57

Parallel Implementation of the Two-level FETI Method

François-Xavier Roux and Charbel Farhat

1 Introduction

Recently, a new preconditioning technique for the FETI method based upon a coarse grid problem associated with interface crosspoints has been introduced [MTF]. This gives optimal convergence property for high-order problems. In the present paper the problem of the parallel implementation of this new preconditioning technique is addressed and the performance of this approach is demonstrated for real life structural analysis problems.

For fourth-order problems, like plate or shell problems, the singularity with interface crosspoints, that means nodes that belongs to more than two subdomains, deteriorates the condition number of the dual Schur complement operator, the condensed interface operator defining the FETI method [FR94].

A new preconditioning technique leading to a two-level handling of interface continuity requirements has been recently developed [MTF]. The independence upon the number of subdomains and the polylogarithmical dependence upon the number of elements per subdomain of the condition number of the preconditioned interface problem has been proved.

In the present paper the problem of the parallel implementation of this new preconditioning technique is addressed. After recalling the principle of the FETI method, the new preconditioning technique is introduced and reinterpreted as a two-level FETI algorithm. A parallel implementation strategy is derived from this formulation. Last, the performance of this approach is demonstrated for real life structural analysis problems on an Intel-PARAGON system.

2 The FETI Method

The FETI method is based on introducing the Lagrange multiplier of the continuity condition on interfaces between subdomains. In the case of linear elasticity equations, the Lagrange multipler is equal to the field of interaction forces between subdomains.

Ninth International Conference on Domain Decomposition Methods
Editor Petter E. Bjørstad, Magne S. Espedal and David E. Keyes

In each subdomain, Ω_i, the local displacement field is solution of the linear elasticity equations with external loadings and boundary conditions inherited from the complete problem, and imposed forces (Neumann boundary conditions) on the interfaces with other subdomains.
With a finite element discretization, this leads to the following set of equations:

$$K_i u_i = B_i^t \lambda + b_i \tag{2.1}$$

where K_i is the stiffness matrix, u_i the displacement field, B_i a signed boolean matrix associated with the discrete trace operator, and λ the Lagrange multipler.
The continuity requirement along the interfaces is written as follows:

$$\sum_i B_i u_i = 0 \tag{2.2}$$

where the signed discrete trace matrices B_i are such that if subdomains Ω_i and Ω_j are connected by the interface Γ_{ij}, then restriction of equation (2.2) on Γ_{ij} is: $u_i - u_j = 0$.
In most subdomains, local problems (2.1) are ill posed, because only Neumann boundary conditions are imposed.
So, if K_i^+ is a pseudo-inverse of matrix K_i, and if columns of matrix N_i form a basis of the kernel of K_i (rigid body motions), equation (2.1) is equivalent to:

$$\left\{ \begin{array}{r} u_i = K_i^+[b_i + B_i^t \lambda] + N_i \alpha_i \\ N_i^t[b_i + B_i^t \lambda] = 0 \end{array} \right. \tag{2.3}$$

The first equation means that the solution of the problem is defined as the sum of a particular solution computed using the pseudo-inverse of K_i plus an element of the kernel. The second equation means that the right-hand side of equation (2.1) must be in the image space of K_i.
Introducing u_i given by equation (2.3) in the continuity condition (2.2) gives:

$$\sum_i B_i K_i^+ B_i^t \lambda + \sum_i B_i N_i \alpha_i = -\sum_i B_i K_i^+ b_i \tag{2.4}$$

With the constraint on λ set by the second equation of (2.3), the global interface problem can be written:

$$\begin{bmatrix} D & -G \\ -G^t & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \alpha \end{bmatrix} = \begin{bmatrix} d \\ c \end{bmatrix} \tag{2.5}$$

With:

- $D = \sum B_i K_i^+ B_i^t$, dual Schur complement matrix,
- $G\alpha = \sum B_i N_i \alpha_i$, jump of rigid body motions defined by α_i in Ω_i,
- $(G^t \lambda)_i = N_i^t B_i^t \lambda$, $d = -\sum B_i K_i^+ b_i$, $c_i = -N_i^t b_i$.

3 Parallel Solution of the Condensed Interface Problem

The number of constraints of the hybrid condensed interface problem (2.5) is the total number of rigid body modes. As this number is low, the projector associated with this constraint can be explicitly computed:

$$P = I - G(G^tG)^{-1}G^t \tag{3.6}$$

The computation of the product by projector P requires products by G and G^t and the solution of systems with form:

$$(G^tG)\alpha = G^t g \tag{3.7}$$

The product by G^t can be performed independently in each subdomain, the product by G requires exchanging data through interfaces between neighbouring subdomains. Both products can be easily performed in parallel in a message passing programming environment.
Parallelizing the solution of problem (3.7) is more challenging, because of its global implicit nature. To avoid the construction of matrix G^tG, this problem can be solved by the conjugate gradient algorithm. Then only products by G and G^t that can be performed in parallel are required.
Applying the projected conjugate gradient algorithm to the condensed interface problem (2.5), requires the following two main steps.

1. Given an approximate value λ^p, compute the particular solution of the local Neumann problem in each subdomain:

$$u_i^{p+} = K_i^+[b_i + B_i^t\lambda^p] \tag{3.8}$$

   and compute the jump of the local displacement fields along interfaces between subdomains that is the gradient of the condensed interface problem:

$$g^p = \sum_i B_i u_i^{p+} \tag{3.9}$$

2. Compute the projected gradient, Pg^p given by formula:

$$Pg^p = g^p + G\alpha^p \text{ with } (G^tG)\alpha^p = -G^t g^p \tag{3.10}$$

The projection step consists in fact in computing the rigid body motions coefficients α_i that minimize the jump of the complete displacement fields given by:

$$u_i^p = u_i^{p+} + N_i\alpha_i \tag{3.11}$$

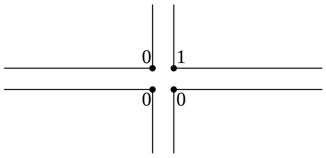
This minimization is performed in the sense that the jump of the complete displacement fields u_i^p, which is in fact the projected gradient, is orthogonal to the traces of all the local rigid body modes:

$$Pg^p = \sum_i B_i u_i^{p+} + \sum_i B_i N_i \alpha_i = \sum_i B_i u_i^p \tag{3.12}$$

$$(G\beta)^t Pg^p = 0 \ \forall \beta \ \Leftrightarrow \ (B_iN_i)^t Pg^p = 0 \ \forall i \tag{3.13}$$

This projection phase consists in solving a global coarse problem associated with the rigid body coefficients. The condition number of the projected dual Schur complement can be proved to be independent upon the number of subdomains ([FMR94]).

Figure 1 A "corner mode" for a scalar problem



4 The Second-level FETI Preconditioner

With domain decomposition method using local Neumann problems, the jumps of local solution fields at interface crosspoints can be discontinuous. For second-order problems, this singularity entails a polylogarithmic dependence of the condition number upon the local mesh size h/H ([Le 94]). For higher order problems, like plate and shell problems in structural analysis, the singularity is polynomial. So it is of great importance to get rid of these singularities.
The solution consists in constraining the Lagrange multiplier to generate local displacement fields that are continuous at interface crosspoints. Enforcing this constraint induces another level of preconditioning for the FETI method, that restores the optimality property of the FETI method ([FM]).
To get a practical formulation of this constraint, it can be observed that requiring the continuity of displacement fields at interface crosspoints is equivalent to imposing their jump to be orthogonal to the jump of "corner modes" defined as displacement fields with unit value in one space direction at a node connected to a crosspoint as in Figure 1.
Note C_i the set of corner modes in subdomain Ω_i, then the Lagrange multiplier λ^p satisfies the continuity requirement of associated displacement fields at interface crosspoints if the projected gradient satisfies:

$$(B_i C_i)^t P g^p = 0 \ \forall i \ \Leftrightarrow \ (\sum_i B_i C_i \gamma_i)^t P g^p = 0 \ \forall \gamma \tag{4.14}$$

The analogy between constraints defined by equations (3.13) and (4.14) suggests that the preconditioner can be constructed as a correction based upon jumps of corner modes in the same way as the projected gradient is constructed as a correction of the gradient based upon jumps of rigid body modes in (3.12).

$$MPg^p = Pg^p + \sum_i B_i C_i \delta_i \tag{4.15}$$

In term of structural analysis, this means that correcting the interaction forces at interface crosspoints should be enough to make the local displacement fields continuous at these points. In fact, the direction vector must be constructed from the projection of the preconditioned vector MPg^p to satisfy the constraint of orthogonality to the traces of rigid body modes (3.13). This projected preconditioned projected gradient takes the following form:

$$w^p = PMPg^p = Pg^p + \sum_i B_i C_i \delta_i + \sum_i B_i N_i \alpha_i \tag{4.16}$$

The variation of displacement fields induced by the variation w^p of interaction forces must have a null jump at interface crosspoints. By definition of the dual Schur complement, the jump of displacement fields induced by interface forces w^p is PDw^p. According to (4.14) this condition can be written:

$$(\sum_i B_i C_i \gamma_i)^t PDw^p = 0 \ \forall \gamma \tag{4.17}$$

By definition of projector P given by (3.13), this jump satisfies also :

$$(\sum_i B_i N_i \beta_i)^t PDw^p = 0 \ \forall \beta \tag{4.18}$$

Introduce now the coarse subspace defined by both the rigid body modes and the corner modes of all subdomains. Given α_i and δ_i, coefficients of rigid body motions and corner displacements in each subdomain Ω_i, define local coarse grid coefficients ξ_i by merging α_i and δ_i vectors. Then, the global coarse correction of interaction forces is defined as:

$$C_G \xi = (\sum_i B_i N_i \alpha_i) + (\sum_i B_i C_i \delta_i) \tag{4.19}$$

Thus the direction vector w^p takes form: $w^p = Pg^p + C_G \xi$ (4.20)

From equations (4.18) and (4.17) the coarse correction must satisfy the variational equality:

$$(C_G \zeta)^t PDw^p = (C_G \zeta)^t PDPw^p = 0 \ \forall \zeta \ \Leftrightarrow \ (C_G \zeta)^t PDPC_G \xi = -(C_G \zeta)^t Pg^p \ \forall \zeta \tag{4.21}$$

From (4.16), w^p must satisfy the constraint:

$$Pw^p = w^p \ \Leftrightarrow \ PC_G \xi = C_G \xi \tag{4.22}$$

Equations (4.21) and (4.22) represent in fact a constrained variational problem for the coarse grid space defined by rigid body and corner modes, which is similar to problem (2.5). Its formulation as an hybrid algebraic system of equation can be written:

$$\begin{bmatrix} C_G^t D C_G & -C_G^t G \\ -G^t C_G & 0 \end{bmatrix} \begin{bmatrix} \xi \\ \beta \end{bmatrix} = \begin{bmatrix} -C_G^t P g^p \\ 0 \end{bmatrix} \tag{4.23}$$

This system is precisely a coarse FETI problem, posed in the subspace of Lagrange multipliers defined as the image space of C_G. With this coarse grid preconditioner, the solution algorithm appears clearly as a two-level FETI method: at each iteration of projected conjugate gradient at the fine level, an additional preconditioning problem of the same type has to be solved at the coarse grid level.

5 Parallel Implementation and Performance Results

To keep a local representation of each operator, and to exploit domain-based parallelism, both the fine and the coarse grid problems must be solved through the same projected gradient procedure. However, this approach may appear very costly for the coarse grid problem, firstly because this problem must be solved exactly at each projected conjugate gradient iteration for the fine grid problem, and secondly because each coarse grid iteration is as expensive as a fine grid iteration, as local problems are the same for both.

In order to limit the cost, the coarse grid dual Schur complement can be preassembled at each subdomain level. In practice, if ζ_j is a coarse vector that has non zero entries only in subdomain Ω_j, $DC_G\zeta_j$ is non zero only in neighboring subdomains Ω_i. Hence, to precompute the $C_G^t DC_G$ matrix in each subdomain, it is necessary to solve the Neumann problem in subdomain Ω_i for each coarse grid mode in neighbouring subdomain Ω_j with non zero trace on interface Γ_{ij}.

The solution time for solving fine grid problems iteratively can be also drastically reduced using the restarted (projected) conjugate gradient technique presented in [Rou95]. The principle is as follows: if a set of conjugate directions (w^k), $1 \leq k \leq p$, is given, then the element x^0_{start} of x^0+Span$\{w^1, w^2, \ldots, w^p\}$ that minimizes the residual can be easily computed:

$$x^0_{start} = x^0 - \sum_{k=1}^{p} \frac{(g^0, w^k)}{(Aw^k, w^k)} w^k \tag{5.24}$$

Applying the standard conjugate gradient algorithm from starting vector x^0_{start} does not ensure that the new direction vectors are conjugate to the vectors w^k. To enforce these additional conjugacy relations, the new direction vector d^q at iteration number q must be reconjugated to the vectors w^k through the following procedure:

$$d^q = g^q - \frac{(g^q, Ad^{q-1})}{(Ad^{q-1}, d^{q-1})} d^{q-1} - \sum_{k=1}^{p} \frac{(g^q, Aw^k)}{(Aw^k, w^k)} w^k \ , \tag{5.25}$$

where g^q is the gradient vector at iteration q, $g^q = Ax^q - b$.

When this procedure is applied for successive right-hand sides with accumulation of conjugate direction vectors, it finally consists in using the conjugate gradient algorithm as a direct solver with an explicit computation of the inverse problem. For the two-level FETI method, this technique is applied for both coarse grid problems. As the dimensions of these problems are small, the procedure is very efficient numerically, and it can be parallelized via a domain-based approach.

Table 1 gives the performance results for a real life application of the two-level FETI method on an Intel-PARAGON machine with increasing number of subdomains and processors. The model is a submarine shell structure featuring 60332 nodes and 362000 degrees of freedom. The first 4 columns of this table give the numbers of subdomains, rigid body modes, corner modes and iterations at fine level. Column 5 features the condition number of the preconditioned condensed interface problem. Columns 6 and 7 give the parallel wall clock times for the initialization phase, including local matrices factorization and coarse matrix forming, and for the iterative solution phase. In both

Table 1 Comparison of parallel performance of one- and two-level FETI

number of					timings	
proc.	rigid body modes	corner modes	iter.	CN	init. (coarse)	total iter. (coarse)
Two-level FETI, local Dirichlet preconditioner						
30	132	351	93	822	361 (108)	514 (74)
40	168	474	94	662	298 (86)	453 (92)
60	318	762	105	828	128 (57)	355 (146)
80	396	954	87	537	69 (34)	240 (128)
One-level FETI, local Dirichlet preconditioner						
30	132	0	289	367398	253	1374
40	168	0	312	3206569	212	1217
60	318	0	406	3505918	71	869
80	396	0	416	315799	35	597

columns the fraction of time spent at coarse grid level is given for the two-level case. According to theory, thanks to the coarse grid preconditioner associated with rigid body modes, the condition number should not depend upon the number of subdomains but only upon a the local mesh size h/H. For a fixed global problem, increasing the number of subdomains increases the local mesh size, and consequently should decrease the condition number. But the condition number depends also upon the aspect ratio of subdomains. In the case of the real life problems like the one presented here, different mesh splittings lead to different aspect ratios of subdomains. Hence, although the local mesh size increases with the number of subdomains, the condition number does not necessarily decrease in a regular way.

Nevertheless, these tests show firstly the great improvement of condition number due to the second level FETI preconditioner. Secondly, it leads to a significant decrease, more than a factor of 2, of the total solution time compared to the one-level method. Thirdly, the parallel implementation exhibits a good scalability, although the time spent for coarse grid iterations reaches 50% of the solution time. These results are very representative of the ones obtained for various industrial problems.

6 Conclusion

The two-level FETI method appears to be a very efficient method for solving real life shell or plate problems. With a parallel implementation using domain-based parallelism and the restarted conjugate gradient method for coarse grids problems, the performance of the method on distributed memory parallel machines with message passing programming environment is already quite satisfactory. Nevertheless, there is still room for improvement in the parallel solution of coarse grid problems.

REFERENCES

[FM] Farhat C. and Mandel J. The two-level FETI method for static and dynamic plate problems- part1: an optimal iterative solver for biharmonic systems. *Comput. Meths. Appl. Mech. Engrg. (submitted)* .

[FMR94] Farhat C., Mandel J., and Roux F.-X. (1994) Optimal convergence properties of the feti domain decomposition method. *Comput. Meths. Appl. Mech. Engrg.* 115: 367–388.

[FR94] Farhat C. and Roux F.-X. (1994) In Oden J. T. (ed) *Implicit parallel processing in structural mechanics*, volume 2 of *Computational Mechanics Advances*, pages 1–124. Nort-Holland.

[Le 94] Le Tallec P. (1994) In Oden J. T. (ed) *Domain decomposition methods in computational mechanics*, volume 1 of *Computational Mechanics Advances*, pages 121–220. Nort-Holland.

[MTF] Mandel J., Tezaur R., and Farhat C. An optimal Lagrange multiplier based domain decomposition method for plate bending problems. *SIAM J. Sc. Stat. Comput. (submitted)* .

[Rou95] Roux F.-X. (1995) Parallel implementation of a domain decomposition method for non-linear elasticity problems. In David E. Keyes Y. S. and Truhlar D. G. (eds) *Domain-Based Parallelism and Problem decomposition Methods in Computational Science and Engineering*, pages 161–176. SIAM, Philadelphia.