate the enrichment of the linear numerical step with an elastic contribution. Such an approach joins the conclusions in [1] for a related investigation.

7 Appendix: Parallel conjugate gradient

The domain decomposition distributed database is herein used in a parallel conjugate gradient algorithm. In Algorithm 6, $X_{\ell, E}$ is the interface operator reduced to the contribution of the E th subdomain and $B_{\Gamma \Gamma_E}$ is a boolean mapping from

the local interface Γ_E belonging to the E th subdomain to the global interface Γ .

Algorithm 6 Parallel conjugate gradient on augmented interface problem

Initialization

Right-hand-side: parallel contribution of each subdomain, sequential assembling, back to each subdomain

$$C_{\Gamma_E} \leftarrow A_{\Gamma_E E} \tilde{M}_{\ell, E}^{-1} (R_E + \ell_E v_E + R_E^d)$$

$$C_{\Gamma} \leftarrow \sum_E B_{\Gamma \Gamma_E} C_{\Gamma_E} \quad C'_{\Gamma_E} \leftarrow B_{\Gamma \Gamma_E}^T C_{\Gamma}$$

First iterate, residual and search direction

$$F_{\Gamma_E}^0 \leftarrow 0 \quad R_{\Gamma_E}^0 \leftarrow C_{\Gamma_E} \quad D_{\Gamma_E}^0 \leftarrow C'_{\Gamma_E}$$

Dot product: local contribution per subdomain and assembling

$$\rho^0 \leftarrow \sum_E (R_{\Gamma_E}^0)^T D_{\Gamma_E}^0$$

Matrix-vector product: contribution per subdomain (requires a global solve per subdomain)

$$Q_{\Gamma_E}^0 \leftarrow X_{\ell, E} D_{\Gamma_E}^0$$

First update: contribution per subdomain, assembling the dot product, update per subdomain

$$\alpha^0 \leftarrow \rho^0 / \sum_E (Q_{\Gamma_E}^0)^T D_{\Gamma_E}^0 \quad F_{\Gamma_E}^0 \leftarrow F_{\Gamma_E}^0 + \alpha^0 D_{\Gamma_E}^0$$

Iterations

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

Residual: $R_{\Gamma_E}^k \leftarrow R_{\Gamma_E}^k - \alpha^{k-1} Q_{\Gamma_E}^k \quad R_{\Gamma}^k \leftarrow$

$$\sum_E B_{\Gamma \Gamma_E} R_{\Gamma_E}^k \quad R_{\Gamma_E}^k \leftarrow B_{\Gamma \Gamma_E}^T R_{\Gamma}^k$$

Dot product: $\rho^k \leftarrow \sum_E (R_{\Gamma_E}^k)^T R_{\Gamma_E}^k$

if $(\rho^k / \rho^0 < \epsilon)$ **then**

Terminate conjugate gradient iterations

end if

Conjugate parameter: $\beta^k \leftarrow \rho^k / \rho^{k-1}$

Search direction: $D_{\Gamma_E}^k \leftarrow R_{\Gamma_E}^k + \beta^k D_{\Gamma_E}^{k-1}$

Matrix-vector product: $Q_{\Gamma_E}^k \leftarrow X_{\ell, E} D_{\Gamma_E}^k$

Update: $\alpha^k \leftarrow \rho^k / \sum_E (Q_{\Gamma_E}^k)^T D_{\Gamma_E}^k \quad F_{\Gamma_E}^k \leftarrow F_{\Gamma_E}^{k-1} + \alpha^k D_{\Gamma_E}^k$

end for

References

1. Alart P (2014) How to overcome indetermination and interpenetration in granular systems via nonsmooth contact dynamics: an exploratory investigation. *Comput Methods Appl Mech Eng* 270:37–56
2. Alart P, Dureisseix D (2008) A scalable multiscale LATIN method adapted to nonsmooth discrete media. *Comput Methods Appl Mech Eng* 197(5):319–331
3. Alart P, Iceta D, Dureisseix D (2012) A nonlinear domain decomposition formulation with application to granular dynamics. *Comput Methods Appl Mech Eng* 205–208:59–67
4. Amestoy PR, Duff IS, Koster J, L'Excellent J-Y (2001) A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM J Matrix Anal Appl* 23(1):15–41
5. Avery P, Farhat C (2009) The FETI family of domain decomposition methods for inequality-constrained quadratic programming: application to contact problems with conforming and nonconforming interfaces. *Comput Methods Appl Mech Eng* 198:1673–1683
6. Breitenkopf P, Jean M (1999) Modélisation parallèle des matériaux granulaires. In 4e Colloque National en Calcul des Structures, pp 387–392, Giens. CSMA
7. Dostál Z, Kozubek T, Markopoulos A, Brzobohatý T, Vondrák V, Horyl P (2012) A theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction. *Comput Methods Appl Mech Eng* 205:110–120
8. Dubois F, Jean M, Renouf M, Mozul R, Martin A, Bagneris M (2011) LMGC90. In 10e Colloque National en Calcul des Structures, Giens, CSMA
9. Dureisseix D, Farhat C (2001) A numerically scalable domain decomposition method for the solution of frictionless contact problems. *Int J Numer Method Eng* 50(12):2643–2666
10. Farhat C (1991) A Lagrange multiplier based divide and conquer finite element algorithm. *J Comput Syst Eng* 2:149–156
11. Farhat C, Chen PS, Mandel J (1995) A scalable Lagrange multiplier based domain decomposition method for time-dependent problems. *Int J Numer Method Eng* 38(22):3831–3854
12. Farhat C, Lesoinne M, Pierson K (2000) A scalable dual-primal domain decomposition method. *Numer Linear Algebr Appl* 7:687–714
13. Hoang TMP, Alart P, Dureisseix D, Saussine G (2011) A domain decomposition method for granular dynamics using discrete elements and application to railway ballast. *Ann Solid Struct Mech* 2(2–4):87–98
14. Jean M (1999) The non-smooth contact dynamics method. *Comput Methods Appl Mech Eng* 177:235–257
15. Jourdan F, Alart P, Jean M (1998) A Gauss-Seidel like algorithm to solve frictional contact problems. *Comput Methods Appl Mech Eng* 155(1–2):31–47
16. Koziaziara T, Bićanić N (2011) A distributed memory parallel multibody contact dynamics code. *Int J Numer Methods Eng* 87(1–5):437–456
17. Le Tallec P (1994) Domain-decomposition methods in computational mechanics. *Comput Mech Adv* 1(2):121–220
18. Mandel J (1993) Balancing domain decomposition. *Commun Appl Numer Methods* 9:233–241
19. Mandel J, Tezaur R, Farhat C (1999) A scalable substructuring method by Lagrange multipliers for plate bending problems. *SIAM J Numer Anal* 36(5):1370–1391
20. Moreau JJ (1999) Numerical aspects of sweeping process. *Comput Methods Appl Mech Eng* 177:329–349
21. Nineb S, Alart P, Dureisseix D (2007) Domain decomposition approach for nonsmooth discrete problems, example of a tensegrity structure. *Comput Struct* 85(9):499–511
22. Radjai F, Wolf DE, Jean M, Moreau JJ (1998) Bimodal character of stress transmission in granular packings. *Phys Rev Lett* 80(1):61–64
23. Shojaee Z, Shaebani MR, Brendel L, Török J, Wolf DE (2012) An adaptive hierarchical domain decomposition method for parallel contact dynamics simulations of granular materials. *J Comput Phys* 231(2):612–628
24. Visseq V, Alart P, Dureisseix D (2013) High performance computing of discrete nonsmooth contact dynamics with domain decomposition. *Int J Numer Meth Eng* 96(9):584–598
25. Visseq V, Martin A, Iceta D, Azema E, Dureisseix D, Alart P (2012) Dense granular dynamics analysis by a domain decomposition approach. *Comput Mech* 49:709–723