



HAL
open science

A multi-level and parallel strategy for structural analysis

David Dureisseix, Pierre Ladevèze

► **To cite this version:**

David Dureisseix, Pierre Ladevèze. A multi-level and parallel strategy for structural analysis. 5th International Conference on Computational Plasticity, Computational Plasticity - Fundamentals and Applications - COMPLAS V, Mar 1997, Barcelone, Spain. pp.1790-1797. hal-00321768

HAL Id: hal-00321768

<https://hal.science/hal-00321768>

Submitted on 12 Oct 2020

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.

A multi-level and parallel strategy for structural analysis

D. Dureisseix and P. Ladevèze

Laboratoire de Mécanique et Technologie de Cachan (LMT)
E.N.S. de Cachan / C.N.R.S. / Université Paris 6
61 Av. du Président Wilson, F-94235 CACHAN CEDEX, FRANCE

Abstract. In order to perform non-linear structural analysis on parallel architecture computers, with shared or distributed memory, an iterative approach related to domain decomposition methods has been designed. The studied structure is seen as the assembly of two mechanical entities: substructures and interfaces. The Large Time INcrement method (LATIN) is then used to build an iterative algorithm, in order to solve the implicit structural problem concurrently [14]. As a first step, the implemented version is concerned here with linear elasticity, in which case our approach is similar to those proposed by [11], [16], [10]. Such a method is a “mixed” approach as both displacements and efforts are the unknowns.

A general drawback of domain decomposition methods consists in the decrease of convergence when increasing the number of substructures. Using a global mechanism to propagate information among all the substructures can avoid this drawback [1]. This mechanism is now implemented on several domain decomposition-like algorithms, such as the FETI method, into which the rigid movement of substructures has to be introduced [8], [9]. We are proposing herein an extension to the previous LATIN algorithm that takes into account two scales introduced when decomposing the structure into substructures. The large scale problem is then used to build a global exchange of information and therefore to improve performance. Moreover, comparisons with different domain decomposition methods, and in particular with the variants of the FETI method, are proposed.

1 INTRODUCTION

The most powerful computers are now based on a parallel architecture [18]. For using such architectures, designing suited algorithms is not obvious, and numerous researches are under development to build “parallel” algorithms [5], [2], [19].

The proposed parallel approach is related to domain decomposition methods [7], [21], [6]. In order to perform a parallel treatment of the problem, the elastic structure is decomposed into substructures Ω^E ($E \in \mathbf{E}$) and interfaces $L^{EE'}$. Each is a mechanical entity with its own behaviour [4].

2 FORMULATION OF THE PROBLEM

2.1 Substructure behaviour

Each substructure Ω^E is submitted to the action of its environment (the neighbouring interfaces): an effort density \underline{F}^E and a displacement field \underline{W}^E on its boundary $\partial\Omega^E$. Eventually, \underline{f}_d is an additional density of body forces which the substructure is submitted to (Figure 1).

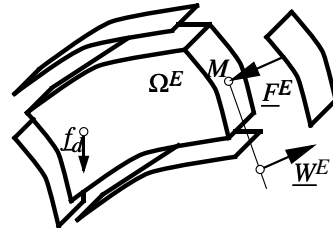


Figure 1. Substructure

For each $E \in \mathbf{E}$, $(\underline{W}^E; \underline{F}^E)$ has to satisfy:

- kinematic equations:

$$\begin{aligned} \exists \underline{U}^E \in \mathcal{U}^E, \\ \varepsilon^E = \varepsilon(\underline{U}^E) \text{ and } \underline{U}_{|\partial\Omega^E}^E = \underline{W}^E \end{aligned} \quad (1)$$

where \mathcal{U}^E is the set of displacement fields defined onto Ω^E which posses a finite energy.

- equilibrium equations:

$\exists \sigma^E$ that equilibrates \underline{f}_d and \underline{F}^E , i.e.

$$\begin{aligned} \forall \underline{U}^* \in \mathcal{U}^E, \\ \int_{\Omega} \text{Tr}[\sigma^E \varepsilon(\underline{U}^*)] d\Omega = \int_{\Omega^E} \underline{f}_d \cdot \underline{U}^* dS + \\ + \int_{\partial\Omega^E} \underline{F}^E \cdot \underline{U}_{|\partial\Omega^E}^* dS \end{aligned} \quad (2)$$

- constitutive relation:

$$\sigma^E = \mathbf{K} \varepsilon^E \quad (3)$$

where \mathbf{K} is Hooke's tensor, which characterise the material behaviour.

σ^E and ε^E are the stress and strain fields, while \underline{U}^E is the displacement field within the substructure Ω^E . The set of unknowns $(\underline{W}^E, \underline{F}^E, \underline{U}^E, \sigma^E)$, for $E \in \mathbf{E}$, that characterises the state of substructures is denoted by \mathbf{s} .

2.2 Interface behaviour

The state of the liaison $L^{EE'}$ between substructures Ω^E and $\Omega^{E'}$ is defined by values on its surface $\Gamma^{EE'}$ of the displacements and efforts $(\underline{W}^E; \underline{F}^E)$ and $(\underline{W}^{E'}; \underline{F}^{E'})$ (Figure 2).

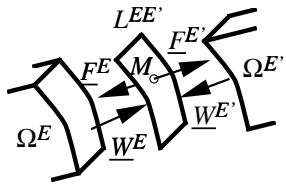


Figure 2. Interface

For instance, for a perfect liaison, they have to satisfy:

$$\underline{F}^E + \underline{F}^{E'} = 0 \quad \text{and} \quad \underline{W}^E = \underline{W}^{E'} \quad (4)$$

Of course, other kinds of liaison can be expressed, such as prescribed effort liaison, prescribed displacement liaison, unilateral contact with or without friction, ..., as described in [13].

2.3 Description of the algorithm

The structure is now considered as an assembly of substructures and interfaces. Following the framework of the LArge Time INcrement method (LATIN) [14], [12], equations are splitted into two groups:

- \mathbf{A}_d is the set of unknowns \mathbf{s} satisfying each substructure behaviour (1), (2), (3);
- Γ is the set of unknowns \mathbf{s} satisfying the interfaces behaviour (4).

The solution \mathbf{s}_{ex} is then the intersection of \mathbf{A}_d and Γ . A two-stage algorithm builds successively an element of \mathbf{A}_d and an element of Γ (Figure 3).

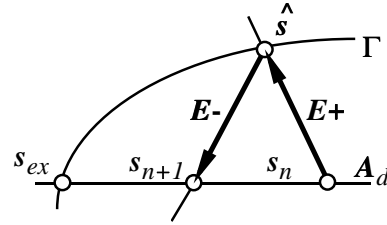


Figure 3. The LATIN 2-stage algorithm

Each stage involves a search direction; they are the parameters of the method:

- the local stage uses the search direction $E+$:

$$(\hat{\underline{F}} - \underline{F}) - k(\hat{\underline{W}} - \underline{W}) = 0 \quad (5)$$

Finding $\hat{\mathbf{s}} \in \Gamma$ such that $\hat{\mathbf{s}} - \mathbf{s}_n$ belongs to the search direction $E+$ is a local problem on interfaces, which is then parallelisable.

- the global stage uses the search direction $E-$:

$$(\underline{F} - \hat{\underline{F}}) + k(\underline{W} - \hat{\underline{W}}) = 0 \quad (6)$$

The problem is then to find $\mathbf{s}_{n+1} \in \mathbf{A}_d$ such that $\mathbf{s}_{n+1} - \hat{\mathbf{s}}$ belongs to E^- . It leads to a problem which is global only per substructure, though it can be solved in parallel once the substructures are distributed among the available processors as well as their related interfaces.

In the case of linear elasticity, this algorithm is similar to those proposed by [11], [16], [10]. For a continuum point of view, the previous algorithm converges toward the solution of the ‘‘reference problem’’ without decomposition. Precise convergence conditions are detailed in [13].

3 IMPLEMENTATION OF THE ALGORITHM

When implementing the algorithm in a finite element code, we need to discretise the unknowns. A first approach is to treat the problem with a displacement-oriented formulation. In this case, the displacement \underline{U}^E is discretised according to the mesh of the substructure Ω^E , while \underline{W}^E is the trace of \underline{U}^E on the boundary $\partial\Omega^E$. Let \mathcal{U}_h^E be the corresponding space of discretisation of \underline{U}^E . The displacement formulation of the global stage is then:

$$\min_{\underline{U}^E \in \mathcal{U}_h^E} J(\underline{U}^E) \quad (7)$$

where

$$\begin{aligned} J(\underline{U}^E) = & \frac{1}{2} \int_{\Omega^E} \text{Tr}[\varepsilon(\underline{U}^E) \mathbf{K} \varepsilon(\underline{U}^E)] d\Omega + \\ & + \frac{1}{2} \int_{\partial\Omega^E} \underline{U}^E \cdot k \underline{U}^E dS - \int_{\Omega^E} \underline{f}_d \cdot \underline{U}^E d\Omega + \\ & - \int_{\partial\Omega^E} (\underline{F}^E + k \underline{W}^E) \cdot \underline{U}^E dS \quad (8) \end{aligned}$$

We can notice here that the problem to solve is global on the substructure and is an elasticity-like one. After discretisation, it can be written:

$$([K^E] + [k^E])[q^E] = [f^E] \quad (9)$$

where $[K^E]$ is a classical rigidity matrix and $[k^E]$ an additional rigidity on the boundary, traducing the search direction. Both are constant all along the iterations. \underline{W}^E is then the

trace of \underline{U}^E on the boundary and \underline{F}^E is back-substituted thanks to the search direction:

$$\underline{F}^E = \underline{F}^E + k \underline{W}^E - k \underline{W}^E \quad (10)$$

The algorithm has been implemented within the industrial-like finite element code CASTEM 2000 [20].

4 A MULTI-LEVEL EXTENSION

A general and well-known behaviour of domain decomposition approaches is to have a degrading convergence rate when increasing the number of substructures [1].

For instance, let us consider the model example of a two dimensional beam with a parabolic bending loading (Figure 4). The reference $(\underline{U}; \sigma)_{\text{ref}}$ is here the direct finite element solution without decomposition. It allows us to define the convergence rate related to error in energy:

$$\begin{aligned} \tau = & -\log \frac{e_{n+1}}{e_n}, \quad \text{where} \\ e_n^2 = & \frac{\int_{\Omega} \text{Tr}[(\sigma_n - \sigma_{\text{ref}}) \mathbf{K}^{-1} (\sigma_n - \sigma_{\text{ref}})] d\Omega}{\int_{\Omega} \text{Tr}[\sigma_{\text{ref}} \mathbf{K}^{-1} \sigma_{\text{ref}}] d\Omega} \end{aligned}$$

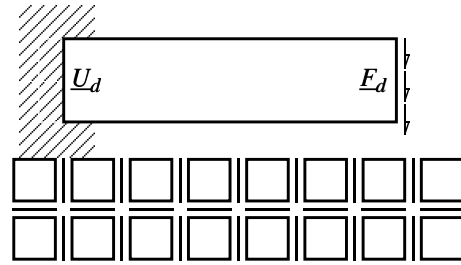


Figure 4. Model problem and example of decomposition into 16 substructures and interfaces

The form of the search direction parameter k has been chosen to be $k = k_0 I_d$ where I_d is the identity operator and $k_0 = E/L_0$, E is the Young modulus of the material and L_0 a length that characterises the search direction. For the former one-level LATIN approach, the ‘‘optimal’’ value of L_0 is related to the length of the structure (typically L_0 is twice the length for such a bending problem) [4]. The figure 5 reports in particular the averaged convergence rate (up to convergence:

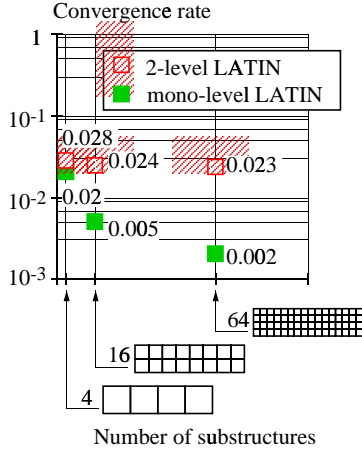


Figure 5. Convergence rate versus number of substructures

$e_n \leq 1\%$) versus the number of substructures.

To remedy to such a drawback, a global exchange of information is generally efficient [7], [17]. In order to build such a mechanism, we choose to express the solution onto two different scales:

$$(\underline{U}^E; \sigma^E) = (\underline{U}_1^E; \sigma_1^E) + (\underline{U}_2^E; \sigma_2^E) \quad (11)$$

where 1 and 2 denote respectively effective unknowns on the large scale and corrections on the fine scale. Each scale can arise from a different modelling of the structure, for instance here, from two meshes with embedded elements. Let Ω_1 and Ω_2 denote them. Transfers between them is performed through a prolongation and a restriction operators, respectively \mathbb{P} and $\mathbb{R} = \mathbb{P}^T$. Transposition is derived from the symmetric form:

$$\begin{aligned} \sum_{E \in \mathbf{E}} \int_{\partial\Omega^E} \text{Tr}[\varepsilon(\mathbb{P}^E \underline{U}) \sigma^E] d\Omega = \\ = \int_{\partial\Omega_1} \text{Tr}[\varepsilon(\underline{U}) \sum_{E \in \mathbf{E}} \mathbb{R}^E \sigma^E] d\Omega \end{aligned} \quad (12)$$

Where $(\underline{U}, \mathcal{T})$ denotes the effective solution, i.e. the solution defined onto the mesh Ω_1 (then $\underline{U}_1^E = \mathbb{P}^E \underline{U}$ and $\mathcal{T} = \sum_{E \in \mathbf{E}} \mathbb{R}^E \sigma_1^E$).

When using two embedded grids, prolongation can be performed with a hierarchical finite element projection [23], [22].

With such a splitting, the problem is now expressed separately on both scales: the equilibrium equations become

- on the fine scale 2, on each substructure Ω_2^E

$$\begin{aligned} \forall \underline{U}^* \in \mathcal{U}_2^E, \\ \int_{\Omega^E} \text{Tr}[\sigma_2^E \varepsilon(\underline{U}^*)] d\Omega = \int_{\Omega^E} \underline{f}_d \cdot \underline{U}^* d\Omega + \\ + \int_{\partial\Omega^E} \underline{F}_2^E \cdot \underline{U}^* dS + \\ - \int_{\Omega^E} \text{Tr}[\varepsilon(\mathbb{P}^E \underline{U}) \mathbf{K}_\varepsilon(\underline{U}^*)] d\Omega \end{aligned} \quad (13)$$

\mathcal{U}_2^E is the part of \mathcal{U}^E corresponding to the fine scale 2.

- on the large scale 1

$$\begin{aligned} \forall \underline{U}^* \in \mathcal{U}_1 \\ \int_{\Omega_1} \text{Tr}[\mathcal{T} \varepsilon(\underline{U}^*)] d\Omega = \int_{\Omega_1} \underline{f}_d \cdot \mathbb{P} \underline{U}^* d\Omega + \\ - \int_{\Omega_1} \text{Tr}[\varepsilon(\underline{U}^*) \sum_{E \in \mathbf{E}} \mathbb{R}^E \mathbf{K}_\varepsilon(\underline{U}_2^E)] d\Omega \end{aligned} \quad (14)$$

where \mathcal{U}_1 is this time related to fields defined on the whole structure Ω satisfying homogeneous boundary condition.

One can notice that the two scales are not separated, due to coupling terms: on the large scale problem, the stress field \mathcal{T} has to equilibrate also $-\sum_{E \in \mathbf{E}} \mathbb{R}^E \mathbf{K}_\varepsilon(\underline{U}_2^E)$; on the fine scale, the stress field σ_2^E has to equilibrate also $-\mathbf{K}_\varepsilon(\mathbb{P}^E \underline{U})$.

The large scale problem being chosen to remain global, we search successively the solution on the two scales at each iteration of the LATIN method on the substructured fine scale.

On the previous example of a two dimensional beam, the convergence rate gain is illustrated on figure 5 when each substructure is meshed with 2048 6-node elements for each substructuring (4, 16 and 64 substructures). The reference problems possess respectively 33 400, 132 354 and 526 850 d.o.f. The computations have been realised on a CRAY-T3D at the IDRIS in Orsay, France.

The convergence rate quasi-independence of the number of substructures illustrates the numerical scalability of the multi-level LATIN approach. A drawback is the necessity to solve a global problem on the whole structure due to the large scale. Here, this scale

has been meshed with a number of 3-node elements equal to twice the number of substructures.

One can notice that the optimal search direction for this example when using the large scale, is now related to the length of the interfaces (L_0 was chosen to be 0.25 times the length of one substructure). It is then no more characterised by the global structure behaviour.

5 COMPARISONS

With the proposed approach, one can remark that all over the iterations, nor displacement neither efforts of \mathbf{s}_n are continuous through interfaces but both gaps are improved.

Some other approaches privilege a continuous displacement, as primal Schur method [15] does. At each iteration, it proposes a solution that satisfies displacement continuity throughout interfaces, and improves effort gap with a conjugate gradient algorithm on the displacement-condensed problem.

On the other hand, the dual Schur method, [7], proposes a solution that satisfies normal stress continuity and improves displacement gap along the conjugate gradient iterations on the effort-condensed problem. More precisely, with the FETI method, the current solution must satisfy the global equilibrium onto each subdomain, and the displacement is known via the determination of an additive rigid solid movement. This leads to the resolution of a global problem onto all the rigid solid movements of all the subdomains. This is the equivalent of the large scale problem.

The proposed example is this time a three dimensional elastic beam with bending. Two meshes with 20-node elements are considered for this problem on the figure 6. For the large scale problem, the influence of the discretisation with 8-node elements is studied (Figure 7).

The figure 8 reports the error in energy e_n versus iterations for the primal SCHUR algorithm, with no preconditioner, for the FETI method without preconditioner, and with Lumped and Dirichlet one. These last three have been performed by F.-X. Roux with a PARAGON at ONERA. The former LATIN algorithm and multi-level exten-

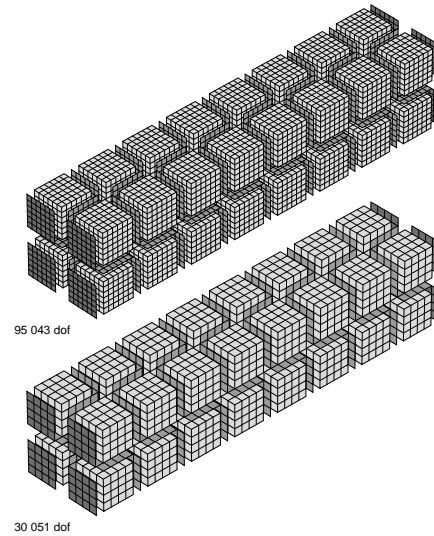


Figure 6. Meshes of the substructured problem

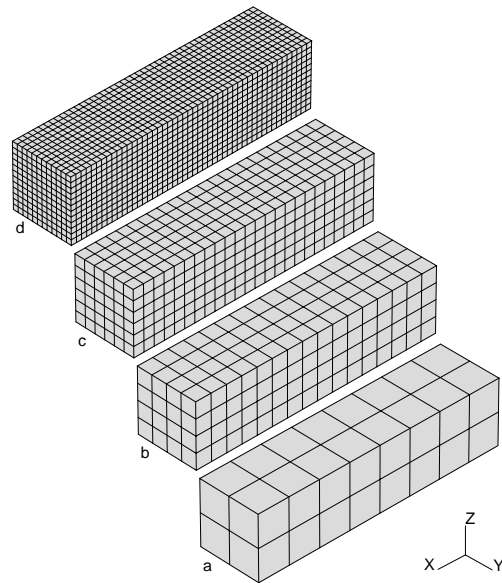


Figure 7. Meshes of the large scale problem

sions related to several discretisations of the large scale are also reported. These have been done on the CRAY-T3D at IDRIS. Both multiprocessors use 32 processors.

As time comparisons between two different approaches depend on processor, communication network, compiler, programming environment, disk usage, . . . , we decide to only obtain principal tendencies of algorithms by weighting previous results: after analysing costly parts of simulations, we identify both

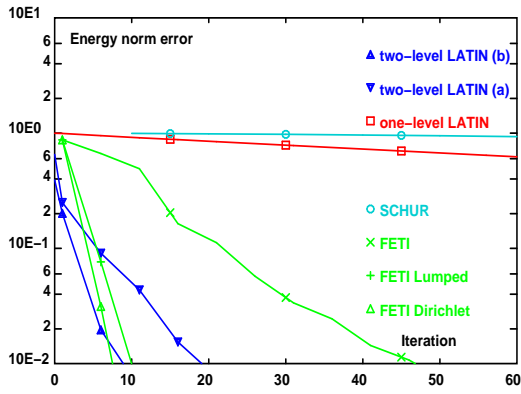


Figure 8. Comparison of several approaches for the first test

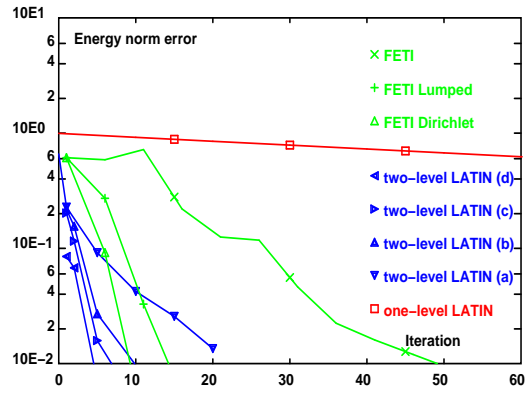


Figure 10. Comparison of several approaches for the second test

FETI and one-level LATIN initialisations to 1 in CPU equivalent time (cumulated on the 32 processors). Afterwards, both FETI and two-level (a) LATIN iteration are identified in terms of cost. Figure 9 then reports the error versus this CPU equivalent time. Figures 10

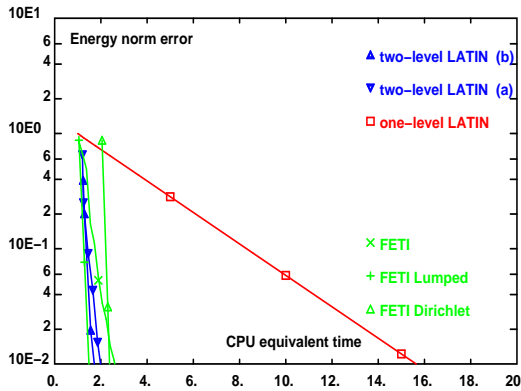


Figure 9. Comparison of costs for the first test

and 11 are similar for the second mesh of the reference problem.

The cost of direct finite element analysis of the two problems are, respectively with their own CPU equivalent time, 6.5 for the 30 051 d.o.f. problem and 18 for the 95 043 d.o.f. problem (Figure 6): a substructuring approach is more and more efficient as the size of the problem increases. Moreover, for the LATIN approaches, convergence rate seems to be weakly dependent on mesh size, even less than FETI Dirichlet version (Figures 8 and 10).

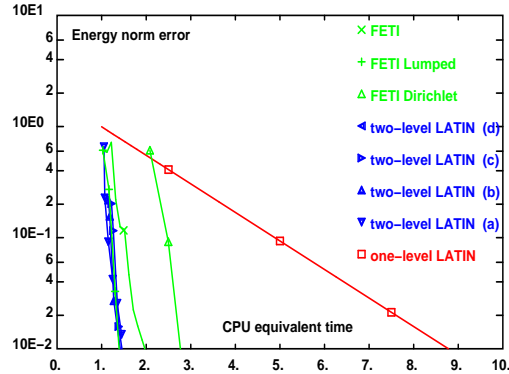


Figure 11. Comparison of costs for the second test

6 CONCLUSIONS

The originality of the coupling of the Large Time INcrement method with a substructuring technique is the major role played by interfaces which are considered as structures on their own.

It conducts to a “pure parallel” algorithm that can be improved by introducing a two-level scheme. The consequence is the arising of a global problem to solve onto the whole structure at each iteration. It conducts to an algorithm which is numerically scalable; moreover it is weakly dependent on the refinement of the mesh.

The final goal of the approach is the extension to non-linear structural analysis, following the LATIN design.

ACKNOWLEDGEMENTS

The authors wish to thank F.-X. ROUX from the ONERA at Châtillon, France, for having performed the computations about FETI approach on the PARAGON, and the IDRIS at Orsay, France, for accessing their CRAY-T3D.

REFERENCES

- [1] J. H. Bramble, J. E. Pasciak, and A. H. Schatz. The construction of preconditioners for elliptic problems by substructuring, I. *Math. Comp.*, 47(175):103–134, 1986.
- [2] J. J. Buoni, P. A. Farrell, and A. Ruttan. Algorithms for LU decomposition on a shared memory multiprocessor. *Parallel Computing*, 19:925–937, 1993.
- [3] T. F. Chan, R. Glowinski, J. Périaux, and O. B. Widlund, editors. *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1990. SIAM.
- [4] J.-Y. Cognard, D. Dureisseix, P. Ladevèze, and P. Lorong. Expérimentation d’une approche parallèle en calcul de structures. *Revue Européenne des Éléments Finis*, 5(2):197–220, 1996.
- [5] O. Débordes and J. C. Michel. Parallélisation des problèmes non-linéaires. In J.-M. Fouet, P. Ladevèze, and R. Ohayon, editors, *Calcul des Structures et Intelligence Artificielle*, volume 4, pages 223–232. Pluralis, 1989.
- [6] Y. Escaig, G. Touzot, and M. Vayssade. Parallelization of a multilevel domain decomposition method. *Computing Systems in Engineering*, 5(3):253–263, 1994.
- [7] C. Farhat, J. Mandel, and F.-X. Roux. Optimal convergence properties of the FETI domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 115:365–385, 1994.
- [8] C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32:1205–1227, 1991.
- [9] C. Farhat and F.-X. Roux. Implicit parallel processing in structural mechanics. In J. T. Oden, editor, *Computational Mechanics Advances*, volume 2. North-Holland, June 1994.
- [10] R. Glowinski and P. Le Tallec. Augmented lagrangian interpretation of the nonoverlapping Schwarz alternating method. In Chan et al. [3], pages 224–231.
- [11] J. Ladevèze. Algorithmes adaptés aux calculs vectoriel et parallèle pour des méthodes de décomposition de domaines. In J. P. Grellier and G. M. Campel, editors, *Actes du 3ème colloque Tendances Actuelles en Calcul de Structures*, pages 893–907, Bastia, Nov. 1985. Pluralis.
- [12] P. Ladevèze. New advances in large time increment method. In P. Ladevèze and O. C. Zienkiewicz, editors, *New advances in computational structural mechanics*, pages 1–18. Elsevier, 1991.
- [13] P. Ladevèze. *Mécanique non-linéaire des structures — Nouvelle approche et méthodes de calcul non incrémentales*. Hermès, Paris, 1996.
- [14] P. Ladevèze and P. Lorong. A large time increment approach with domain decomposition technique for mechanical non linear problems. In R. Glowinski, editor, *Comput. Meths. Appl. Sc. Engng.*, pages 569–578, New York, 1992. INRIA, Nova Science.
- [15] P. Le Tallec. Domain decomposition methods in computational mechanics. In *Computational Mechanics Advances*, volume 1. North-Holland, 1994.
- [16] P.-L. Lions. On the Schwarz alternating method III: a variant for nonoverlapping

- subdomains. In Chan et al. [3], pages 202–223.
- [17] J. Mandel. Balancing domain decomposition. *Communications in Applied Numerical Methods*, 9:233–241, 1993.
- [18] A. K. Noor. New computing systems, future high performance computing environments and their implications on large-scale problems. In B. H. V. Topping and M. Papadrakakis, editors, *Advances in parallel and vector processing for structural mechanics*, pages 1–22, Edinbourg, Scotland, 1994. Civil-Comp Press.
- [19] V. Pan and J. Reif. Fast and efficient parallel solution of sparse linear systems. *SIAM J. Sci. Comput.*, 22(6):1227–1250, 1993.
- [20] P. Verpeaux, T. Charras, and A. Millard. CASTEM 2000 : une approche moderne du calcul des structures. In J.-M. Fouet, P. Ladevèze, and R. Ohayon, editors, *Calcul des Structures et Intelligence Artificielle*, volume 2, pages 261–271. Pluralis, 1988.
- [21] G. Yagawa, A. Yoshioka, S. Yoshimura, and N. Soneda. A parallel finite element method with a supercomputer network. *Computers & Structures*, 47:407–418, 1993.
- [22] H. Yserentant. On the multi-level splitting of finite element spaces. *Num. Math.*, 49:379–412, 1986.
- [23] O. C. Zienkiewicz, J. P. De S. R. Gago, and D. W. Kelly. The hierarchical concept in finite element analysis. *Computers & Structures*, 16(1–4):53–65, 1983.