A mixed and multiscale domain decomposition approach for transient dynamic analysis

David Odièvre, Pierre-Alain Boucard, Fabrice Gatuingt

A mixed and multiscale domain decomposition approach for transient dynamic analysis

D. Odièvre[1], P.-A. Boucard[1] and F. Gatuingt[1]

[1]LMT-Cachan (ENS Cachan/CNRS/Universit Paris 6/PRES UniverSud Paris)
61 av. du President Wilson,F-94230 Cachan, France

Keywords: multiscale computational method, transient dynamic, domain decomposition, LATIN method, contact, friction, assembly.

Modeling and simulation have an important role in engineering and design departments and raise multiple problems, particularly in dynamics in the case of large assemblies with connections. These connections play a major role in the dimensioning process because they are subject to highly nonlinear local phenomena (contact and friction) which are even more important in fast transient dynamic problems and require very fine meshes in order to be represented correctly [1]. Therefore, the choice of an efficient computational method is of vital importance to simulate such problems within reasonable calculation time.

The aim of the present work is to build a specialized method to answer the problems previously describe due to the fact that the non-linearities are localised in the connections. The applications concern elastic structural assemblies in dynamics with local nonlinearities, such as unilateral contact with friction. Our approach is based on a decomposition of the assembly into substructures and interfaces. An iterative scheme based on the multiscale LArge Time INcrement (LATIN) method [2] is used to solve the sub-structured problem. The multiscale LATIN method is a mixed method which deals with both velocities and forces at the interfaces simultaneously. Within each substructure, the problem is solved using the finite element method. This strategy has already been applied successfully to a variety of static problems. Here, the present work concerns the extension of the multiscale approach to dynamic problems. The multiscale approach consists in solving a homogenized macroscopic problem in order to accelerate the convergence of the iterative scheme.

First, we introduce the multiscale approach in the LATIN strategy for the dynamic case, focusing particularly on the construction of the "macroscopic" problem in space, which has a less conventional meaning in this case than in statics. We present some results about the convergence property of the multiscale LATIN method in dynamics. This iterative computational strategy is also suitable for parallel computing on PCs cluster. It allows to compute problems with high degrees of freedom. We present the property of this parallel algorithm and illustrate its efficiency through 3D examples. Finally we present 3D applications with high numbers of contact interface.

Abstract

The aim of the present work is to develop an efficient strategy for the simulation of dynamic problems with multiple contacts. The approach is based on the multiscale LATIN method with domain decomposition. This iterative computational strategy is suitable for parallel computing on PCs cluster. It allows to compute problems with a large number of degrees of freedom. This strategy has already been applied successfully to a variety of static problems; here, it is extended to dynamics.

Keywords: multiscale computational method, transient dynamics, domain decomposition, LATIN method, contact, friction, assembly.

1 Introduction

Modeling and simulation have an important role in engineering and design departments and raise multiple problems, particularly in dynamics in the case of large assemblies with connections. These connections play a major role in the dimensioning process because they are subject to highly nonlinear local phenomena (contact and friction) which are even more important in fast transient dynamic problems and require very fine meshes in order to be represented correctly [1]. Therefore, the choice of an efficient computational method is of vital importance to simulate such problems within reasonable calculation time.

Among the methods usually used to deal with these problems in dynamics, one can quote the FETI method (often qualified dual Schur method) applied to transient response simulations [2]. The dual substructuring method can be also associated with multispace-multiscale methods: for example, in [3].

The aim of the present work is to build a specialized method to answer the problems

previously describe due to the fact that the non-linearities are localised in the connections. The applications concern elastic structural assemblies in dynamics with local nonlinearities, such as unilateral contact with friction. Our approach is based on a decomposition of the assembly into substructures and interfaces. An iterative scheme based on the multiscale LArge Time INcrement (LATIN) method [4, 5] is used to solve the sub-structured problem. The multiscale LATIN method is a mixed method which deals with both velocities and forces at the interfaces simultaneously. Within each substructure, the problem is solved using the finite element method. This strategy has already been applied successfully to a variety of static problems. Here, the present work concerns the extension of the multiscale approach to dynamic problems. The multiscale approach consists in solving a homogenized macroscopic problem in order to accelerate the convergence of the iterative scheme.

First, we introduce the multiscale approach in the LATIN strategy for the dynamic case, focusing particularly on the construction of the "macroscopic" problem in space, which has a less conventional meaning in this case than in statics. This iterative computational strategy is also suitable for parallel computing on PCs cluster. It allows to compute problems with a large number of degrees of freedom. This strategy was programmed in C++ in the framework of the finite element platform developed at the LMT Cachan. Librairies such as MPI and METIS are used for the parallelization of the strategy. We present the property of this parallel algorithm and illustrate its efficiency through a 3D example of a bolted joint.

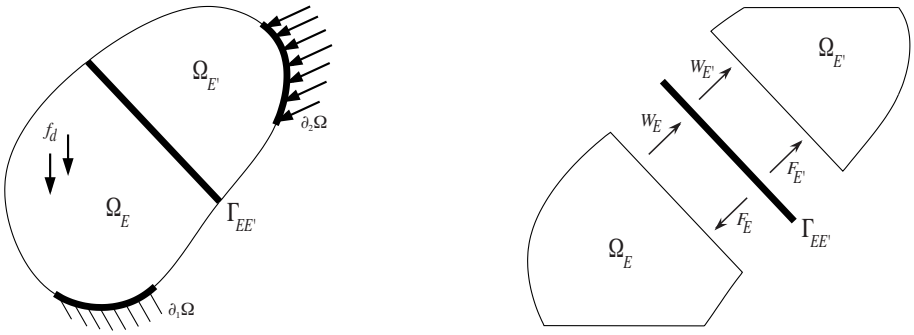
2 The multiscale LATIN method

First we present the multiscale domain decomposition method we propose to carry out transient dynamic problem. This method is based on three ingredients: spatial decomposition of the domain, separation of the scales and a resolution algorithm. The main features of these three ingredients are developed below. The details of the method itself can be found in [4].

2.1 Decomposition into substructures and interfaces

An assembly is a set of substructures which communicate with one another through interfaces, see Figure 1(a). Each interface represents a connection. The substructures and interfaces have their own variables and equations (admissibility, equilibrium and behavior). Two connected substructures are denoted Ω_E and $\Omega_{E'}$ and the associated interface is designated by $\Gamma_{EE'}$.

Each interface is a mechanical entity with its own variables and its specific behavior, which depends on the type of connection. Many different types of connections, e.g. frictional contact, can be modeled with this approach. The interface variables consist of two force fields F_E, $F_{E'}$ and two dual velocity fields W_E, $W_{E'}$, see Figure 1(b). By convention, F_E and $F_{E'}$ represent the action of the interface on the substructures, and



(a) Reference problem

(b) Exchange between interface and substructure

Figure 1: Decomposition of the reference problem into substructures and interfaces

W_E and $W_{E'}$ are the velocities of the substructures viewed from the interface. Thus, the interface concept can be easily extended to the boundary, where the displacements or the velocities or the forces are prescribed.

2.2 Multiscale extension

In order to ensure the theoretical scalability of the method, our approach introduces a spatial description of the unknowns on two scales, called the macroscale and the microscale. In this multiscale strategy, the interfaces play a major role of scale separation: the definitions of the microscopic and macroscopic fields are related to the interface quantities of the substructured problem and are expressed prior to any discretization.

Let us consider an interface $\Gamma_{EE'}$ whose unknowns (W_E, F_E) are divided into two parts: $W_E = W_E^m + W_E^M$ and $F_E = F_E^m + F_E^M$, where W^M and W^m denote respectively the macro parts and the micro complements of the velocity field. The separation of the two scales is obtained by means of the projection operator $\Pi_{\Gamma_{EE'}}$, defined for each interface. Over $\Gamma_{EE'}$, we write W^M and F^M in the form $X^M = \sum (X, e_i^M) e_i^M = \Pi_{\Gamma_{EE'}} X$.

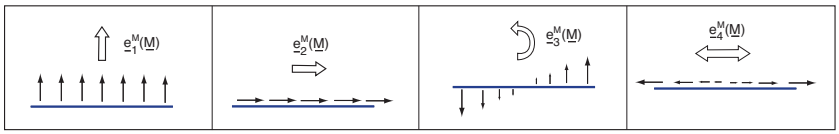


Figure 2: The affine basis functions $\{e_i^M\}$ of an interface $\Gamma_{EE'}$

The choice of the macroscopic projector influences the efficiency of the algorithm.

The selection of the optimum projector was studied in [7]. The basis functions $\{e_i^M\}$ for a 2D problem are represented in Figure 2. The macroscopic kinematics which results from this choice consists of two translations, one rotation and one strain.

2.3 The substructured problem

■ **The problem within a substructure:** We assume a linear behavior of the material under small perturbations. The external loadings are time dependent. The displacement field at any point M of Ω_E and at any time t of $[0,T]$ is $u_E(M,t)$, and the associated space is $\mathcal{U}^{[0,T]}$. ε_E is the strain field and the current state of the structure is characterized by the stress field σ_E, whose associated space is $\mathcal{S}^{[0,T]}$. The mechanical problem to be solved within each substructure Ω_E is:

Find the evolutions of the displacement field $u_E(M,t)$ and stress field $\sigma_E(M,t)$ such that:

- Kinematic admissibility: $\forall t \in [O,T], u_E \in \mathcal{U}^{[0,T]}$
  - Initial condition: $\forall M \in \Omega_E$
$$u_E(t=0) = U_E^0 \qquad \frac{du_E}{dt}(t=0) = V_E^0$$
  - Boundary condition: $\forall t \in [O,T], \forall M \in \Gamma_{EE'}$
$$\frac{du_E}{dt}\Big|_{\Gamma_{EE'}} = W_E \qquad u_E\big|_{\partial\Omega_1} = U_d \qquad F_E\big|_{\partial\Omega_2} = F_d$$
- Equilibrium: $\forall t \in [0,T], \forall \dot{u}^* \in U_0^{[0,T]}, \sigma_E \in \mathcal{S}^{[0,T]}$
$$\int_{\Omega_E} \Big(\rho \frac{d^2 u_E}{dt^2} + f_d\Big)\dot{u}^* d\Omega + \int_{\Omega_E} Tr\left(\sigma_E \varepsilon(\dot{u}^*)\right) d\Omega = \sum_{E'} \int_{\Gamma_{EE'}} F_{EE'} \dot{u}^* d\Gamma$$
- Elastic behavior: $\forall t \in [0,T], \forall M \in \Omega_E$
$$\sigma_E = \mathbf{K}_E \varepsilon(u_E)$$
  where $\mathbf{K}_E$ is the Hooke's operator.

■ **The problem at the interfaces:** The mechanical problem to be solved at each interface $\Gamma_{EE'}$ is:

Find the evolutions of the force fields $F_E(M,t), F_{E'}(M,t)$ and velocity fields $W_E(M,t)$, $W_{E'}(M,t)$ such that:

- Behavior: $\forall t \in [O,T], \forall M \in \Gamma_{EE'}$
$$(F_E, F_{E'}) = A_{\Gamma_{EE'}}(W_E, W_{E'})$$
  where the behavior is expressed as an evolution law $A_{\Gamma_{EE'}}$. This law can be nonlinear, e.g. for frictional contact.

2.4 Resolution strategy: the LATIN method

The LATIN (LArge Time INcrement) method [6] is a general, mechanics-based computational strategy for the resolution of time-dependent nonlinear problems which operates over the entire time-space domain. It has been applied successfully to a variety of problems [7, 8, 9, 10].

In our particular case of linear elastic substructures, the solution $u_E(M,t)$, $\sigma_E(M,t)$ can be calculated from the boundary values $W_E(M,t)$, $F_E(M,t)$. Thus, a solution s is represented only by the force and velocity fields on both sides of an interface. The solution of Problem s_{ref} is expressed as a set of time-dependent fields within each substructure and at the corresponding interfaces:

$$s_{ref} = \sum_E s_E \qquad s_E = \big\{F_E(M,t), W_E(M,t)\big\}$$

■ **Separation of the difficulties:** The LATIN approach is based on the idea of dealing with each difficulty separately in order not to have to solve a global problem and a nonlinear problem at the same time. The equations are divided into global linear equations and local nonlinear equations, so that $s_{ref} = A_d \cap \Gamma$ is the intersection of two subspaces:

- A_d, the space of the solutions of the linear equations associated with the substructures Ω_E: kinematic admissibility, equilibrium, elastic behavior and admissibility of macroquantities;
- Γ, the space of the solutions of the local equations related to the interfaces $\Gamma_{EE'}$ and expressing their behavior.

■ **A two-step iterative strategy:** The LATIN method consists in seeking fields of Γ and A_d alternatively in two search directions E^+ and E^-, as shown in Figure 3. Each iteration involves two stages, called the local stage and the linear stage:

Local stage: given $s_n \in A_d$, find $\widehat{s}$ such that:

$$\begin{array}{rcll} \widehat{s}_{n+1/2} & \in & \Gamma & \text{(interfaces)} \\ \widehat{s}_{n+1/2} - s_n & \in & E^+ & \text{(search directions)} \end{array}$$

Linear stage: given $\widehat{s} \in \Gamma$, find s_{n+1} such that:

$$\begin{array}{rcll} s_{n+1} & \in & A_d & \text{(substructures)} \\ s_{n+1} - \widehat{s}_{n+1/2} & \in & E^- & \text{(search directions)} \end{array}$$

In our particular case of linear elastic substructures, the search directions are defined as follows:

$$\begin{array}{rcl} \widehat{s}_{n+1/2} - s_n \in E^+ & \Longleftrightarrow & \widehat{F}_E - F_E = \mathbf{k}_0(\widehat{W}_E - W_E) \\ s_{n+1} - \widehat{s}_{n+1/2} \in E^- & \Longleftrightarrow & F_E - \widehat{F}_E = -\mathbf{k}_0(W_E - \widehat{W}_E) \end{array} \tag{1}$$

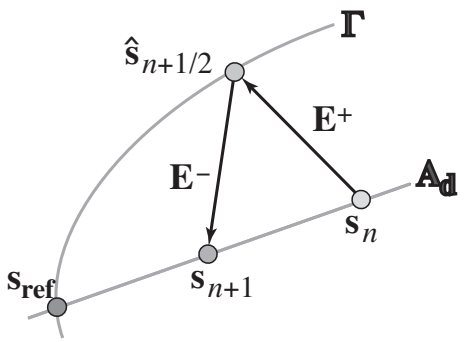


Figure 3: An iteration of the LATIN method

where $\mathbf{k}_0$ is a scalar parameter of the method. As long as $\mathbf{k}_0$ is positive, the solution of the problem does not depend on the value of this parameter, which affects only the convergence rate of the algorithm. In the dynamic cases which we are studying here, the optimum value of $\mathbf{k}_0$ for a 1D problem is given by [10]:

$$\mathbf{k}_0 = \sqrt{\rho E}$$

where E is the Young's modulus and ρ the density. $\mathbf{k}_0$ can be interpreted as a local impedance of the material.

An error indicator η is used to control the convergence of the algorithm. This indicator is a measure of the distance between the two solutions s_{n+1} and $\widehat{s}_{n+1/2}$:

$$\eta = \frac{\sum_E \|s_{n+1} - \widehat{s}_{n+1/2}\|^2}{\sum_E \|s_{n+1}\|^2 + \sum_E \|\widehat{s}_{n+1/2}\|^2}$$

where: $\|s_{n+1}\|_E^2 = \int_0^T \int_{\partial\Omega_E} F_E^T \mathbf{k}_0^{-1} F_E + W_E \mathbf{k}_0 W_E dS dt$

2.5 The local stage: $\widehat{\mathbf{s}}_{\mathbf{n+1/2}}$

This stage consists in building $\widehat{s}_{n+1/2} \in \Gamma$ knowing $s_n \in Ad$. Then, $(\widehat{s}_{n+1/2} - s_n)$ must follow the search direction E^+. Let us consider the case of a perfect interface $\Gamma_{E'E}$. The unknowns are $(\widehat{W}_E, \widehat{W}_{E'}, \widehat{F}_E, \widehat{F}_{E'})$ and must verify the behavior equation:

$$\widehat{F}_E + \widehat{F}_{E'} = 0 \qquad \widehat{W}_E = \widehat{W}_{E'} \tag{2}$$

The solution of Equations 1 and 2 is:

$$\underline{\widehat{W}}_E = \underline{\widehat{W}}_{E'} = \frac{1}{2}(\underline{W}_E + \underline{W}_{E'}) - \frac{1}{2\mathbf{k}_0}(\underline{F}_E + \underline{F}_{E'})$$
$$\underline{\widehat{F}}_E = -\underline{\widehat{F}}_{E'} = \frac{1}{2}(\underline{F}_E - \underline{F}_{E'}) - \frac{\mathbf{k}_0}{2}(\underline{W}_E - \underline{W}_{E'})$$

The local stage consists in solving local problems at this interface. The case of more complex interfaces (contact, friction,...) was developed in [7].

2.6 The linear stage: s_{n+1}

This stage consists in building $s_{n+1} \in Ad$ knowing $\widehat{s}_{n+1/2} \in \Gamma$:

- Macro admissibility: in order to ensure the admissibility conditions of the macro variables, we introduce Lagrange multipliers $\widetilde{W}_E^M$ at the interfaces.
- Search direction: the unknowns (W_E, F_E) must follow the search direction. Equation 1 must be modified by introducing the Lagrange multipliers defined at the interfaces. The new search direction E^- is defined as follows:

$$(F_E - \widehat{F}_E) + \mathbf{k}_0(W_E - \widehat{W}_E - \widetilde{W}_E^M) = 0 \tag{3}$$

- Equations associated with the substructures: the unknowns (W_E, F_E) must verify the dynamic equilibrium and elastic behavior equations.

These equations lead to the resolution of an independent problem, called the "micro" problem, in each substructure:
Find $u_E(M,t), \forall t \in [0,T], \forall \dot{u}^* \in U_0^{[0,T]}, \sigma_E \in \mathcal{S}^{[0,T]}$

$$\int_{\Omega_E} \Big(\rho \frac{d^2 u_E}{dt^2} + f_d\Big)\dot{u}^* d\Omega + \int_{\partial\Omega_E} \mathbf{k}_0 \frac{du_E}{dt}\dot{u}^* dS + \int_{\Omega_E} \mathbf{K}_E \varepsilon(u_E)\varepsilon(\dot{u}^*) d\Omega = \sum_{E'} \int_{\Gamma_{EE'}} \Big(\widehat{F}_E + \mathbf{k}_0 \widehat{W}_E + \mathbf{k}_0 \widetilde{W}_E^M\Big)\dot{u}^* d\Gamma \tag{4}$$

■ **Discretization:** In each substructure, using a classical finite element discretization $u_E(M) = \{N\}^T\{U\}$ and $\varepsilon_E(M) = [B]\{U\}$, Equation 4 leads to the resolution of an evolution problem: Find $U(M,t), \forall t \in [O,T]$ such that:

$$[M_E]\ddot{U}(t) + [c_E]\dot{U}(t) + [K_E]U(t) = \widehat{F} + \mathbf{k}_0(\widehat{W} + \widetilde{W}^M) \tag{5}$$

where $[M_E]$ and $[K_E]$ are the classical finite element mass and stiffness matrices. Matrix $[c_E]$ is less classical and due only to the LATIN method. These matrices are defined by:

$$\begin{array}{rcl} M_E & = & \int_{\Omega_E} \rho \{N\}^T\{N\} d\Omega \\ c_E & = & \int_{\partial\Omega_E} \mathbf{k}_0 \{N\}^T\{N\} dS \\ K_E & = & \int_{\Omega_E} [B]\mathbf{K}_E[B] d\Omega \end{array} \tag{6}$$

In order to solve the evolution problem (Equation 5), the finite element discretization must be associated with a time integration scheme which can be explicit or implicit.

Here we present the method for an explicit time integration scheme, the classical central difference scheme:

$$\begin{aligned} \dot{U}_{t+\Delta t} &= \dot{U}_t + \frac{\Delta t}{2}(\ddot{U}_t + \ddot{U}_{t+\Delta t}) \\ U_{t+\Delta t} &= U_t + \Delta t \dot{U}_t + \frac{\Delta t^2}{2}\ddot{U}_t \end{aligned} \tag{7}$$

Then, the linear system which needs to be solved at each time step has the following form:

$$\begin{aligned} \Big(\frac{2}{\Delta t}[M_E] + [c_E]\Big)\dot{U}_{t+\Delta t} = \widehat{F}_{t+\Delta t} + \mathbf{k}_0(\widehat{W}_{t+\Delta t} + \widetilde{W}^M_{t+\Delta t}) - [K_E]U_t \\ + \Big(\frac{2}{\Delta t}[M_E] - \Delta t[K_E]\Big)\dot{U}_t + \Big([M_E] - \frac{\Delta t^2}{2}[K_E]\Big)\ddot{U}_t \end{aligned} \tag{8}$$

We use a lumped mass matrix $[M_E]$, and $[c_E]$ is also a diagonal matrix. System (8) cannot be solved because there are two unknowns, $\dot{U}_{t+\Delta t}$ and $\widetilde{W}^M_{t+\Delta t}$. Therefore, we divide the field $\dot{U}$ into two fields, $\dot{U}^1$ and $\dot{U}^2$, such that $\dot{U} = \dot{U}^1 + \dot{U}^2$ and fields $\dot{U}^1$ and $\dot{U}^2$ are solutions of two microproblems:

$$\Big(\frac{2}{\Delta t}[M_E] + [c_E]\Big)\dot{U}^1_{t+\Delta t} = \widehat{F}_{t+\Delta t} + \mathbf{k}_0\widehat{W}_{t+\Delta t} + f(\ddot{U}_t, \dot{U}_t, U_t) \tag{9}$$

$$\Big(\frac{2}{\Delta t}[M_E] + [c_E]\Big)\dot{U}^2_{t+\Delta t} = \mathbf{k}_0\widetilde{W}^M_{t+\Delta t} \tag{10}$$

Equation 9 can be easily solved after a local stage. Equation 10, however, cannot be solved without the knowledge of $\widetilde{W}^M$. This linear system (10) can be easily inverted because of the very small number of degrees of freedom of $\widetilde{W}^M$ (four DOFs per interface for a 2D problem). Thus, we can write:

$$W^{2,M} = \mathbf{L}_E^{-1}\widetilde{W}^M \tag{11}$$

where $W^{2,M} = \Pi_{\Gamma_{EE'}}\dot{U}^2\Big|_{\Gamma_{EE'}}$
and $\mathbf{L}_E$ represents a homogeneous behavior operator for Substructure Ω_E. These operators are calculated only once for all the substructures at the beginning of the algorithm. $W^{2,M}$ is the macro part of the restriction of $\dot{U}^2$ to the interfaces.

■ **The macro problem:** The admissibility of the macroquantities at all the interfaces and the homogeneous behavior of all the substructures (Equation 11) lead to the definition of the "macro" problem.
In order to explain the construction of the macro problem, let us consider a perfect interface. The admissibility of the macroquantities at such an interface corresponds to the continuity of the velocities and to the equilibrium of the macroscopic forces. For an interface $\Gamma_{EE'}$, one has:

$$W^M_E = W^M_{E'} \quad \text{and} \quad F^M_E + F^M_{E'} = 0 \tag{12}$$

The decomposition of Field $\dot{U}$ into $\dot{U}^1 + \dot{U}^2$ using the search direction (3) and the projector $\Pi_{\Gamma_{EE'}}$ involves some other relations for $W^{1,M}$ and $W^{2,M}$ (the macro parts of the restrictions of fields $\dot{U}^1$ and $\dot{U}^2$ to the interfaces):

$$\begin{aligned} W^M &= W^{1,M} + W^{2,M} \\ F^{1,M} - \widehat{F}^M + \mathbf{k}_0(W^{1,M} - \widehat{W}^M) &= 0 \\ F^{2,M} + \mathbf{k}_0(W^{2,M} - \widetilde{W}^M) &= 0 \end{aligned} \tag{13}$$

With such admissibility conditions, we need to introduce two Lagrange multipliers for each interface, as shown in Figure 4.

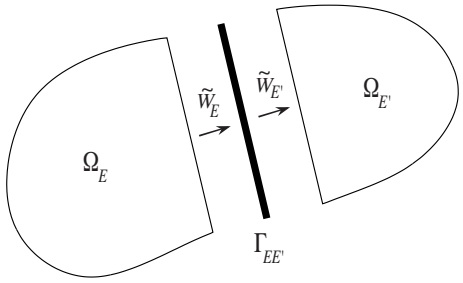


Figure 4: Lagrange multipliers $\widetilde{W}_E^M$ and $\widetilde{W}_{E'}^M$ for a perfect interface

The contribution of the perfect interface $\Gamma_{EE'}$ to the macro problem (Equation 14) is expressed through Equations 11, 12 and 13. The quantity $\underline{\widetilde{W}}_E^M$ is a vector which contains all the Lagrange multipliers of Substructure Ω_E.

$$\begin{bmatrix} \mathbf{L}_E & -\mathbf{L}_{E'} \\ \mathbf{k}_0(1-\mathbf{L}_E) & \mathbf{k}_0(1-\mathbf{L}_{E'}) \end{bmatrix} \begin{bmatrix} \underline{\widetilde{W}}_E^M \\ \underline{\widetilde{W}}_{E'}^M \end{bmatrix} = \begin{bmatrix} -W_E^{1,M} + W_{E'}^{1,M} \\ -F_E^{1,M} - F_{E'}^{1,M} \end{bmatrix} \tag{14}$$

This problem couples all the macro variables of the entire structure and enables us to define the Lagrange multiplier $\widetilde{W}_E^M$ for all the substructures. $W_E^{1,M}$ is the macro part of the solution of the first microproblem (Equation 9); $F_E^{1,M}$ is calculated using the search direction (Equation 13).

3 The algorithm and its parallelization

The LATIN method associated with the mixed domain decomposition method is inherently parallelizable [11]. In our case, this strategy was programmed in C++ in the framework of the finite element platform developed by H. Leclerc [12]. Libraries such as MPI (Message Passing Interface) which carry out transfers of information among machines were used in order to be able to use PCs cluster types of architectures.

■ **Description of the parallel algorithm:** The parallelization of the strategy consists first in distributing the substructures and interfaces among the different processors. The allocation of the substructures to the different processors is carried out through the METIS libraries [13], which enable us to minimize the number of data which must circulate among the processors and not cause an excessive decline in speedup. Next, the different operators specific to the substructures are constructed on each processor. During the iterative resolution phase, there is two kind of problem to solve. First the microproblems which concerns all the substructures are solved on their own processors. As far as the macroproblem is concerned, its parallelization has not been developed yet, and it is solved on a single processor. Finally, the local stage is completely parallelized because the interfaces are distributed among the different processors.

The LATIN method consists in processing linear and local stages alternatively. Algorithm 1 shows the key steps of an iteration of the multiscale method.

Algorithm 1: The micro/macro LATIN method (velocity approach)

- Linear stage

  - Loop over the substructures (on each processor):
    First microproblem: Determination of $(\dot{U}_E^1, W_E^1)$ given $(\widehat{W}_E, \widehat{F}_E)$ (Equation 9). Calculation of $W_E^{1,M} = \Pi_{\gamma_{EE'}} W_E^1$, then of $F_E^{1,M}$, using the search direction and the admissibility conditions of the macroquantities.
  - Macroproblem (on a single processor):
    Determination of $\widetilde{W}_E^M$ given $W_E^{1,M}$ and $F_E^{1,M}$ (Equation 14)
  - Loop over the substructures (on each processor):
    Second microproblem: Determination of $(\dot{U}_E^2, W_E^2)$ given $\widetilde{W}_E^M$ (Equation 10)

  Calculation of $\dot{U}_E = \dot{U}_E^1 + \dot{U}_E^2$

- Local stage

  - Loop over the interfaces (on each processor):
    Determination of $(\widehat{W}_E, \widehat{F}_E)$ given (W_E, F_E) (Equations 3 and 3)

  Iteration until convergence

■ **Property of the parallel algorithm:** In order to reduce the total size of the problem, one can use more substructures than processors. Moreover the management of several substructures per processor leads to a well-balanced load on the different processors. In this case the parallel algorithm has a good speedup because the cost of the communication between the different processor is low. However if the number of substructure is too large, the computation of the macroproblem which is solve on

a single processor became significant in comparison to the microproblem. Then the macroproblem involves a decline of the speedup.

4 Application

4.1 3D linear academic problem - speedup

In order to test the method described previously, let us consider the simple 3D example of the propagation of a compression wave in a parallelepiped (Figure 5). The bar is 1 m long and 0.25 m wide, with Young's modulus 200 GPa, mass density 7,800 kg/m^3 and Poisson's ratio 0.3. The loading consists in a prescribed velocity going from 0 m/s initially to a maximum value of 1 m/s over a period T of 60 μs.

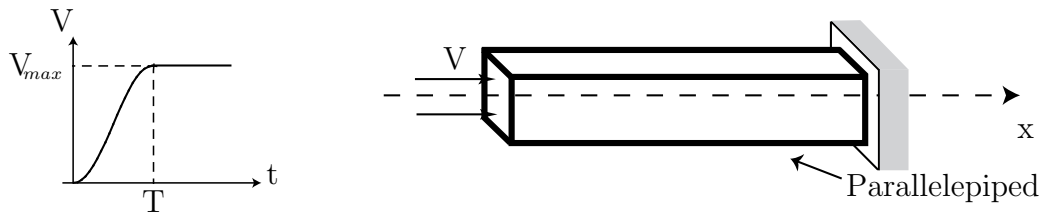


Figure 5: The numerical example

For the purpose of this test, the parallelepiped can be decomposed into several substructures. We used 79 time steps of 5 μs each for a total duration of 395 μs. Figure 6 shows the speedup of the parallel algorithm obtained. For this application, we used a 150,000-DOF mesh decomposed into 144 substructures of 1,000 DOFs each.

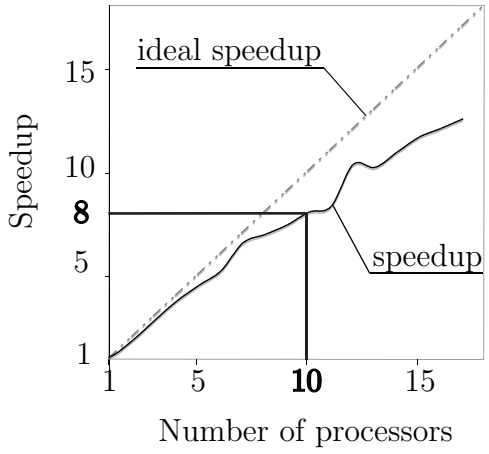


Figure 6: Speedup of the academic example

With a small number of processors, the speedup is very good because the calculation of the macroproblem and the data exchanges among the processors are negligible compared to the calculation of the two microproblems. This is no longer true when the number of processors increases. The decline As the number of processors increases the macroproblem becomes more and more significant compared to the microproblem. Moreover the repartition of the substructures can not be optimum for each number of used processors that's why the behavior of the parallel algorithm can not be fully linear.

4.2 Application to a 3D assembly

This example concerns the propagation of a compression wave in a 3D assembly. This assembly is a joint between two sandwich composites which is made with two metal parts and three bolts (Figure 7). For this example, the core and the composite material of the sandwich composite are assumed to be homogeneous elastic isotropic media. Interface between the sandwich composites and the metal parts are assumed to be perfect, frictional contact is considered for the interfaces between the bolts and the metal parts on the one hand and between the two metal parts on the other hand. The friction coefficient is equal to 0.1. For this example, there is initially a static loading and prestresses in the bolts. Then, a static computation is necessary before the dynamic analysis. This static solution corresponds to the initial conditions of the dynamics analysis. For the dynamic computation, the loading consists of a prescribed velocity as shown in Figure 7 (V_{max} = 1.5 m/s).

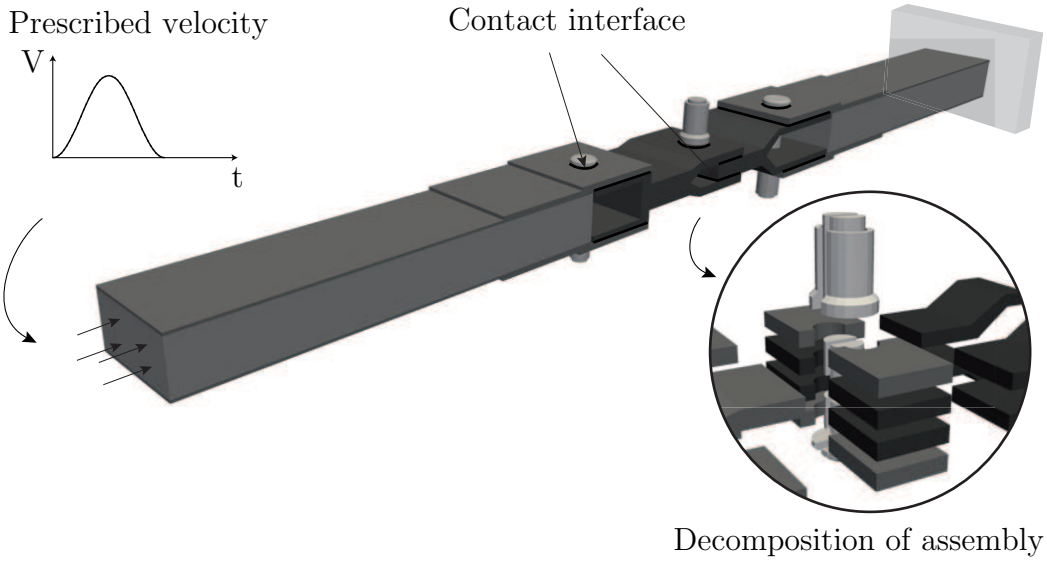


Figure 7: Modeling of the joint

We used a 1,500,000-DOFs mesh for the whole assembly. In order to deal with this problem with our approach, we decomposed the different parts of the assembly into

several substructures. Figure 7 shows a part of this decomposition into substructures. We choose to use substructures with similar number of DOFs, this choice leads to a well-balanced load on the different processors in order to have a good speedup of the parallel algorithm. We used 194 substructures of about 7,500 DOFs each. An implicit scheme is used for the time integration with 159 time steps of 2 μs each for a total duration of 320 μs.

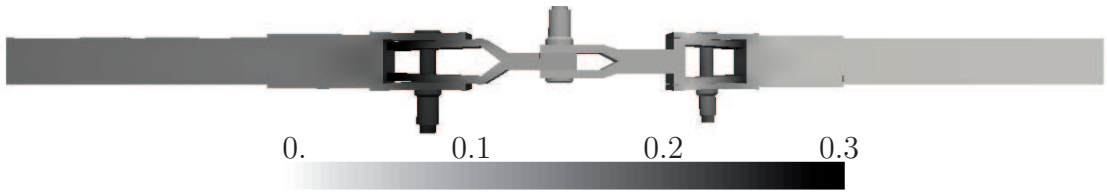


Figure 8: Displacement at time step 40 (mm)

We used a PCs cluster of 35 processors to carry out 100 iterations per time step. These calculations took 7 hours. Figure 8 shows the displacement fields obtained at time step 40. One can observe partial separations between the metal parts and the composite plate near the bolts. Those separations are due to the prestresses in the bolts.

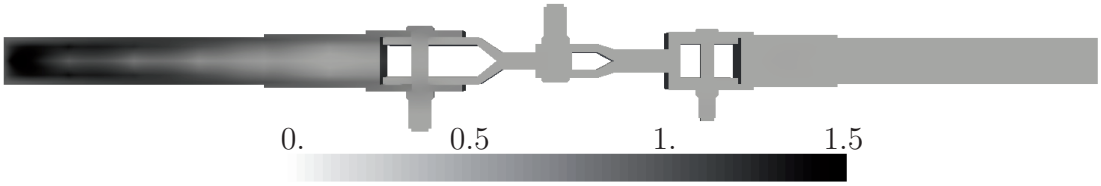


Figure 9: Velocity at time step 20 (m/s)

Figure 9 shows the velocity fields obtained at time step 20, the wave is still in the sandwich composite. Here, one can observe the different wave speed in the composite plate and the core of the sandwich composite. This snapshot shows the importance of using fine mesh to represent correctly the propagation of the wave in particular at the interface between the plate and the core of the sandwich composite.

5 Conclusion

We have presented a "parallel-oriented" algorithm for the resolution of three-dimensional problems in dynamics based on a decomposition of the structure and on an iterative resolution. As was already proven in statics, the multiscale LATIN method which we study in this work is highly parallelizable. On parallel computers, this approach allows an important reduction of the numerical costs compared to the sequential approach.

This strategy allows also a reduction of the size of the problmes compared to a direct approach. The only stage of the strategy developped here which is not parallelized yet is the macroproblem. When the number of substructures is large, the macroproblem causes an excessive decline in speedup of the parallel strategy, moreover this problem may be important in comparison of the microproblem. Some solution have been already developed to solve this problem with a parallel strategy for statics and quasi-statics case [14].

The strategy presented here is also well-adapted to carry out parametric studies in dynamics [15]. However parametric studies on problems with large number of degrees of freedoms was not possible with a sequential strategy. The next step will consist in using this new parallel implementation of the strategy to lead parametric studies for more complex problems like those we present in this work.

References

[1] J.O. Hallquist, G.L. Goudreau and D.J. Benson, *"Sliding interfaces with contact-impact in large-scale lagrangien computations"*. Computer Methods in Applied Mechanics and Engineering, 51, 107-137, 1985.

[2] C. Farhat, P.-S. Chen and J. Mandel, *"A scalable Lagrange multiplier based domain decomposition method for time-dependent problems"*, Int Jal for Numerical Methods in Engineering, 38, 3831-53, 1995.

[3] A. Gravouil and A. Combescure, *"Multi-time-step and two-scale domain decomposition method for non-linear structural dynamics"*, Int Jal for Numerical Methods in Engineering, 58, 1545-69, 2003.

[4] P. Ladevèze, O. Loiseau and D. Dureissex, *"A micro-macro and parallel computational strategy for highly heterogeneous structures"*. Int Jal for Numerical Methods in Engineering, 52, 121-138, 2001.

[5] P. Ladevèze, D. Néron and P. Gosselet, *"On a mixed and multiscale domain decomposition method."*. Computer Methods in Applied Mechanics and Engineering, Vol 196, Num 8, 1526-40, 2007.

[6] P. Ladevèze, *"Nonlinear computational structural mechanics – New approaches and non-incremental methods of calculation"*, Springer-Verlag, 1999.

[7] P. Ladevèze, A. Nouy, O. Loiseau, *"A multiscale computational approach for contact problems"*, Computer Methods in Applied Mechanics and Engineering Sofware, 191, 4869-4891, 2002.

[8] P. Ladevèze, A. Nouy, *"On a multiscale computational strategy with time and space homogenization for structural mechanics*, Computer Methods in Applied Mechanics and Engineering, 192, 3061-3087, 2003.

[9] P.A. Boucard, P. Ladevèze and H. Lemoussu, *"A modular approach to 3D impact computation with frictional contact"*, Computer & Structures, 78, 45-52, 2000.

[10] H. Lemoussu, P.A. Boucard, P. Ladevèze, *"A 3D shock computational strategy for real assembly and shock attenuator"*, Adv. Engrg. Software, 33, 517-526,

2002.

[11] L. Champaney, J.Y. Cognard, D. Dureisseix and P. Ladevèze, *"Large scale applications on parallel computers of a mixed domain decomposition method."*, Computational Mechanics, 19, 253-263, 1997.

[12] H. Leclerc, *"Outil d'assistance au développement d'application d'analyse numérique et de mécanique"*, Séminaire du LMT Cachan, 2005.

[13] G. Karypsis and V. Kumar, *"METIS"*, A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices, Version 4.0, Department of Computer Science, University of Minnesota, 1998.

[14] P. Kerfriden, O. Allix, P. Gosselet, *"A three-scale domain decomposition method for the 3D analysis of debonding in laminates."* Computatinnal Mechanics (to be published).

[15] D. Odièvre, P.A. Boucard, F. Gatuingt, *"A multiscale method for transient dynamic analysis of 3D assemblies with frictional contact."* ECT 2008 - 6th International Conference on Engineering Computational Technology, 2008.