Distributed Nonsmooth Contact Domain Decomposition (NSCDD): algorithmic structure and scalability

V. Visseq[1], A. Martin[2], D. Dureisseix[3], F. Dubois[1], and P. Alart[1]

1 Introduction

Numerical simulations of the dynamics of discrete structures in presence of numerous impacts with frictional contacts leads to CPU-intensive large time computations. To deal with these problems (e.g. granular materials, masonry structures), numerical tools have been developed, such as the nonsmooth contact domain decomposition (NSCDD), presented Sec 2. We focus herein on a distributed version with parallel detection of fine contacts (Sec. 3) and on two possible communication schemes to solve the interface problem (Sec. 4). Those improvements allow to study scalability and numerical performances of the method for 2D and 3D granular media (Sec. 5).

2 The nonsmooth contact domain decomposition

2.1 Nonsmooth contact dynamics reference problem

In this section we recall briefly the background theory of nonsmooth contact dynamics in the context of time-stepping schemes before an analysis of the main steps of the NSCDD method.

With a time-stepping scheme, no event detection is performed. Once the solution is known at the beginning of a time slab $[t_i, t_{i+1}]$, whose known quantities are denoted with a superscript $(-)$, the quantities at the end of the time slab (without a superscript) have to be determined.

Grain nonsmooth dynamics. Considering a rigid model for the grains, the dynamics of the granular medium is written as the vector equation [4]:

---

[1] LMGC – UMR 5508, Université Montpellier II / CNRS, CC 048 Place Eugène Bataillon, F-34095 Montpellier Cedex 5, France e-mail: `{FirstName}.{Name}@univ-montp2.fr` .[2] LaMSID – UMR 8193 EDF / CNRS / CEA , EDF R&D, 1 avenue du Général de Gaulle, F-92141 Clamart Cedex, France e-mail: `alexandre-externe.martin@edf.fr` .[3] David Dureisseix LaMCoS – Université de Lyon, UMR 5259 INSA Lyon / CNRS, Bâtiment Jean d'Alembert, 18-20 rue des Sciences, F-69621 Villeurbanne Cedex, France e-mail: `David.Dureisseix@insa-lyon.fr`

$$MV - R = R^d, \tag{1}$$

where the prescribed right-hand side is $R^d = R^D + MV^-$. V is the velocity of the grain (it contains the translational degrees of freedom – dof, and the rotational ones); R is the resultant impulse on the grain due to interactions with other grains and R^D are the external prescribed impulses. The matrix M contains both the mass (for the translational dof) and the inertia (for the rotational dof). The assembly of these equations (independent for each grain) is formally written in the same way (1).

Contact interaction. For a unilateral contact Moreau proved via a viability lemma [4], that we can use a velocity-impulse complementary law:

$$\mathscr{R}(v, r) = 0, \tag{2}$$

v is the velocity jump at the contact point between the two interacting grains, r is the impulse at the same contact point. $\mathscr{R}$ is usually a non linear and multivalued relationship between the previous two dual quantities. Both v and r are expressed in the local coordinate basis to the contacts between the interacting grains. Therefore, they are linked to the global kinematic and static quantities with compatibility conditions $v = H^T V$ and $R = Hr$.

Reduced dynamics. Taking the dynamics (1) and the compatibility conditions into account, the reduced dynamics involving material variables can be obtained:

$$Wr - v = -v^d, \tag{3}$$

where W is the Delassus operator: $W = H^T M^{-1} H$, and $v^d = H^T M^{-1} R^d$. To close the problem, one adds the constitutive relation (2), and the reference problem reads:

$$\begin{cases} Wr - v = -v^d \\ \mathscr{R}(v, r) = 0 \end{cases}. \tag{4}$$

The difficulty to solve this problem is at least two-folds: on one hand, the number of unknowns (number of interaction quantities r and v) may be large (for instance, an average of $6.5 \ 10^5$ unknowns for the 3D problem illustrating this paper), and the Delassus operator W is not well conditioned. On the other hand, the constitutive relation is nonsmooth (e.g. it is non linear, and not differentiable). To address the nonsmoothness issue, the NSCD (nonsmooth contact dynamics) method with a non-linear Gauss-Seidel (NLGS) solver [4, 2] is used. To address the large size of the problem, a substructuring approach is proposed.

2.2 Sub-structuring

The proposed sub-structuring may be seen as a FETI-like domain decomposition. Indeed, after the partition of the sample (step detailed in section 3) constraints are

added on the interface grain velocities, with E the index of a subdomain:

$$\sum_{E=1}^{n_s} A_{\Gamma E} V_E = 0,$$ (5)

n_s is the number of subdomains, $A_{\Gamma E}$ is a signed boolean matrix selecting interface grains among subdomains to construct their velocity jumps. This step consists of a perfect gluing procedure, which is quite different from the approach proposed in [3]. The dynamics per subdomain reads:

$$M_E V_E - R_E = R_E^d - A_{\Gamma E}^T F_\Gamma,$$ (6)

where F_Γ are the Lagrange multipliers associated to the previous constraints. One shows that combining equation (5) and (6) the interface problem reads:

$$X F_\Gamma = \sum_{E=1}^{n_s} A_{\Gamma E} M_E^{-1} \left(R_E + R_E^d \right),$$ (7)

with $X = \sum_{E=1}^{n_s} A_{\Gamma E} M_E^{-1} A_{\Gamma E}^T$ the interface operator [5]. The reduced dynamics problem per subdomain has the same structure that the global one provided the addition of Lagrange multipliers as additional external impulses on the given right hand side:

$$\begin{cases} W_E r_E - v_E = -v_E^d + v_E^\Gamma \\ \mathscr{R}(v_E, r_E) = 0, \end{cases},$$ (8)

where $v_E^\Gamma = H_E^T M_E^{-1} A_{\Gamma E}^T F_\Gamma$. To close the problem, the interface behavior (5) or (7) should be added.

2.3 NSCDD algorithmic structure in the LMGC90 platform

The NSCDD method has been implemented into the LMGC90 platform[1] [1] for time-evolution problems (N is the number of time steps). Algorithm 1 describes its structure. A NSCDD iteration is then composed of n_{GS} Gauss Seidel iterations on the reduced dynamics and an update of interface quantities. In practice n_{GS} is chosen equals to 1. In the next two sections we will focus on the underlined stages (with boldface) in the following algorithm 1.

---

[1] www.lmgc.univ-montp2.fr/LMGC90

Algorithm 1 NonSmooth Contact Domain Decomposition (NSCDD)

---

for $i = 1, \ldots, N$ **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**

        In parallel for $E = 1, \ldots, n_s$:

            Compute the velocity $\bar{v}_E^\Gamma$

            Compute $(\bar{r}_E, \bar{v}_E)$ with n_{GS} non-linear Gauss-Seidel iterations on:

$$\begin{cases} W_E \bar{r}_E - \bar{v}_E = -\bar{v}_E^d + \bar{v}_E^\Gamma \\ \mathscr{R}(\bar{v}_E, \bar{r}_E) = 0 \end{cases} \qquad (9)$$

            Update $(r_E, v_E) \leftarrow (\bar{r}_E, \bar{v}_E)$

            Compute $\bar{R}_E$ and correct the velocity on interface grains to get $A_{\Gamma E} \bar{V}_E$

        In sequential, but may be possibly parallelized (DCS version):

            Compute ΔF_Γ **as:** $X \Delta F_\Gamma = \sum_{E=1}^{n_s} A_{\Gamma E} \bar{V}_E$ **and update interface impulses** F_Γ

    end while

    Update grain positions in parallel

end for

---

3 Contact detection

At the beginning of a time step, positions and velocities of grains are known and the contact network between bodies has to be computed. Contact detection is a CPU time consuming task, especially for a large number of bodies —this is directly related to the number and the shape of the elements considered. Usually, an efficient solution is to proceed to a two-level detection, i.e. a rough (and cheap) detection followed by an elimination of loose contact predictions and the computation of contact frame (the fine detection).

3.1 Partitioning based on "rough" contact network

Once a rough detection has been performed, the interaction graph consists in nodes associated to grains and edges associated to interactions. We choose to distribute interactions among subdomains as in [5] (we proceed by distributing the middle points between the centers of mass of interacting grains, according to their coordinates, using an arbitrary regular underlying grid, Figure 1(a)). Indeed, with such a choice, the "boundary" grains are duplicated in the subdomains. If a grain indexed with i is connected with m^i subdomains, m^i is called its multiplicity number. For consistency for the rigid model of the grains, the masses and moments of inertia are distributed among the neighboring subdomains according to their multiplicity number, in a partition of unity manner. We remark that rough detection, and so the domain partitioning, does not have to be done at each time step, but at a user-defined frequency (fixed at 10 time steps for numerical tests of section 5).

Fig. 1 Rough (a) and fine (b) interaction network and their associated domain partitioning. Striped grains represent grains of multiplicity $m^i > 1$; dashed lines represent interactions roughly detected which vanishes in effective contact network.
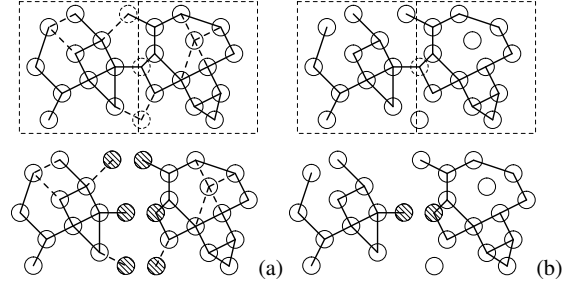
3.2 Parallelized fine detection

Once the domain decomposition has been performed, data can be distributed among the processors and a fine contact detection can be performed in parallel on each substructure local data. Nevertheless contacts roughly detected may disappear at the end and the multiplicity number of the grains may have been incorrectly predicted (Figures 1(a) and 1(b) show cases we may encountered). In particular, predicted boundary grains could turn out not to belong to the minimal interface (computed thanks to the fine contact graph). Their adding to the interface gluing step does not change the problem to solve but increases the size of data to transfer between processors. A future optimization should be to correct interface structures and material parameters to take this phenomenon into account.

4 Communication schemes for solving interface problem

In this section we present two communication schemes associated to centralized and distributed interface problem solving procedure. As one has to solve the interface problem for each NSCDD iterations, to define an appropriate algorithmic formulation, minimizing the data exchanges between processes, is a key issue for the performances of the proposed method.

4.1 Centralized communication scheme (CCS)

At a first glance, the interface gluing step (7) is defined as a global linear equation linking all the subdomains. This is replaced in the iterative algorithm by requiring communications between the subdomains such that one process gathers all the velocity contributions to the vector of velocity jumps. The value of the Lagrange multipliers F_Γ computed sequentially is then distributed such that subdomain E receives its minimal data amount $A_{\Gamma E}^T F_\Gamma$.

4.2 Decentralized communication scheme (DCS)

Due to the structure of the interface operator X, extensively studied in [5], each distributed database (per process related to subdomain E^*) is sufficient to construct the elementary contribution to the interface operator:

$$X_{\Gamma_{E^*}} = \sum_{E=1}^{n_s} A_{\Gamma_{E^*}E} M_E^{-1} A_{\Gamma_{E^*}E}^T, \tag{10}$$

$A_{\Gamma_{E^*}E}$ is a signed boolean matrix, mapping grains of subdomain E to velocity jumps of the elementary interface Γ_{E^*} (restriction of the global interface to the boundary of subdomain E^*). Then, an elementary interface problem can be defined as:

$$X_{\Gamma_{E^*}} \Delta F_{\Gamma_{E^*}} = \sum_{E=1}^{n_s} A_{\Gamma_{E^*}E} V_E. \tag{11}$$

Finally, the data gathering of $\sum_{E=1}^{n_s} A_{\Gamma_{E^*}E} V_E$ on each process corresponds to data exchanges over an unstructured topology. Indeed discrete element methods, such contact dynamics, may deal with large/elongated bodies, possibly related to all subdomains. A common example of such bodies is a wall which support contacts on a large range. With the computation of the assembling of local contributions, it is easy to show that this is the expected iterated vector:

$$\Delta F_\Gamma = \sum_{E=1}^{n_s} B_{\Gamma E} D_E B_{\Gamma_E E}^T \Delta F_{\Gamma_E}, \tag{12}$$

$B_{\Gamma E}$ is a boolean matrix selecting interface grains among subdomains, $B_{\Gamma_E E}$ is a boolean matrix selecting elementary interface grains among subdomains and D_E is a diagonal matrix with value $1/m^i$ for entries corresponding to grain i.

4.3 Performance comparison of the two communication schemes

The influence of the proposed communication schemes is studied regarding the CPU time percentage consumed during MPI exchanges (Table 1) with respect to the whole CPU time of a simulation. The test consists of a sample with 55000 spheres submitted to an isotropic compaction, over 500 time steps (Figure 2).

Results presented in Table 1 show clearly the gain we may obtain considering DCS compared to CCS. Decentralized communication scheme indeed allows to avoid MPI collective communications (especially expensive, in our case, to scatter updating of Lagrange multipliers) and to partially parallelize interface treatment.

Table 1 Comparison of elapsed CPU time percentage consumed during MPI exchanges for centralized (CCS) and decentralized (DCS) communication schemes; isotropic compaction of a 55000 spheres sample.

n_s	Partitioning parameters (x, y, z)	CPU percentage (CCS)	CPU percentage (DCS)
1	1 1 1	0 %	0 %
3	3 1 1	31.3 %	14.0 %
4	2 2 1	35.6 %	9.1 %
8	2 2 2	58.3 %	18.4 %

(a)   (b)
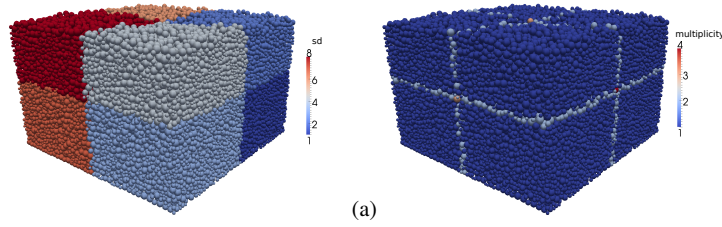
Fig. 2 Sample of 55000 spheres submitted to isotropic compaction. Subdomains indexes (a) and multiplicity number of grains (b).

5 Scalability preliminary results

We propose to study scalability of the NSCDD method on tests consisting in samples of (2D) disks and (3D) spheres submitted to basic loadings. The speedup S_p, function of the number of processes N_p (supposed equals to the number of subdomains), and the number of total iterations, over 100 time steps, are then highlighted. On both tests, friction is considered at contact between particles. Simulations are performed on a 48 cores AMD processor.

(a)   (b)
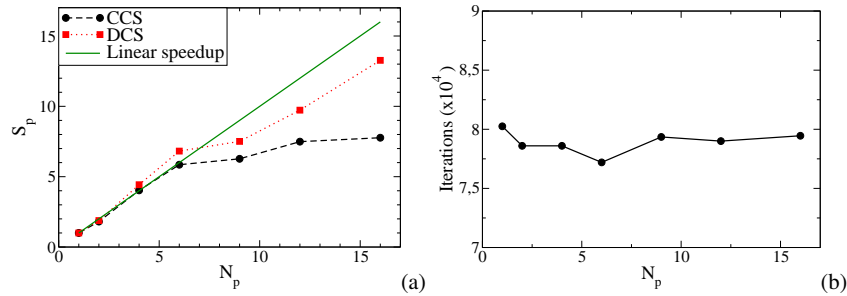
Fig. 3 Speedup (a) and total number of iterations (b); biaxial loading of a 13000 disks sample.

2D – biaxial test. As shown in Figure 3, the speedup does not change drastically depending on the communication scheme for a quite small 2D sample, at least

for a small number of processes. The number of iterations (independent from the communication scheme) is nearly constant for all the tested domain splittings.

3D – triaxial test. For 3D granular samples (Figure 4) the centralized communication scheme has very poor efficiency so it is not reported here. We consider a random closed packing of 64000 spheres subjected to triaxial compaction (downward displacement of the top wall with a constant velocity and confining stress acting on the lateral walls). That is the hardest mechanical configuration one may encountered because of the strong indeterminacy of the problem cumulated to the high number of contacts unknowns (6.5×10^5 in average in our case), but also the most interesting numerical case for the domain decomposition method proposed. We see that the speedup has good quantitative behavior, even if the hardware and MPI library optimization may be improved. Indeed, the use of about a hundred processors (for larger problems than those studied here) implies to mobilize a supercomputing platform to obtain reasonable speedup.
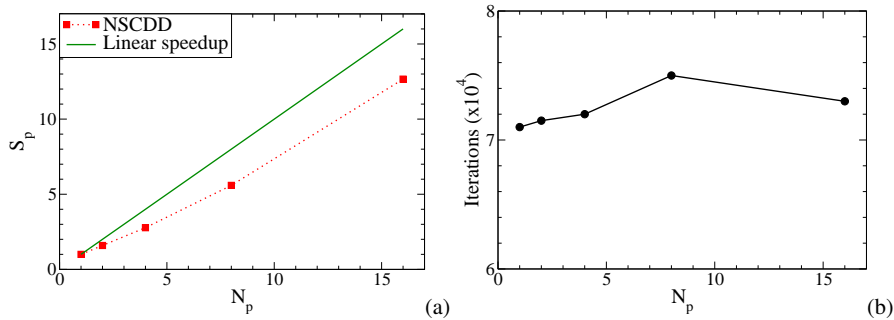
Fig. 4 Speedup (a) and mean number of iterations (b); 64000 spheres sample.

References

1. Dubois, F., Jean, M., Renouf, M., Mozul, R., Martin, A., Bagneris, M.: LMGC90. In: 10e colloque national en calcul des structures. Giens, France (2011)
2. Jean, M.: The non-smooth contact dynamics method. Comput Method Appl M **177**, 235–257 (1999)
3. Koziara, T., Bićanić, N.: A distributed memory parallel multibody contact dynamics code. Int J Numer Meth Eng **87**(1-5), 437–456 (2011)
4. Moreau, J.J.: Numerical aspects of sweeping process. Comput Method Appl M **177**, 329–349 (1999)
5. Visseq, V., Martin, A., Iceta, D., Azema, E., Dureisseix, D., Alart, P.: Dense granular dynamics analysis by a domain decomposition approach. Comput Mech **49**, 709–723 (2012)