

## Large scale applications on parallel computers of a mixed domain decomposition method

Laurent Champaney, Jean-Yves Cognard, David Dureisseix, Pierre Ladevèze

► **To cite this version:**

Laurent Champaney, Jean-Yves Cognard, David Dureisseix, Pierre Ladevèze. Large scale applications on parallel computers of a mixed domain decomposition method. Computational Mechanics, Springer Verlag, 1997, 19 (4), pp.253-263. 10.1007/s004660050174 . hal-00321318

**HAL Id: hal-00321318**

**<https://hal.archives-ouvertes.fr/hal-00321318>**

Submitted on 9 Dec 2017

**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.

# Large scale applications on parallel computers of a mixed domain decomposition method

This is a post-peer-review, pre-copyedit version of an article published in *Computational Mechanics* 19(4):253-263, 1997. The final authenticated version is available online at:  
<http://dx.doi.org/10.1007/s004660050174>

L. Champaney, J. Y. Cognard, D. Dureisseix, P. Ladevèze

**Abstract** A strategy using a mixed domain decomposition technique is discussed in the case of problems involving a large number of degrees of freedom. The implementation, on parallel computers, of this “mechanical” strategy which splits the structure into substructures and interfaces is presented. Several industrial-type examples, in the case of elastic behaviour and frictional contact, show the interesting features and the capabilities of such an approach. The numerical behaviour on different computers of the iterative procedure and comparisons with the direct resolution method are also presented.

## 1

### Introduction

The analysis of an assembly of three-dimensional structures is not so straightforward when the behaviour is globally nonlinear. Taking into account a complex geometry and trying to obtain an accurate solution lead to increase the size of the problem. Moreover, the treatment of contact and friction nonlinearities needs the use of an iterative process on the whole problem (Simo and Laursen, 1992; Zhong and Mackerle, 1992) which can be costly when the number of contact conditions is high. The use of parallel computers (Noor, 1994) can lead to a reduction in the numerical cost, but their efficiency is improved when adapted algorithms are used (Escaig, Vayssade and Touzot, 1994; Farhat and Roux, 1992, 1994; Heath, Ng and Peyton, 1991; Ladevèze, 1996). For instance, domain decomposition methods have been a great asset for structural analysis over the last few years because they allow parallel-oriented algorithms for use with modern scientific super-computers.

We use a method, based on a decomposition of the structure, that takes into account the mechanical properties of the problem in order to build a “mechanical and parallel”-oriented algorithm. It allows a reduction of both the size of the problem and the numerical cost of its resolution with respect to classical F.E. techniques, even on sequential computers. A general presentation of this approach is given by Ladevèze and Lorong (1992) and some sufficient convergence conditions are given by Ladevèze (1996). In the particular case of perfect liaisons and linear elasticity, it has been formulated with other entirely different approaches. One of them is based on an extension of the Schwarz algorithm to a nonoverlapping subdomain method by Lions (1990) and two others use an augmented Lagrangian formulation (Glowinski and Le Tallec, 1990; Ladevèze, 1985). The extension of this approach to the treatment of unilateral contact nonlinearities is presented in Blanzé et al (1996) and Cognard et al (1996).

A partition of the structure is introduced from two entities: substructures and interfaces. Each substructure is considered as a structure on its own and communicates only with its neighbouring interfaces. The local nonlinearities are treated by a local and mixed manner through a constitutive law associated to an interface. Thus, interfaces are bidimensional entities with their own behaviour and their own associated unknowns. An iterative scheme leads to the resolution of independent problems on each substructure and independent nonlinear mixed problems on the interfaces. It is, therefore, well suited to the use of parallel computers. This domain decomposition method also introduces some modularity when considering local nonlinearities through suited interfaces which easily model the technological reality (prestresses, gaps, unilateral contact, friction, rubber joints, etc.).

This paper focuses on the resolution of problems involving a large number of degrees of freedom. The size of the stiffness matrices is a severe limitation for the finite element treatment of such problems. The reduction of this size, brought by the decomposition technique, and the influence of the number and of the shape of the substructures is shown on a simple problem. The computational cost of the direct resolution method and of the proposed iterative approach are compared on a CRAY C90, for an example of linear elasticity with more than 340 000 d.o.f.. The analysis of an assembly of three-dimensional structures (with prestresses, gaps, unilateral contact and friction) illustrates the flexibility and the possibilities of the method. Moreover, the numerical behaviour of the iterative procedure on a parallel computer, IBM SP2 with 16 processors, is presented.

---

L. Champaney, J. Y. Cognard, D. Dureisseix, P. Ladevèze  
Laboratoire de Mécanique et Technologie  
(E.N.S. de Cachan/C.N.R.S./Université Paris 6)  
61, Avenue du Président Wilson, 94235 Cachan Cedex, France

The study on the structure assembly analysis has been carried out within the scope of a joint research contract with Aérospatiale. We would like to thank in particular J. P. Moreau and P. Blanchard from Aérospatiale-les Mureaux (France) for their collaboration. We also would like to express our gratitude to the CEA (Commissariat à l’Energie Atomique at Saclay – France) for the use of the IBM SP2 and to the IDRIS (Institut du Développement et des Ressources en Informatique Scientifique at Orsay – France) for the use of the CRAY C90.

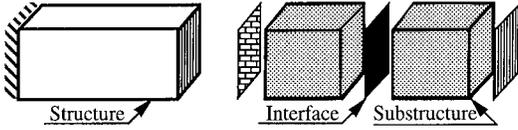


Fig. 1. The two mechanical entities

## 2 Principle of the mechanical strategy

### 2.1

#### Partition of the structure and problem to be solved

A structure  $\Omega$  is decomposed into substructures in order to reduce the globality of the problem. This partition is introduced from two different mechanical entities: substructures denoted by  $\Omega^E$  and interfaces denoted by  $\gamma^{EE'}$ . The considered structure  $\Omega$  is now represented as the assembly of substructures that communicate only with their neighbouring interfaces (Fig. 1).

A displacement field  $W^E$  and a field of surface traction density  $F^E$  ( $F^E$  is the action of the interface onto the substructure) represent the interaction between a substructure and an interface (Fig. 2). These fields are defined on the interfaces and on the boundaries of the substructures. Boundary conditions are also prescribed through interfaces between the substructure and the outside.

We assume a linear behaviour of the material under small perturbations. We suppose that the loading is quasi-static.  $f_d$  denotes the density of body forces.

On each substructure  $\Omega^E$ , the following problem is posed: find  $\{U^E, W^E; \sigma^E, F^E\}$  satisfying:

- Kinematic constrains:

$$U^E|_{\partial\Omega^E} = W^E, U^E \in \mathcal{W}^E \quad (1)$$

$$\text{with } \mathcal{W}^E = \{U^E, \text{ regular in } \Omega^E\}$$

- Equilibrium equation:  $\forall U^* \in \mathcal{W}^E$

$$\int_{\Omega^E} \text{Tr}[\sigma^E \varepsilon(U^*)] d\Omega = \int_{\Omega^E} f_d \cdot U^* d\Omega + \int_{\partial\Omega^E} F^E \cdot U^* dS \quad (2)$$

- Constitutive law:

$$\sigma^E = \mathbf{K} \varepsilon(U^E) \quad (\mathbf{K} \text{ denotes Hooke's elasticity tensor}) \quad (3)$$

$U^E$  and  $\sigma^E$  denote the displacement field and the stress tensor in  $\Omega^E$ , respectively, and  $\varepsilon(U^*)$  denotes the strain tensor produced by a displacement field  $U^*$ .

For each interface  $\gamma^{EE'}$ , the equations depend on the behaviour that has to be modelled. They are local in space variables and can be written as a constitutive law:

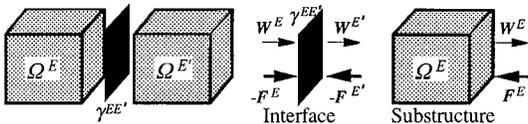


Fig. 2. Interactions between substructures and interfaces

$$\mathcal{R}(W^E, F^E, W^{E'}, F^{E'}) = 0 \quad (4)$$

which can be a nonlinear relation (unilateral contact and friction). For instance, the equations which describe the behaviour of a perfect liaison between  $\Omega^E$  et  $\Omega^{E'}$  are:

$$W^E = W^{E'} \quad \text{continuity of displacement} \quad (5)$$

$$F^E + F^{E'} = 0 \quad \text{equilibrium of forces} \quad (6)$$

### 2.2

#### Description of the algorithm

This approach is based on the LARge Time INcrement method (LATIN method) proposed by Ladevèze (1996, 1992). We are, in this case, in a degenerate situation where we only consider the final configuration.

The LATIN method separates the difficulties of the problem: it avoids the simultaneity of the global aspect and the nonlinear aspect. Thus, it takes into account the mechanical properties of the equations to separate them into two groups: local in space variables and possibly nonlinear equations, on the one hand, and linear and possibly global in space variable equations, on the other hand. They define two subspaces of elements  $s$  which denote the set of unknowns of the complete problem:

$$s = (\varepsilon, W; \sigma, F) = \cup_E s^E = (\varepsilon^E, W^E; \sigma^E, F^E);$$

$$s^E \text{ defined on } \Omega^E \}$$

$S$  denotes the corresponding functional space of elements  $s$ . Since the only nonlinearities studied are defined on the interfaces and in order to have independent global linear problems on each substructure, the two subspaces are defined as follows:

$$A_d = \{s \in S \text{ satisfying } \forall \Omega^E :$$

- the kinematic constrains (Eq. (1))
- the equilibrium equation (Eq. (2))
- the constitutive law on  $\Omega^E$  (Eq. (3))

$$\Gamma = \{s \in S \text{ satisfying } \forall M \in \gamma^{EE'} :$$

- $\mathcal{R}(W^E, F^E; W^{E'}, F^{E'}) = 0$  (Eq. (4))

The problem is to find an element  $s_{ex}$  satisfying both the behaviour of the substructures ( $s \in A_d$ ) and the behaviour of the interfaces ( $s \in \Gamma$ ), i.e. which is located at the intersection of  $A_d$  and  $\Gamma$  (Fig. 3) ( $s_{ex} = \Gamma \cap A_d$ ). The LARge Time INcrement method starts with  $s_0$ , an element of  $A_d$  (for instance  $s_0 = 0$ ), and builds elements  $s$ , which belong to  $\Gamma$  and  $A_d$  successively, up to the solution  $s_{ex}$ . Each

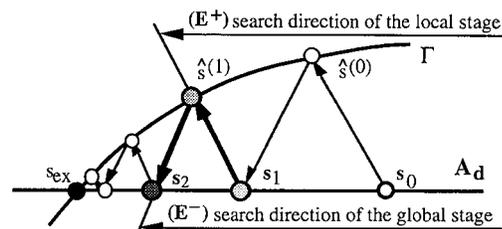


Fig. 3. Presentation of the iterative procedure

iteration, i.e. the building of a new element  $\mathbf{s}_{n+1}$  of  $\mathbf{A}_d$  from a given one  $\mathbf{s}_n$ , requires two stages: the local stage and the global stage. The problem associated with the equations that define  $\mathbf{A}_d$  is ill-posed, and the problem associated with the equations that define  $\Gamma$  has more than one solution. Thus, other equations have to be introduced; they are called “search directions”.

#### local stage:

At the local stage, independent computations are carried out on each interface (on each point of the interfaces). The selected search direction  $\mathbf{E}^+$  is associated with a constant operator  $\mathbf{k}_0$ . The problem to be solved, at iteration “n”, is:  $\mathbf{s}_n \in \mathbf{A}_d$  being known, find  $\hat{\mathbf{s}}$  satisfying:

$$\hat{\mathbf{s}} \in \Gamma \Leftrightarrow \mathcal{R}(\hat{\mathbf{W}}^E, \hat{\mathbf{F}}^E, \hat{\mathbf{W}}^{E'}, \hat{\mathbf{F}}^{E'}) = 0 \quad \forall \mathbf{M} \in \gamma^{EE'} \quad (7)$$

$$(\hat{\mathbf{s}} - \mathbf{s}_n) \in \mathbf{E}^+$$

$$\Leftrightarrow \begin{cases} \hat{\mathbf{F}}^E - \mathbf{F}_n^E = \mathbf{k}_0(\hat{\mathbf{W}}^E - \mathbf{W}_n^E) & \forall \mathbf{M} \in \gamma^{EE'} \\ \hat{\mathbf{F}}^{E'} - \mathbf{F}_n^{E'} = \mathbf{k}_0(\hat{\mathbf{W}}^{E'} - \mathbf{W}_n^{E'}) & \forall \mathbf{M} \in \gamma^{EE'} \end{cases} \quad (8)$$

$$\hat{\mathbf{F}}^{E'} - \mathbf{F}_n^{E'} = \mathbf{k}_0(\hat{\mathbf{W}}^{E'} - \mathbf{W}_n^{E'}) \quad \forall \mathbf{M} \in \gamma^{EE'} \quad (9)$$

In the case of perfect liaisons, an explicit solution is obtained from Eqs. (7), (8) and (9):

$$\begin{aligned} \hat{\mathbf{W}}^E &= \hat{\mathbf{W}}^{E'} \\ &= \frac{1}{2} \left\{ (\mathbf{W}_n^E + \mathbf{W}_n^{E'}) - \mathbf{k}_0^{-1} (\mathbf{F}_n^E + \mathbf{F}_n^{E'}) \right\} \quad \forall \mathbf{M} \in \gamma^{EE'} \end{aligned} \quad (10)$$

$$\hat{\mathbf{F}}^E = -\hat{\mathbf{F}}^{E'} = \frac{1}{2} \left\{ (\mathbf{F}_n^E - \mathbf{F}_n^{E'}) - \mathbf{k}_0 (\mathbf{W}_n^{E'} - \mathbf{W}_n^E) \right\} \quad \forall \mathbf{M} \in \gamma^{EE'} \quad (11)$$

The treatment of unilateral contact with friction is presented in a following section.

For a given interface  $\gamma^{EE'}$ , the computation of the local stage only requires information on this interface; therefore at the local stage, only information between neighbouring substructures is exchanged. Moreover, using the equations that define the search direction, one may only need the displacement quantities  $\mathbf{W}_n^E$  and  $\mathbf{W}_n^{E'}$  on  $\gamma^{EE'}$ .

#### global stage:

The search direction  $\mathbf{E}^-$  is also associated with the  $\mathbf{k}_0$  operator on the interface:

$\hat{\mathbf{s}} \in \Gamma$  being known, find  $\mathbf{s}_{n+1}$  satisfying:  
 $\mathbf{s}_{n+1} \in \mathbf{A}_d$

$$(\mathbf{s}_{n+1} - \hat{\mathbf{s}}) \in \mathbf{E}^- \quad (12)$$

$$\Leftrightarrow \mathbf{F}_{n+1}^E - \hat{\mathbf{F}}^E = -\mathbf{k}_0(\mathbf{W}_{n+1}^E - \hat{\mathbf{W}}^E) \quad \forall \mathbf{M} \in \gamma^{EE'}$$

Using Eq. (12) to express  $\mathbf{F}_{n+1}^E$  as a function of  $\mathbf{W}_{n+1}^E$  and of quantities generated by the solution at the local stage, the problem to be solved is:

Find  $\mathbf{U}_{n+1}^E \in \mathcal{U}^E$  such that  $\forall \mathbf{U}^* \in \mathcal{U}^E$ ,

$$\begin{aligned} \int_{\Omega^E} \text{Tr}[\mathbf{K}\varepsilon(\mathbf{U}_{n+1}^E)\varepsilon(\mathbf{U}^*)]d\Omega + \int_{\partial\Omega^E} \mathbf{k}_0 \mathbf{U}_{n+1}^E \cdot \mathbf{U}^* dS \\ = \int_{\Omega^E} \mathbf{f}_d \cdot \mathbf{U}^* d\Omega + \int_{\partial\Omega^E} \tilde{\mathbf{F}} \cdot \mathbf{U}^* dS \end{aligned} \quad (13)$$

$$\text{where } \tilde{\mathbf{F}} = \hat{\mathbf{F}}^E + \mathbf{k}_0 \hat{\mathbf{W}}^E \quad (\text{boundary density of surface traction on } \partial\Omega^E) \quad (14)$$

then:

$$\forall \Omega^E \quad \sigma_{n+1}^E = \mathbf{K}\varepsilon_{n+1}^E = \mathbf{K}\varepsilon(\mathbf{U}_{n+1}^E) \quad (15)$$

$$\mathbf{W}_{n+1}^E = \mathbf{U}_{n+1}^E|_{\partial\Omega^E} \quad (16)$$

$$\mathbf{F}_{n+1}^E = \tilde{\mathbf{F}} - \mathbf{k}_0 \mathbf{W}_{n+1}^E \quad (17)$$

This is a classical linear problem of structural analysis with a density of body forces and a density of surface traction on  $\partial\Omega^E$ . A finite element discretisation of Eq. (13) leads to the global linear problem on each substructure:

$$(\mathbf{K}_e^E + \mathbf{k}^E) \mathbf{q}^E = \mathbf{F}^E \quad (18)$$

Where  $\mathbf{K}_e^E$ ,  $\mathbf{q}^E$  and  $\mathbf{F}^E$  denote the classical elastic stiffness of the substructure, the vector of nodal displacement and the vector of nodal forces, respectively.  $\mathbf{k}^E$  denotes the interface stiffness.

Therefore, the global stage leads to independent computations on each substructure. It is important to notice that, in Eq. (18)  $(\mathbf{K}_e^E + \mathbf{k}^E)$  is a constant matrix. Thus, the factorisation is performed only once.

For the numerical examples, we choose:

$$\mathbf{k}_0 = k \mathbf{I}_d \quad \text{with} \quad k = E/L_0 \quad (19)$$

where  $\mathbf{I}_d$  is the identity operator,  $E$  the Young’s modulus and  $L_0$  a characteristic length of the structure.  $L_0$  is a parameter of the method and, in most cases, its optimal value is near the maximum length of the structure (Cognard et al, 1996).

An error estimator can be based on the non-satisfaction of the behaviour of the interfaces for an approximate solution  $\mathbf{s}_n$  element of  $\mathbf{A}_d$  (it satisfies all of the other relations). To stop the algorithm, we use an energy error indicator, which is computed after each local stage:

$$\eta^2 = \frac{\sum_E \|\mathbf{s}_n^E - \hat{\mathbf{s}}^E\|^2}{\sum_E \{\|\mathbf{s}_n^E\|^2 + \|\hat{\mathbf{s}}^E\|^2\}} \quad (20)$$

$$\begin{aligned} \text{with} \quad \|\mathbf{s}^E\|^2 &= \int_{\partial\Omega^E} \mathbf{F}^E \cdot \mathbf{k}_0^{-1} \mathbf{F}^E dS \\ &+ \int_{\partial\Omega^E} \mathbf{W}^E \cdot \mathbf{k}_0 \mathbf{W}^E dS \quad \forall \Omega^E \end{aligned} \quad (21)$$

## 2.3

### Simple example: a beam in traction

The beam which is split into 32 substructures is subjected to a prescribed traction  $F_d = 100$  MPa (Figs. 4, 5); the bar length is  $L = 40$  mm; and the bar width is  $l = 10$  mm. The characteristics of the material are Young’s modulus  $E = 200$  GPa and Poisson ratio  $\nu = 0.3$ . To show the influence of the parameter  $k$ , three computations have been carried out. We have chosen: (a)  $k' = k$ , (b)  $k' = 0.5 k$  and (c)  $k' = 1.5 k$ , where  $k$  is the “optimal” value of the search direction parameter ( $k = E/L_0$  with  $L_0 = 2L$ ). Figure 6 shows the evolution of the error indicator with respect to the iteration number, and Figure 7 presents, with respect to the

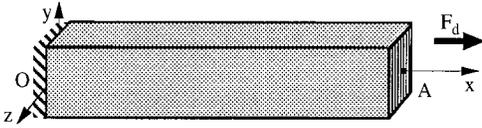


Fig. 4. Studied problem

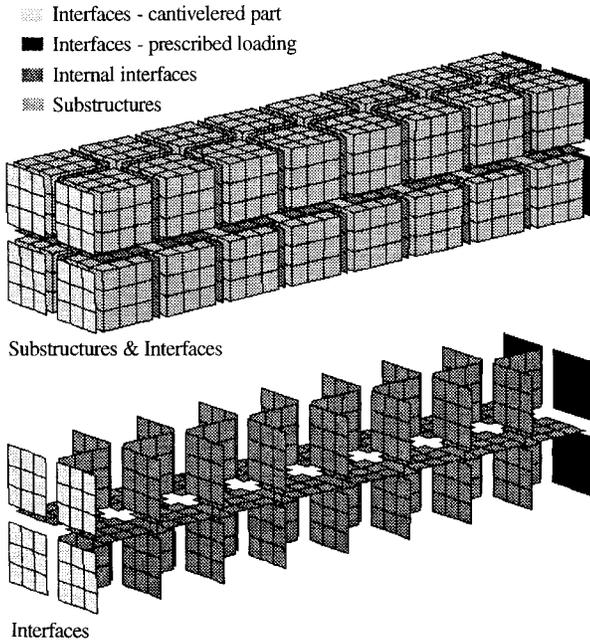


Fig. 5. Splitting of the beam

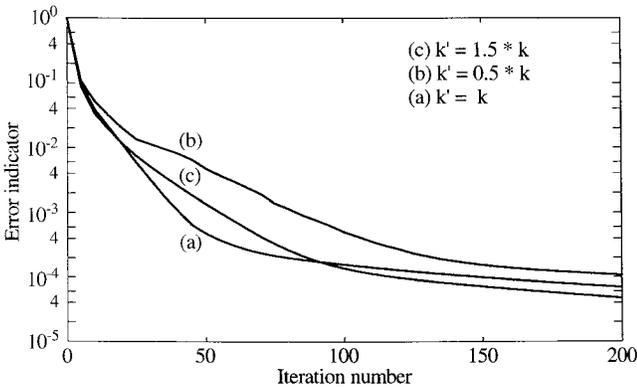


Fig. 6. Evolution of the error indicator with respect to the iteration number

iteration number, the evolution of the stress at the origin point O (near the cantilevered part) and the displacement at the point A (Fig. 4). One can note that a small modification, near the “optimal” value of the parameter  $k$ , has only very little influence on the convergence speed; moreover, the number of elements of the substructures has no influence on the convergence speed (Cognard et al, 1996).

In the case (a), a good level of accuracy is obtained in 60 iterations (Fig. 6); for this value of  $k$ , Figure 8 shows the space evolution of the stress  $\sigma_{xx}$  and of the displacement  $U_x$  with a magnitude of 100.

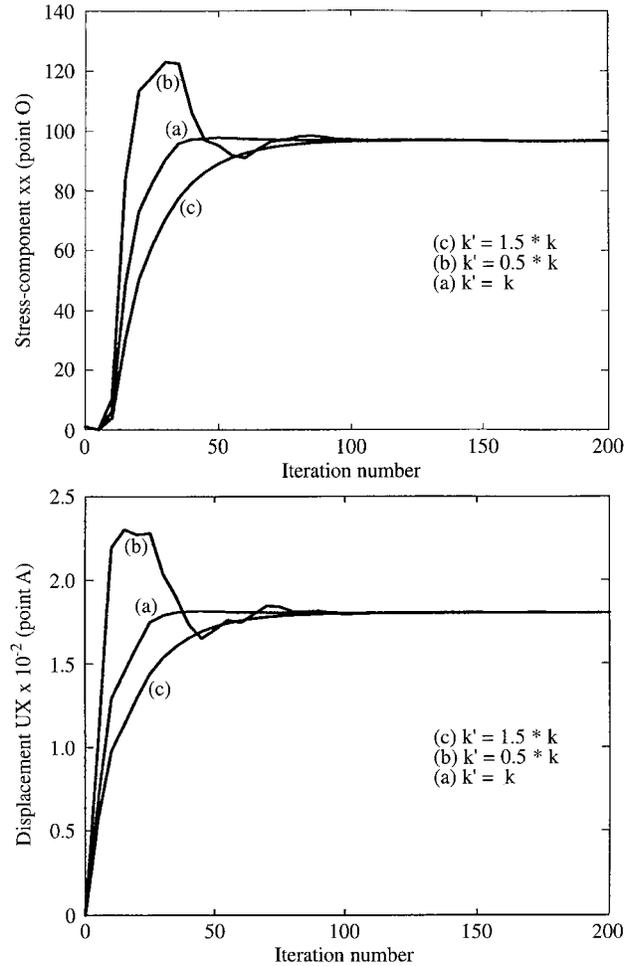


Fig. 7. Evolution of the stress  $\sigma_{xx}$  (MPa) and evolution of the displacement  $U_x$  (mm) with respect to the iteration number

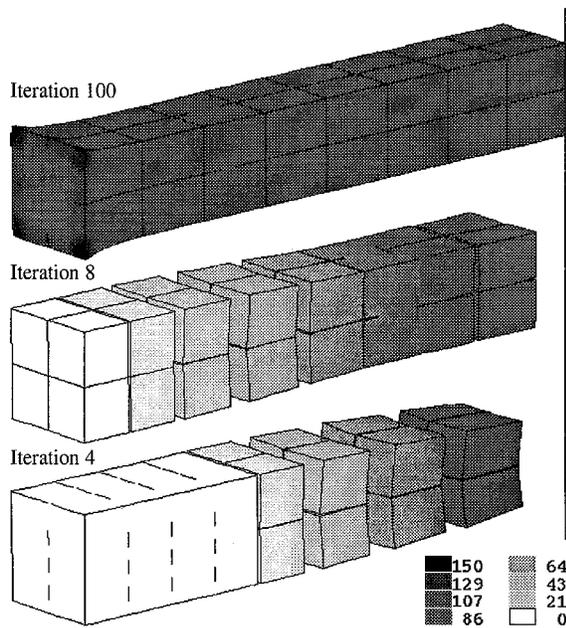


Fig. 8. Space evolution of the stress  $\sigma_{xx}$  (MPa) and of the displacement  $U_x$  (mm) (magnitude 100)

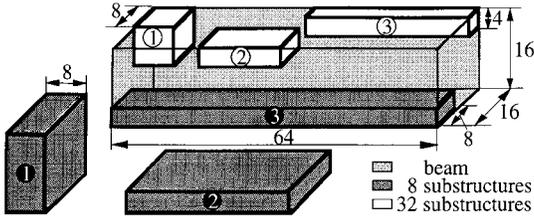


Fig. 9. Splitting of the beam into substructures

Table 1. Size of the problem for different splittings of the beam

number of sub	shape of a sub	nb d.o.f. of a sub	size for a sub (Mb)	total size (Mb)	size sub/size dir
1	64-16-16	217 923	6 168	6 168	1
8 (●)	8-16-16	29 427	440	3 514	0.57
8 (●)	32-4-16	31 227	258	2 062	0.33
8 (●)	64-4-8	32 235	149	1 193	0.19
32 (①)	8-8-8	8 019	66	2 121	0.34
32 (②)	16-4-8	8 331	39	1 238	0.20
32 (③)	32-4-4	8 835	25	808	0.13

### 3 Large-scale linear problems

#### 3.1 Influence of the number of substructures

Very often, for large-scale problems, the size of the stiffness matrix is a limitation. This section presents the influence of the number of substructures and the influence of their shape on the size of the problem. A mesh of a beam with  $64 \times 16 \times 16$  elements (20 node bricks) is taken as an example (Fig. 9). For different configurations of substructures, Table 1 gives the total size required for the proposed substructuring technique resolution (sub) along with a comparison with the required size for a direct resolution (dir). For this example, the size of the problem for a direct resolution is divided by more than 5 when 32 substructures are used.

Long-shaped substructures increase the number of duplicated nodes on the interfaces, but they allow a reduction of the size of the problem. This kind of splitting can be difficult to implement in the case of complex structures; therefore, it is better to increase the number of substructures in order to limit the size of the problem. Since a substructure only communicates with its neighbours, the convergence speed decreases with the number of substructures in a given direction (classical phenomenon for domain decomposition methods). In such cases, convergence acceleration methods (Farhat, Mandel and Roux, 1994; Mandel, 1993) can be used in order to quickly broadcast information between the substructures; these strategies are more powerful for long-shaped structures than for massive three-dimensional structures.

#### 3.2 Implementation on parallel computers with distributed memory

A global stage can be computed concurrently on the substructures  $\Omega^E$ ; for the local stages, the computations can be performed concurrently on the interfaces  $\gamma^{EE}$ . To

SEQUENTIAL	PARALLEL
1 processor ↔ All the substructure	1 processor ↔ 1 substructure
Loop on each substructure ( $\Omega^E$ )	<b>For the substructure <math>\Omega^E</math></b>
Factorisation of $(K_c^E + k^E)$	Factorisation of $(K_c^E + k^E)$
End of loop	Loop on the iterations
Loop on the iterations	Loop on the interfaces ( $\gamma^{EE}$ )
Loop on each interface ( $\gamma^{EE}$ )	Inner interface ← message
Local stage (error estimator)	Local stage (error estimator)
End of loop	End of loop
Convergence test	Convergence test ↔ messages
Loop on each substructure ( $\Omega^E$ )	(contribution of each substructure)
Global stage $(K_c^E + k^E) q^E = F^E$	Global stage $(K_c^E + k^E) q^E = F^E$
End of loop	Loop on the interfaces
	Inner interface → message
	End of loop
End of loop	End of loop

Fig. 10. Sequential and parallel computations

limit the amount of communication between processors, we choose to assign one substructure and its neighbouring interfaces to one processor (Ladevèze and Lorong, 1992). This parallelisation technique leads to the duplication of several local computations on the interfaces, but this is not penalising since the amount of duplicated computation is small. On distributed memory computers, the information exchange between the neighbouring substructures (displacement  $W^E$ ) is performed by message passing. The computation of the accuracy of the approximate solution  $s_n$  needs the contribution of all the substructures, but the information to be exchanged is then small; moreover, this computation does not need to be carried out at each iteration.

The sequential and the parallel algorithms have been implemented with the industrial F.E. code CASTEM 2000 (Verpeaux, Charras and Millard, 1988). The difference between the two versions comes from the fact that, in the first one, a sole program manages all the substructures, whereas in the second one, a program only manages one substructure and only knows the number of its neighbouring substructures. Since the organisation of the computations is different, each version has a specific operating manager system (Fig. 10).

In order to reduce the total size of the problem, one can use more substructures than processors. Therefore, a node-program (on a processor) can manage more than one substructure. This leads to more complicated algorithms, but this technique reduces the amount of communication and is useful in balancing the load on the different processors.

Several tests have been carried out on an IBM SP2. This computer has 16 processors with 128 Mb RAM each. The PVM system (Parallel Virtual Machine system) (Geist et al, 1994; Sunderam et al, 1994) is used to support the communication between different processors.

#### 3.3 Example involving a large number of d.o.f.

These tests were carried out on a biaxial tensile specimen (Fig. 11) (Batisse et al, 1994), whose dimensions are  $230 \times 230 \times 8$  mm (the thickness of the central part is 3 mm). The symmetries allow us to study only one-sixteenth of the specimen. Displacements ( $U_d = 1$  mm) are prescribed on one part of the bores that are drilled into the

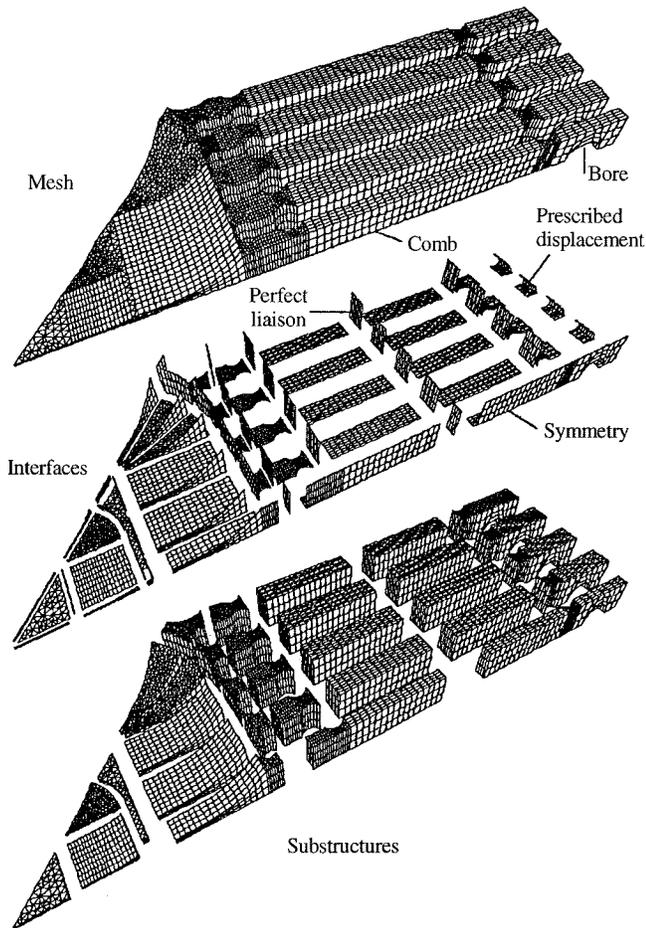


Fig. 11. Biaxial tensile specimen (mesh 1 with 175 740 d.o.f.)

combs (linking system to the testing machine). A decomposition into 31 roughly load-balanced substructures is used to reduce the size of the stiffness matrices. The characteristics of the material are: Young's modulus  $E = 200$  GPa and Poisson ratio  $\nu = 0.3$ , and the search direction parameter is such that  $k = E/L_0$  with  $L_0 = 230$  mm. Table 2 gives the characteristics of the 3 meshes that were used. The size of the problem is given for a skyline storage of the stiffness matrices; a morse storage of the matrices can reduce the size of these problems.

For this example, an accurate solution is obtained after 100 iterations (Fig. 12, 13). Table 3 gives the numerical costs of the 3 tests; "total/proc" is the maximum CPU time per processor, and "total" is the cumulative time. On this computer, a direct computation is feasible only for the first mesh. For a given decomposition into substructures, the numerical cost of the resolution increases slowly with the number of d.o.f.

Table 2. Characteristics of the different meshes

	Direct computation			Substructuring		
	nb d.o.f.	elements	size	nb d.o.f. (sub)	elements (sub)	total size
mesh 1	175 740	11 961	2 160 Mb	6 573	364	750 Mb
mesh 2	259 077	18 471	4 000 Mb	10 011	600	1 350 Mb
mesh 3	340 734	24 764	6 400 Mb	12 687	800	2 130 Mb

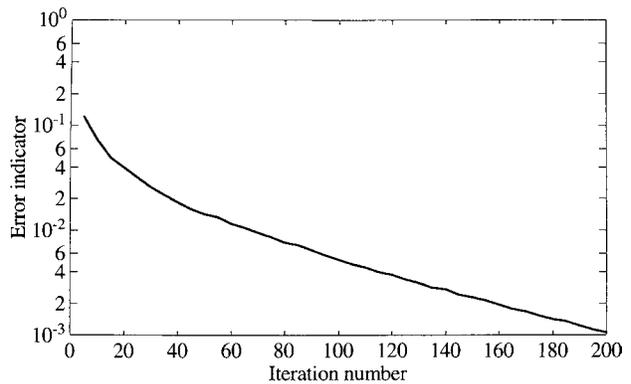


Fig. 12. Evolution of the error indicator with respect to the iteration number

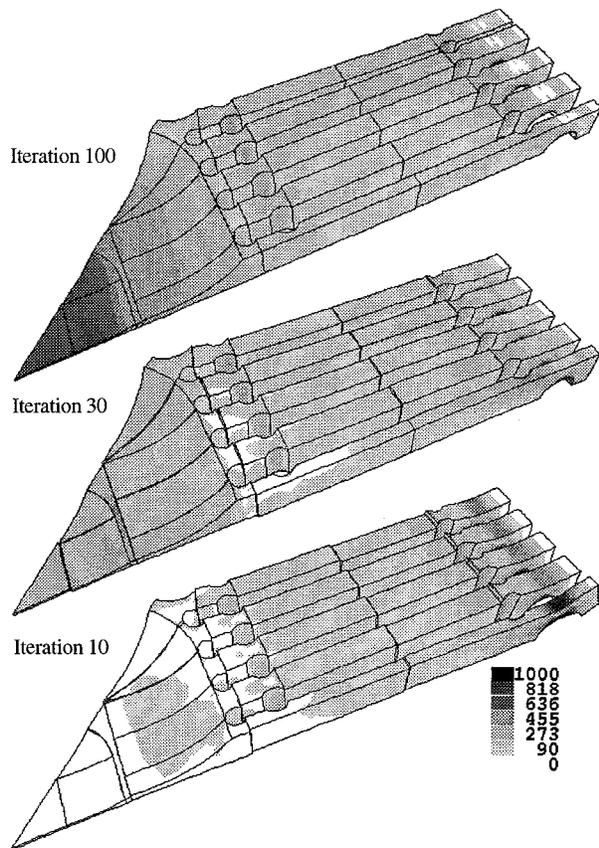


Fig. 13. Mises equivalent stress (in MPa)

Table 3. Cost of the different tests on an IBM SP2 (16 processors) (CPU time in s)

	Direct	Substructuring			
	total	init (sub)	iteration (sub)	total/proc (100 it)	total (100 it)
mesh 1	4 300	37	0.6	194	2 910
mesh 2	-	70	1.0	340	5 100
mesh 3	-	130	1.6	580	8 700

For this example, with a large number of d.o.f., the computational costs of the direct resolution method and of the proposed iterative approach have been compared on a CRAY C90 by using the sequential version of the algorithm

on one processor. Figure 14 displays the results, but they are only indicative; as a matter of fact, this algorithm has not been optimised, and the efficiency of the CRAY C90 is, to a great extent, due to the vectorisation. However, it is important to notice that as the size of the problem increases, the size of the stiffness matrices and the numerical cost for the substructure decomposition method increase more slowly than for the direct resolution method.

#### 4 Structure assembly analysis

Interfaces are also used to model liaisons between structures (prestresses, gaps, unilateral contact, friction, rubber joints, etc.). These interfaces do not increase the computational cost with respect to perfect liaisons because the possibly nonlinear relationships are solved locally in space variables. The following section presents the resolution of the local stage for an interface which models the unilateral contact with friction (Blanzé et al, 1996).

##### 4.1 Unilateral contact with friction

On the supposedly-known part of the boundary susceptible to contact,  $N$  denotes the exterior unit vector to  $\Omega^E$ . The displacement field  $W^E$  and the surface traction density  $F^E$  can be written:

$$W^E = (N \cdot W^E)N + \Pi W^E \text{ and } F^E = (N \cdot F^E)N + \Pi F^E \quad (22)$$

Where  $\Pi$  denotes the tangential projection operator.  $\mu$  denotes the Coulomb friction coefficient and is assumed constant and positive. For static or monotonous quasi-static cases, a displacement formulation is used (Duvaut and Lions, 1976). The different possible cases (separation, contact with sticking or sliding) are easily found by computing, from the known element  $s_n$ , local in space variable quantities at each point of the interface.

The operator which defines the search direction is supposed to be such that  $k_0 = kI_d$  (where  $I_d$  is the identity

**Table 4.** Calculation of the normal components

Separation: $C_n > 0$	Contact: $C_n \leq 0$
$(\hat{W}^{E'} - \hat{W}^E) \cdot N > 0$	$(\hat{W}^{E'} - \hat{W}^E) \cdot N = 0$
$N \cdot \hat{F}^E = N \cdot \hat{F}^{E'} = 0$	$N \cdot \hat{F}^E = -N \cdot \hat{F}^{E'} \leq 0$
$\hat{F}^E = \hat{F}^{E'} = 0$	$\hat{F}^E N = -\hat{F}^{E'} N = kC_n$
$\hat{W}^E = W_n^E - \frac{1}{k} F_n^E$	$\hat{W}^E N = \frac{1}{2} \left\{ \left( W_n^E + W_n^{E'} \right) - \frac{1}{k} \left( F_n^E + F_n^{E'} \right) \right\} \cdot N$
$\hat{W}^{E'} = W_n^{E'} - \frac{1}{k} F_n^{E'}$	$\hat{W}^{E'} \cdot N = \hat{W}^E \cdot N$

**Table 5.** Calculation of the tangential components

Sliding: $\ G_n\  \geq g$	Sticking: $\ G_n\  < g$
$\Pi(\hat{W}^{E'} - \hat{W}^E) = -\lambda \Pi \hat{F}^E$	$\Pi(\hat{W}^{E'} - \hat{W}^E) = 0$
$\ \Pi \hat{F}^E\  = g$	$\ \Pi \hat{F}^E\  = \ \Pi \hat{F}^{E'}\  \leq g$
$\Pi \hat{F}^E = -\Pi \hat{F}^{E'} = g \frac{G_n}{\ G_n\ }$	$\Pi \hat{F}^E = -\Pi \hat{F}^{E'} = G_n$
$\Pi \hat{W}^E = \Pi W_n^E + \frac{1}{k} \Pi(\hat{F}^E - F_n^E)$	$\Pi \hat{W}^E = \Pi W_n^E + \frac{1}{k} \Pi(\hat{F}^E - F_n^E)$
$\Pi \hat{W}^{E'} = \Pi W_n^{E'} + \frac{1}{k} \Pi(\hat{F}^{E'} - F_n^{E'})$	$\Pi \hat{W}^{E'} = \Pi W_n^{E'}$

operator and  $k$  a scalar). A contact scalar indicator  $C_n$  gives the solution on the normal components (Table 4):

$$2C_n = \left( \hat{W}^{E'} - \hat{W}^E \right) \cdot N - \frac{1}{k} \left( \hat{F}^{E'} - \hat{F}^E \right) \cdot N \\ = \left( W_n^{E'} - W_n^E \right) \cdot N - \frac{1}{k} \left( F_n^{E'} - F_n^E \right) \cdot N \quad (23)$$

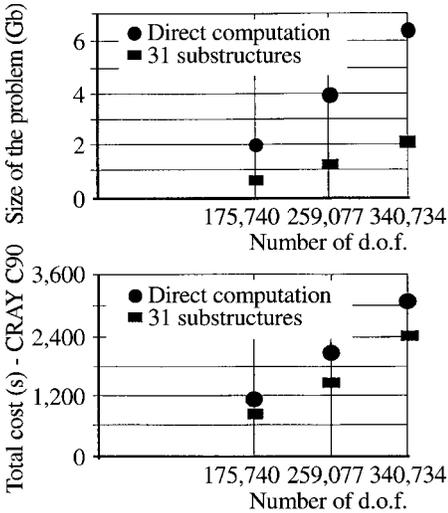
In the contact case, a sliding vectorial indicator  $G_n$  (computed in the same fashion as  $C_n$ ) gives the solution on the tangential components (Table 5):

$$2G_n = k \Pi \left( \hat{W}^{E'} - \hat{W}^E \right) - \Pi \left( \hat{F}^{E'} - \hat{F}^E \right) \\ = k \Pi \left( W_n^{E'} - W_n^E \right) - \Pi \left( F_n^{E'} - F_n^E \right) \\ (g = \mu |\hat{F}^E \cdot N|) \quad (24)$$

##### 4.2 Bolted joint

A comparison with the industrial F.E. code ABAQUS (Pawtucket, 1994) is shown on this example of a bolted joint between three plates. One fourth of the joint is studied (Fig. 15) and the prescribed displacements (zero displacement at the end of the middle plate and  $2.10^{-2}$  mm at the end of the other ones) are such that the forces are transmitted by friction between the plates. Frictional contact is modelled between the plates and between the plates and the bolts (with gap) which are prestressed. The characteristics of the material are: Young's modulus  $E = 200$  GPa and Poisson ratio  $\nu = 0.3$ . The approximate length of the studied part of the joint is 200 mm.

The frictional contact conditions are prescribed in ABAQUS using node-to-node "GAP" elements. A  $10^{-5}$  mm elastic slip is to be allowed on the friction conditions to obtain the convergence of ABAQUS algorithm. The same mesh is used by ABAQUS and by the sequential



**Fig. 14.** Cost of the different computations (sequential algorithm)

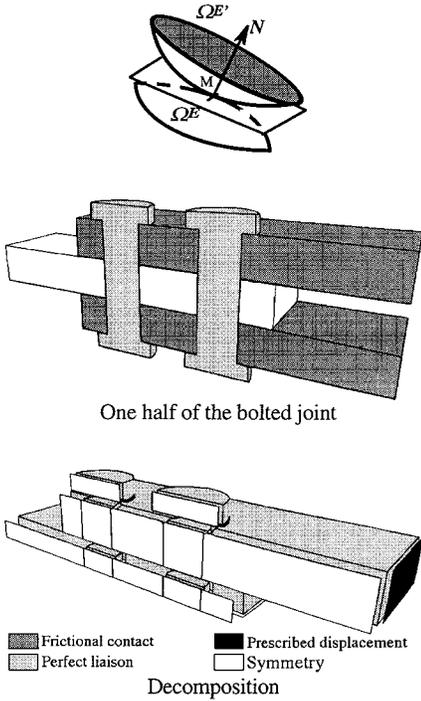


Fig. 15. Bolted joint and the decomposition that is used

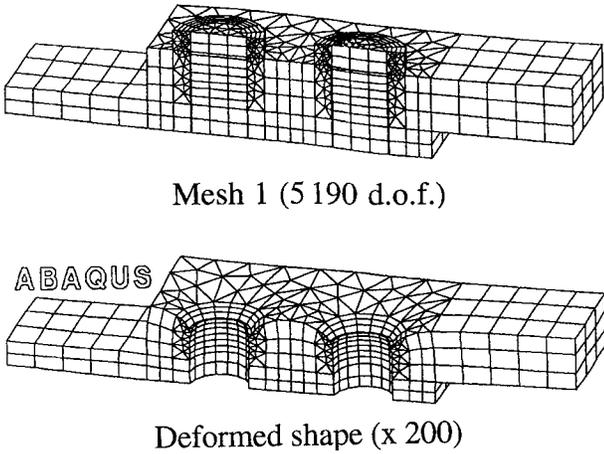


Fig. 16. First mesh and deformed shape

version of the mixed domain decomposition method. Two different meshes (5 190 and 27 078 d.o.f.) were tried.

Table 6 presents the results of the numerical comparisons that were carried out on a HP700 workstation. The number of unknowns is larger in ABAQUS resolution because of the great number of Lagrange multipliers that are used on the prescribed displacement conditions. About twenty load increments are to be carried out in this re-

Table 6. Comparison with ABAQUS (CPU times on HP700)

	ABAQUS				Mixed approach			
	nb d.o.f.	size (Mb)	nb. incr.	CPU time (s)	nb d.o.f.	size (Mb)	nb iter.	CPU time (s)
mesh 1	6 140	15	23	673	5 190	1	200	55
mesh 2	31 137	260	26	35 000	27 078	50	200	720

solution because the amount of elastic slip allowed is small and thus the first increment has to be very small. In fact, after a few load increments, no changing appears on the contact status. A great reduction of the size of the problem and of the computation time is brought by the domain decomposition approach.

It is important to notice that this study only concerns some particular simple cases of frictional conditions where the time evolution has no influence. Some further developments will consider this evolution in order to model complex time dependant loadings. Further developments will also concern the optimisation of the algorithm in order to reduce the numerical costs.

### 4.3 Car differential

This example deals with a car differential (Champany et al, 1995) for which we only model the box, the ring gear, the cap and the screws (Fig. 17). We consider that the sun wheel axis is cantilevered, and we model the action of the sun wheels and of the planet wheels on the differential box. We consider an action on one tooth of the ring gear; the amplitude and the direction of the forces are parametered by the applied torque and the geometry of the gears (Fig. 18). The action of the ball bearings on the box is modelled with some unilateral contact interfaces. The centrifugal force is also modelled. The characteristics of the material are: Young's modulus  $E = 200$  GPa, Poisson ratio  $\nu = 0.3$  and  $\rho = 7 800$  kg/m<sup>3</sup>. The parameter is such that  $k = E/L_0$  with  $L_0 = 800$  mm.

Interfaces are used to model the perfect liaisons between the different substructures of the box, the ring wheel and the cap; they also model the unilateral contact with friction between the different parts. The prestresses in the screws are prescribed with suitable interfaces between the screws and their heads. Three meshes (Fig. 20) have been used and Table 7 presents the size of the problem for two different splittings (each of the 18 screws is always split into 2 substructures). For the first decomposition (39 substructures), the box, the ring gear and the cap are each considered as one substructure; for the other decomposition (84 substructures and 223 interfaces, Fig. 18), the ring gear and the cap are each split into 8 substructures, and the box is split into 24 substructures. For the second decomposition, the symmetries of the structures and an automatic mesh decomposer, like in (Farhat, 1988), have been used. One can note that in this example, the increase of the number of substructures leads to a large reduction of the problem's size.

Convergence is reached after 200 iterations (Fig. 21). Figure 22 shows, at iteration 200, the stress  $\sigma_{xx}$  drawn on the ring (normal traction between the box and the ring gear) and the screws, and the Mises equivalent stress drawn on the box and the cap, for a 2 000 Nm applied torque and a 2 000 rpm rotation speed.

Figure 23 shows how the substructures are distributed on the different processors of a IBM SP2 (with 16 processors). The management of several substructures per processor, by the same node-program, leads to a well-balanced load on the different processors and reduces the amount of the communication. Moreover, the cost of the

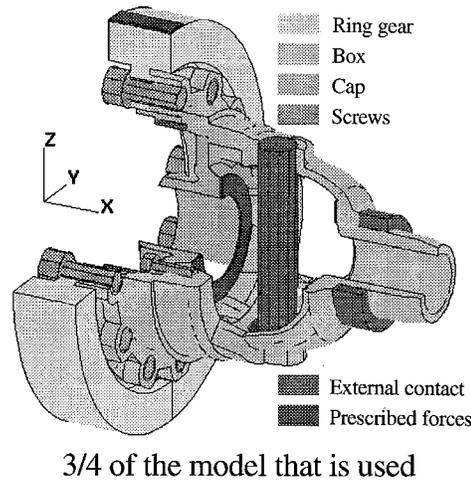
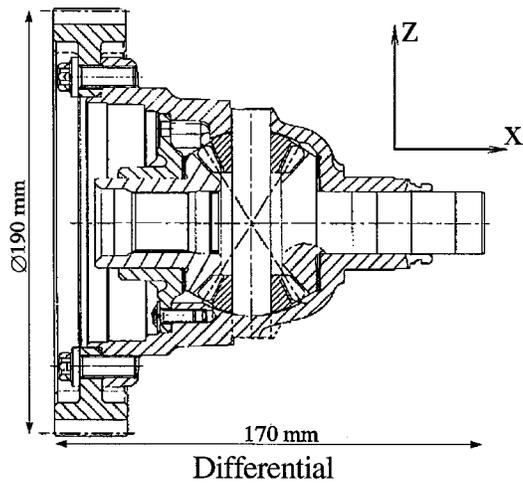


Fig. 17. Differential and the model that is used

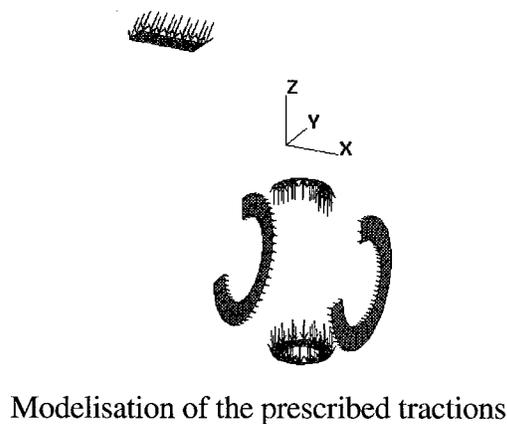
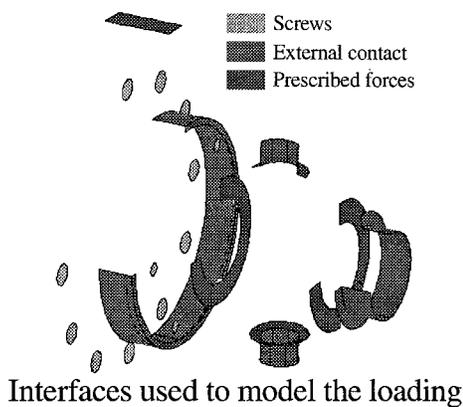


Fig. 18. Prescribed tractions (3/4 of the model)

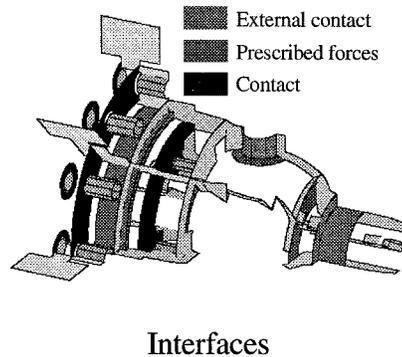
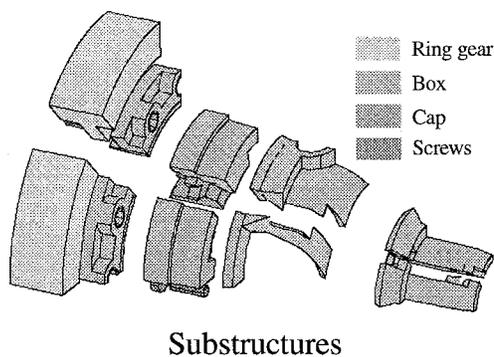
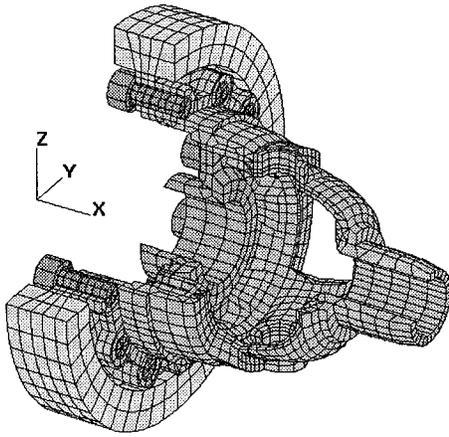


Fig. 19. Model partitioned into 84 substructures (1/4 of the model)

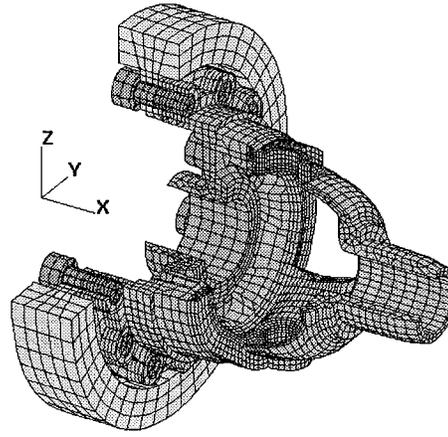
Table 7. Characteristics of different problems

	nb d.o.f.	size for 39 sub	size for 84 sub
mesh 1	35 136	128 Mb	31 Mb
mesh 2	129 924	1 391 Mb	310 Mb
mesh 3	236 976	4 574 Mb	790 Mb

communication between the different processors is quite low. The cumulative CPU time increases slowly with respect to the number of processors (Fig. 24); for one processor, there is no communication. For this example, with a large number of contact zones, the numerical cost increases slowly with respect to the number of d.o.f. (Fig. 24); in fact, the cost for a perfect interface is nearly the same as the cost for an interface which models unilateral contact with friction.



Mesh 1: 35 136 d.o.f (8 node bricks)  
 Mesh 2: 129 924 d.o.f (20 node bricks)



Mesh 3: 236 976 d.o.f (20 node bricks)

Fig. 20. Different meshes used

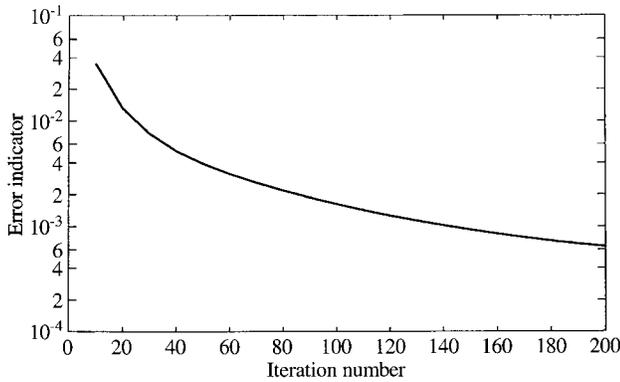


Fig. 21. Evolution of the error indicator with respect to the iteration number

## 5 Conclusions

We have presented a “parallel-oriented” algorithm for the resolution of three-dimensional problems based on a decomposition of the structure and on an iterative resolution

scheme. This algorithm allows a reduction of both the size of the problems and the numerical costs, even on sequential computers. Moreover, the proposed strategy introduces some modularity when considering local nonlinearities through suited interfaces; it is an efficient tool to easily model the technological reality of an assembly of three-dimensional structures. The use of such interfaces leads to solve only local in space variable relations; therefore, these interfaces do not increase the numerical cost with respect to perfect liaisons. The first numerical results in the case of elastic problems involving a large number of d.o.f. are encouraging. On parallel computers, the management of several substructures per processor leads to a well-balanced load on the different processors; moreover, the cost of the communications between the different processors is low. An optimisation of the algorithm and a better management of the information should allow a reduction of the numerical costs on every kind of computer. Another solution would be to use some convergence acceleration methods (Farhat, Mandel and Roux, 1994) in order to quickly broadcast the information between the substructures. Some hierarchical strategies are

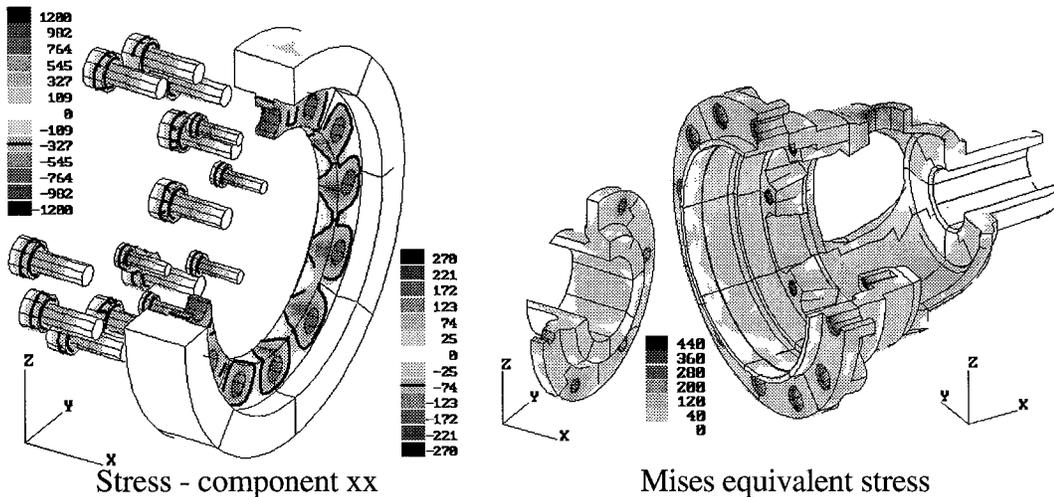


Fig. 22. Numerical results (stress in MPa)

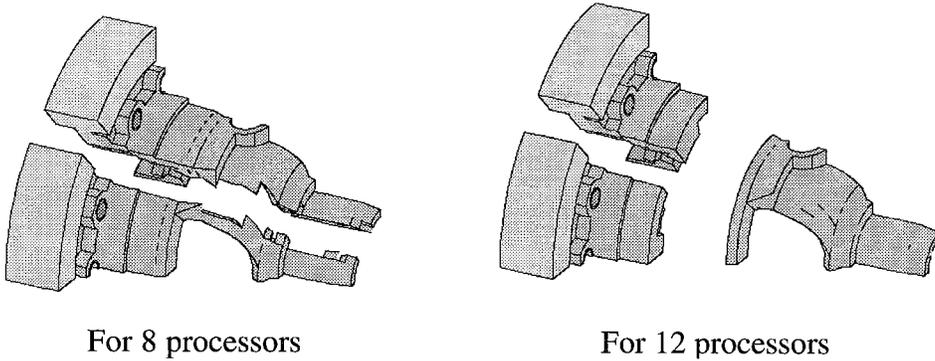


Fig. 23. Groups of substructures managed by the different processors (1/4 of the model)

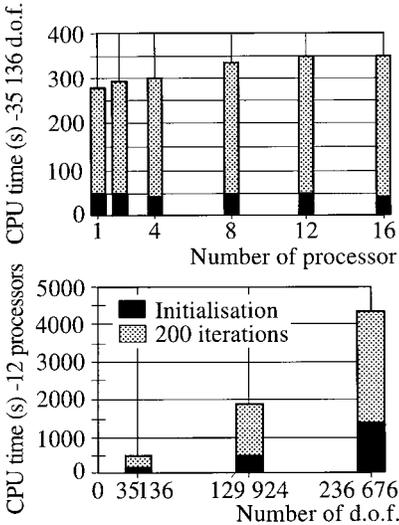


Fig. 24. Cumulative CPU time (s) on IBM SP2 (16 processors)

under way to increase the efficiency of the proposed method when the number of substructures is large.

### References

Batisse, R.; Difant-Jaekels, H.; Curie, F.; Virely, J. M. (1994): Biaxial high-cycle fatigue test on a gas transmission pipeline steel. In: Proceedings of Fourth International Conference on Biaxial/Multiaxial Fatigue. 129–142. St Germain en Laye

Blanzé, C.; Champany, L.; Cognard, J. Y.; Ladevèze, P. (1996): A modular approach to structure assembly computations. Application to contact problems. *Engng Comput.* 13, 15–32

Champany, L.; Cognard, J. Y.; Dureisseix D.; Ladevèze, P. (1995): 3D structure assembly analysis with a modular approach suited to parallel computations. In: Proceedings of 28<sup>th</sup> International Symposium on Automotive Technology and Automation, dedicated conference on Mechatronic – Efficient computer support for engineering, manufacturing, testing & reliability. 285–292. Stuttgart

Cognard, J. Y.; Dureisseix, D.; Ladevèze, P.; Lorong, Ph. (1996): Expérimentation d’une approche parallèle en calcul de structures. *Rev. Europ. Elém. Finis.* 5, 197–220

Duvaut, G.; Lions, J. L. (1976): *Inequalities in Mechanics and Physics.* Springer: Berlin

Escaig, Y.; Vayssade, M.; Touzot, G. (1994): Une méthode de décomposition de domaines multifrontale multiniveaux. *Rev. Europ. Elém. Finis.* 3, 311–337

Farhat, C. (1988): A simple and efficient automatic FEM domain decomposer. *Comput. Struct.* 28, 579–602

Farhat, C.; Roux, F. X. (1992): An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM J. Sci. Stat. Comput.* 13, 379–396

Farhat, C.; Roux, F. X. (1994): Implicit parallel processing in structural mechanics. In: Oden J. Tinsley (ed): *Computational Mechanics Advances.* North Holland

Farhat, C.; Mandel, J.; Roux, F. X. (1994): Optimal properties of the FETI domain decomposition method. *Comput. Meth. Appl. Mech. Engng.* 115, 365–385

Geist, G. A.; Berguelin, A.; Dongarra, J.; Jiang, W.; Manchek, R.; Sunderam, V. (1994): *PVM 3 user’s guide and manuel reference.* Oak Ridge National Laboratory. Tennessee

Glowinski, R.; Le Tallec, P. (1990): Augmented Lagrangian Interpretation of the Nonoverlapping Schwarz Alternating Method. In: *Domain decomposition methods.* SIAM, Philadelphia, 224–231

Heath, M. T.; Ng, E.; Peyton, B. W. (1991): *Parallel algorithms for sparse linear systems.* *SIAM Rev.* 33, 420–460

Ladevèze, J. (1985): Algorithmes adaptés aux calculs vectoriels et parallèles pour des méthodes de décomposition de domaine. In: *Proceedings of Troisième Colloque Tendances Actuelles en Calcul des Structures,* 893–907. Pluralis

Ladevèze, P. (1992): New advances in the large time increment method. In: Ladevèze, P. and Zienkiewicz, O. C. (ed): *New advances in computational structural mechanics.* 3–21. Elsevier

Ladevèze, P. (1996): *Mécanique non linéaire des structures – Nouvelle approche et méthodes de calcul non incrémentales.* Paris: Hermès

Ladevèze, P.; Lorong, P. (1992): A large time increment approach with domain decomposition technique for mechanical nonlinear problems. In: Glowinski, R. (ed): *Computing Methods in Applied Sciences and Engineering,* 569–578. Nova Science

Lions, P. L. (1990): On the Schwarz Alternating Method III. A variant for nonoverlapping subdomains. In: Chan, T., Glowinski, R., Périaux, J. and Windlun, O. (ed): *Proceedings of Domain Decomposition Methods for Partial Differential Equations.* SIAM, Philadelphia

Mandel, J. (1993): Balancing domain decomposition. *Comm. Num. Meth. Engng.* 9, 233–241

Noor, A. K. (1994): New computing systems, future high performance computing environments and their implications in large-scale-problems. In: Topping, B. H. V. and Papadrakakis, M. (ed): *Advances in parallel and vector processing for structural mechanics,* 1–22. Civil-Comp Press

Pawtucket, R. I.; Hibbitt, Karlsson; Sorensen, Inc. (1994): *ABAQUS/Standard Version 5.4 Manuals*

Simo, J. C.; Laursen, T. A. (1992): An augmented lagrangian treatment of contact problems involving friction. *Comput. Struct.* 42, 97–116

Sunderam, V. S.; Geist, G. A.; Dongarra, J.; Manchek, R. (1994): *The PVM concurrent computing system: Evolution, experiences and trends.* *Parallel Computing.* 20, 531–545

Verpeaux, P.; Charras, T.; Millard, A. (1988): CASTEM 2000 une approche moderne du calcul des structures. In: Fouet, J. M., Ladevèze, P. and Ohayon, R. (ed): *Calcul des structures et intelligence artificielle.* 261–271. Pluralis

Zhong, Z.; Mackerle, J. (1992): Static contact problems – a review. *Engng Comput.* 9, 3–37