EFFICIENT NONLINEAR DOMAIN DECOMPOSITION METHODS FOR THE POST-BUCKLING ANALYSIS OF LARGE AERONAUTICAL STRUCTURES

P. Cresta, S. Guinard
EADS Corporate Research Center (CRC), Suresnes, France

O. Allix, C. Rey
Laboratory of Mechanics and Technology (LMT) - Ecole Normale Supérieure de Cachan, France

Performing nonlinear analysis of large structures at a fine scale is one of the industrial challenges of our times. In aeronautical structures such as airplane fuselages the main nonlinearity that arises is local buckling of slender elements. During the structural tests of certification achieved on complete fuselages of aircraft, we observe local buckling of the skin between the stiffeners. At increasing loads, these nonlinear areas can expand and provoke redistributions of stresses in the structure. For workloads usually met in service, these phenomena are reversible, the material staying in the elastic domain. However, they can provoke concentrations of stress near the bases of stiffeners and may be at the origin of local damages leading to global failure. Because of the large number of degrees of freedom ($\sim 10^7$-10^8 degrees of freedom (d.o.f.)) and memory and processors limits, computing this kind of problem with a classical technique is not feasible at the moment. For this reason, engineers have developed approximate methods based on super-elements (linear condensation of inner degrees of freedom) or global-local analysis, constituted in a coarse and linear global calculation followed by fine nonlinear analyses on areas of local interest. If such methods allow for certain computations, they cannot take into account phenomena like stress redistribution or nonlinearity expansion, since they only offer a limited dialogue between the introduced scales. As a result, they can only treat localized nonlinearities having no influence on the global response and thus must be used with care.

However, new advances in computer science, using parallel machines and PC clusters, today enable us to spread computational costs and memory requirements over many processors. Increasingly complex problems are becoming calculable, provided the solving method can take advantage of the inherent parallelism of the hardware architecture. In this study, we explore the possibilities related to multiscale and parallel approaches in the context of geometrically nonlinear problems. The study is part of a long-term collaboration with Airbus intended to integrate numerical tools into the design and certification process. Its goal is to develop high-granularity methods for

nonlinear problems, data exchanges still being relatively time-consuming in cluster architectures.

The work is based on the classical domain-decomposition methods (Balancing Domain Decomposition method (BDD), Finite Element Tearing and Interconnecting methods (FETI)…), which have been thoroughly studied in the past decade, mainly for calculation of linear systems (see [1], [2] and [3]). They rely on Schur non-overlapping decomposition of the structures and Krylov iterative solvers. These Domain Decomposition Methods have also been used as the linear solver of a Newton-Raphson method to solve large-scale nonlinear problems in [2]. However, they proved to have rather poor convergence for unbalanced nonlinear effects, i.e. when the nonlinearity has a significant spatial variation (see [4]).

Following some of the ideas developed in the LATIN method (see [5]), we aimed at concentrating the computational efforts on the sensitive zones of the structure while reducing communications. We developed new methods to take advantage of the locally nonlinear behaviour of the structure (local buckling), which is a typical unbalanced nonlinearity. The first strategy uses local nonlinear iterations per substructure with fixed displacement conditions at the boundaries after each global resolution. The second method explores the possibilities offered by mixed conditions at the boundaries. Elementary cases such as beam-frame structures are used to compare calculation efficiency of both strategies.

Formulation and classical methods of resolution

This paper deals with geometrical nonlinearities due to large displacements and rotations of slender structures made of beams, plates and shells. The materials' behavior is supposed to remain linear, and deformations should be small.

In order to take into account large displacements and rotations we applied a corotational formulation with a consistent tangent operator, which is known to be well adapted to these kinematical assumptions [6].

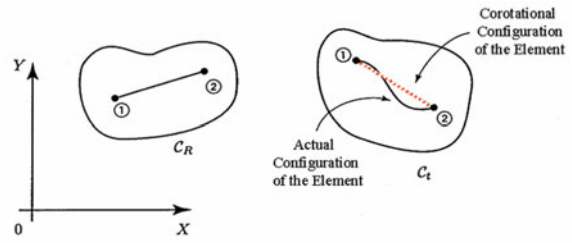

Fig. 1: Definition of the different configurations for a beam element.

Displacements $\tilde{v}$ and efforts $\tilde{f}$ are written in a rotating frame attached to elements (see figure 1). They are connected to each other by the classical Hooke elementary operator of the linear elasticity, expressed in the same frame. Efforts are then balanced by a projector P and brought into the initial configuration by the rotation R:

$$f = R^T P^T \tilde{K}_e \tilde{v}$$

Differentiating this expression, we obtain the coherent tangent operator K_T, which can be expressed under the following form [6]:

$$K_T = R^T (P^T \tilde{K}_e H P - F_{mn} G - G^T F_n^T P) R$$

where H, F_{mn}, G, F_n are terms clarified in [6] contributing to the geometrical stiffness of the prestressed structure. This formulation can be applied to any type of element in a fairly simple and independent way. It also allows us to avoid problems that would occur when applying classical lagrangian formulations on cases with important rotations.

The obtained nonlinear problem is generally solved by the Newton-Raphson method or one of its variants (secant method, etc.). In every case, the system is linearized around a known position and the problem is reduced to the iterative resolution of linear systems of the following type:

$$K_T \Delta u = r$$

where K_T is the tangent operator and r the residue.

The convergence of these algorithms is quadratic but it depends on the size of the loading increment. We thus operate this method within the framework of an incremental process. For a classical Newton-Raphson method with effort loading, we adapt the size of the current step according to the convergence of the previous one. In case of non-convergence the steps' size is halved, and after several successive convergences the size is increased.

In order to process problems with a large number of d.o.f., the linear systems we obtained through linearization of the initially nonlinear problem are solved using efficient parallel methods. Most of these methods are based on a domain decomposition of the structure into sub-domains that can be distributed on various processors [3]. Together, the Schur complements of the tangent operators in each substructure form a global linear problem linking all boundary unknowns. This kind of problem is generally worked out using an iterative Krylov solver, which does not require computation of Schur complements or assemblage of a global system. Thereby, this solver can easily be implemented on parallel computers.

Newton schemes, Schur domain decomposition methods and Krylov solvers together form what is called the family of Newton-Krylov-Schur (NKS) methods. An implementation with the FETI method is proposed in [2], with the efforts on the boundary as the unknown field. We also implemented a classical domain decomposition method to serve as a reference method for this study. We opted herein for a primal (BDD) version by choosing the displacement at the interfaces as unknowns. The following frame illustrates the three main steps for the resolution of a linear system by a primal domain decomposition method (the *i* letter indicating the internal unknowns, *b* the unknowns on the boundaries):

1-*In each substructure (s), condensation of the inner d.o.f. for the tangent operator and the residue (actually, never assembled)*:

$$\begin{cases} S_T^{(s)} = K_{Tbb}^{(s)} - K_{Tbi}^{(s)} K_{Tii}^{(s)^{-1}} K_{Tib}^{(s)} \\ b^{(s)} = r_b^{(s)} - K_{Tbi}^{(s)} K_{Tii}^{(s)^{-1}} r_i^{(s)} \end{cases}$$

2-*Resolution by an iterative method of the problem, condensed over the boundary dof*:

$$S_T^{global} \Delta u_b = b$$

3-*Localization of internal displacements in each substructure*:

$$\Delta u_i^{(s)} = K_{Tii}^{(s)^{-1}} (f_i^{(s)} - K_{Tib}^{(s)} \Delta u_b^{(s)})$$

Each of these steps can be solved in parallel, which allows us to spread computational costs over several processors. However the resolution step (2) requires data transfer for each iteration of the Krylov solver, which can penalize global performances of the algorithm, in particular for cluster architectures having slower data transfer rates. In [7] we proposed improvements of the iterative Krylov solvers by selective re-use of former Krylov bases in order to reduce the number of iterations and, consequently, the number of data exchanges. In return, the need for larger data storage space presents itself.

Definition of a Test structure

In the phase of numerical development we first of all analyze the behavior of assemblies of beams, in order to test the methods on representative problems of low complexity. Figure 2 presents the test structure retained for this study. It is a frame structure made of Navier-Bernoulli-type beam elements, on which we impose a global bending force. It is composed by a set of 10 substructures, each having a few hundreds of inner d.o.f. and 12 boundary d.o.f. We can observe a rather soft global nonlinearity, due to large displacements of the structure, as well as a stronger buckling nonlinearity, lo-

calized on beam elements near the prescribed displacements zone. This structure presents a behavior similar to the aeronautical cases in which we are interested and it has the advantage of the simplicity for developing and testing various methods before trying them on larger and more complex industrial structures.

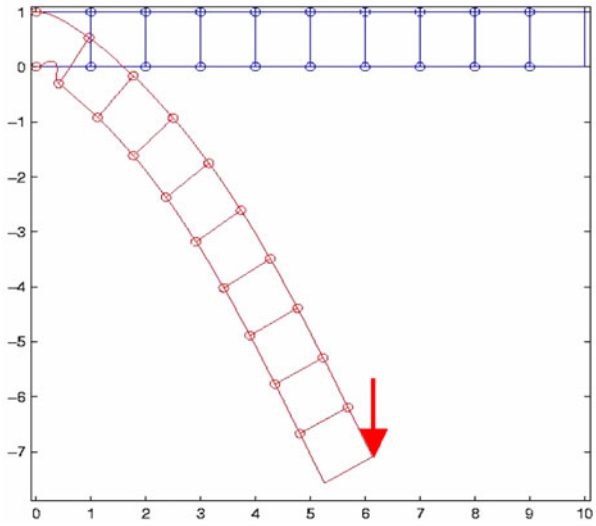

Fig. 2: Studied frame structure.
Initial and deformed configurations.

The convergence of a classical Newton algorithm for this problem turns out to be pretty poor [4], the loading step requiring to be reduced in order to go through the local buckling instability. The domain decomposition does not modify the convergence result. It only enables to distribute the computational efforts uniformly on the whole structure. Actually, these methods cannot take advantage of the local character of the nonlinearity while global convergence is entirely controlled by local phenomena, which impose a large number of iterations for the whole structure.

A two-level primal approach

The analysis of classical Newton-Krylov-Schur methods leads us to propose a new method that allows for better treatment of local buckling nonlinearities. After each global analysis of the condensed problem on the boundaries, a nonlinear analysis is carried out on each substructure, imposing displacements on the interfaces between the substructures. The only adaptation needed with regard to the classical method lies in the relocation step (3) of the internal unknowns. In this new version, we make several local iterations (by substructure) updating the tangent operators which are connected to the geometry, until we obtain a given residue ε_{loc}. Step (3) can then be written as follows:

$$\Delta u_i^{(i)} = K_{Tii}^{(i)^{-1}}(u)(f_i^{(i)} - K_{Tib}^{(i)}(u)\Delta u_b^{(i)})$$

In each relocation step we compel the substructures to verify the nonlinear behavior and equilibrium. The purpose here is to reduce the number of global iterations of the algorithm by concentrating the computational effort at the local level. The method can be seen as an extension for nonlinear problems of the "super-elements" technique, that is generally used for linear calculations.

Figures 3, 4 and 5 present comparative results for various strategies. We obtain the first sequence of results (a) by a classical primal Newton-Krylov-Schur method with a maximal initial loading step (a greater step leading to the divergence of the algorithm). The second series (b) is obtained by a two-level primal approach with the same initial loading step, while the third one (c) is obtained by determining the new maximal initial load for this method. Other parameters (convergence criteria,…) are selected in an identical way.

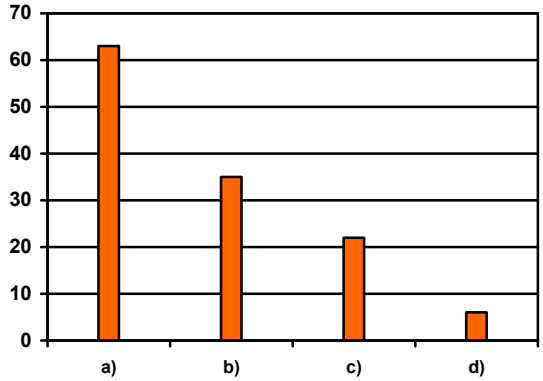

Fig. 3: Number of global iterations for different numerical strategies

On figure 3 we observe a decreasing number of global calculations and thus data transactions when using the method mentioned above. This is in part due to larger acceptable loading steps (see figure 5) and better convergence. In particular, the method goes through the buckling instability more easily than with the classical approach, without the need to decrease the loading step.

In compensation, the number of local computations after each global step is increased. Nevertheless, the decrease in global iterations allows for fewer local analyses for the entire computation. Moreover, we did observe an elevated number of local calculations in substructures presenting strong nonlinear behavior. Computational efforts are thus concentrated on these nonlinear areas. In return, the strategy is no longer symmetric and load balancing might be required for efficient parallel implementation.

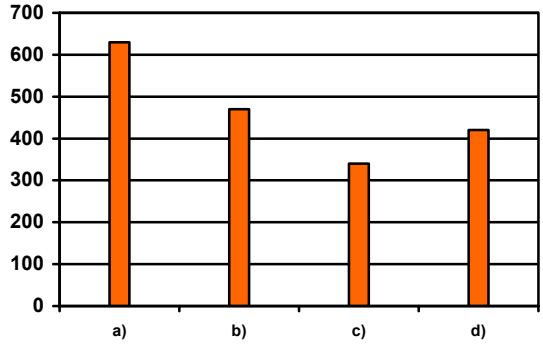

Fig. 4: Total number of local iterations within substructures

Influence of local criterion

To determine the influence of the local convergence criterion on the methods' performance, a parametric study was carried out. We fixed the global convergence criterion of the Newton scheme $\varepsilon_{global}=10^{-6}$ and tried several values for the local criterion, from 1 to 10^{-6}. In the results presented in table 1 it appears that for this criterion there is an optimal local threshold for which computational costs are minimal. The number of global computations indeed decreases with the criterion and reaches a minimum threshold, while the number of local calculations first decreases and then increases again, without any positive influence on the global performances. In practice, a quite high local convergence criterion, about the square root of the global convergence criterion is sufficient to obtain good results.

Table 1

Influence of the local convergence criterion on the algorithm performance

Local convergence criterion	Number of global computations	Number of local computations
$\varepsilon_{loc}=10^{0}$	63	630
$\varepsilon_{loc}=10^{-2}$	45	471
$\varepsilon_{loc}=10^{-3}$	35	470
$\varepsilon_{loc}=10^{-4}$	35	482
$\varepsilon_{loc}=10^{-6}$	35	523

A two-level mixed approach

In order to soften boundary conditions of local computations, we investigated the possibilities offered by mixed conditions at the borders. We chose here to work with both displacements and efforts on the interfaces Γ between substructures. To find the solution we perform the iterative solving of two groups of equations, described hereafter.

The first step of the method consists of nonlinear computations by substructure. These computations can be carried out on separate processors without any data exchange. Like in the method mentioned before, this step permits to locally treat nonlinearities and thus ensures that computational costs are concentrated where they are needed. The difference with the primal versions lies in the boundary conditions that are here Robin (mixed) ones.

The second step consists of a large linear system of equations governing the boundary

d.o.f. It is solved iteratively using the same methods as in the second step of classical domain decomposition methods. Its goal is to ensure the admissibility of boundary unknowns, namely equilibrium of efforts and continuity of displacements, while transmitting the mechanical information to the entire structure, thus giving the needed scalability to the algorithm.

The new algorithm can be summarized as follows:

Step 1: Knowing $\hat{F}$ and $\hat{U}$ at the interfaces, we look for the *f* and *u* fields by substructure that verify the nonlinear behavior and the mixed limit conditions (*k* is a parameter of the method):

$$\begin{cases} f = K(u).u \\ (F - \hat{F}) + k(U - \hat{U}) = 0 \end{cases}$$

Step 2: Knowing $F = f|_{\Gamma}$ and $U = u|_{\Gamma}$ we look for $\hat{F}$ and $\hat{U}$ verifying the interface fields admissibility (i.e. continuity of displacements and equilibrium of efforts) by taking the condensed tangent operator as research direction by substructure:

$$\begin{cases} \hat{U}_1 = \hat{U}_2 \\ \hat{F}_1 + \hat{F}_2 = 0 \\ S_T^{(s)}(\hat{U}^{(s)} - U^{(s)}) = \hat{F}^{(s)} - F^{(s)} \end{cases}$$

which leads to the following global system:

$$\begin{cases} S_T^{global}\hat{U} = \sum_s A^{(s)}(S_T^{(s)}U^{(s)} - F^{(s)}) \\ \hat{F}^{(s)} = F^{(s)} + S_T^{(s)}(\hat{U}^{(s)} - U^{(s)}) \end{cases}$$

At the convergence of the algorithm, we obtain a solution verifying all the equations of the problem. One of the advantages of this method with regard to the previous one is its ability to relax the constraints imposed on the local problems by the boundary conditions. For this reason it can deal with larger load increments at the global level, and reduce in the same time the amount of exchanged data.

Analysis of Results

Tests performed on the previous structure with this last method (case d of figures 3, 4 and 5) show a drop in the number of global iterations, which only amount to 6 linear calculations in case d) compared with 22 for case c) of figure 3 (optimal case of the primal method). This result was attained with a one-step global loading, while 8 loading steps were necessary in the case a), 6 in case b) and 5 in the case c).

Nevertheless, a local incremental strategy for the loads is required to solve the equations in each substructure, thus increasing substantially the number of local iterations, not exceeding 500 iterations (figure 4-d). The local incremental scheme used in this version may not be sufficient but it can be easily improved in the future. The method thus turns out to be very interesting and opens the path toward large loading increments.

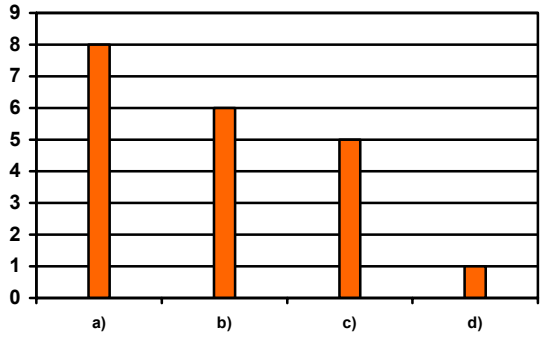

Fig. 5: Comparison of the numbers of loading increments

The *k* parameter defining boundary conditions plays an important role in the convergence rate. Notice that, when *k* tends to infinity, the mixed method turns into the primal method. On the other hand, a zero parameter transforms the method into a dual version where the only unknowns at the interfaces are efforts. In that case, it becomes necessary to correctly process rigid body movements that may appear in substructures without any Dirichlet condition. The *k* parameter thus appears to be an additional variable that enable us to optimize the method, the primal and dual versions only being limit cases. A classical choice for *k* is the tangent operator of the neighboring substructures, condensed on the interfaces. However this operator is very expensive to compute and to exchange between substructures. Consequently it is preferable to opt for less complex operators, in

spite of a loss of convergence results. Testing with the diagonal of K_{Tbb} has provided reasonably good results (9 global iterations) with very low computing costs and a short transfer time.

Conclusions and Perspectives

First we brought forward the problem of nonlinear calculation of large structures and currently used methods, and then we presented two new approaches. These approaches are based on a will of processing nonlinearities at a local scale. The obtained results show a significant improvement compared with classical Newton-Krylov-Schur methods, being the reference strategy, and, foremost, a better granularity, with fewer inter-process exchanges. They allow concentrating the computational efforts on the effectively nonlinear areas while reducing the number of global calculations and thus the quantity of exchanged data. The proposed method are particularly well adapted to the PC cluster architectures, which are penalized by data exchanges between processors. However, due to the local nonlinear computations, the symmetry of classical domain decomposition methods is lost. As a result, an efficient parallel implementation of the method needs a dynamic load-balancing scheme in order to minimize waiting times between processes.

A C++ parallel version for shell structures is under development, the goal being to validate the methods with the nonlinear calculation of a structure close to a documented industrial case (such as static test of a fuselage section of Airbus airplane for example).

Acknowledgements

This research is supported by EADS Corporate Research Center.

References

[1] Le Tallec P., Mandel J., Vidrascu M. A Neumann-Neumann domain decomposition algorithm for solving plate and shell problems. *J. Numer. Anal.* , Vol. 35, No. 2, 836-867, 1998.

[2] Farhat C., Pierson K., Lesoinne M. The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically nonlinear structural analysis problems, *Comput. Meth. In Appl. Mech. and Engrg*., 18, 333-374, 2000.

[3] Fragakis Y., Papadrakakis M. A unified framework for formulating domain decomposition methods in structural mechanics, Technical Report, 2002.

[4] Cai X.-C., Keyes D.E. Nonlinearly preconditioned inexact Newton algorithms, SIAM *J. Sci. Comput*., 24, 183-200, 2002.

[5] Ladevèze P. Nonlinear computational structural mechanics - new approaches and non-incremental methods of calculation, Springer Verlag, Berlin, 1999.

[6] Haugen B., Felippa C. *A unified formulation of co-rotational finite elements (Theory & Applications),* Dpt. of Aerospace Engrg. Sciences, Univ. Colorado at Boulder, CAS Report No. CU-CAS-95-06, 1995.

[7] Rey C., Gosselet P. Solution to large nonlinear systems : acceleration strategies based on domain decomposition methods and reuse of Krylov-subspaces, *6th International ESAFORM Conference on Material Forming*, Salerne (Italie), Nuova Ipsa Editore, 2003.