Scalable Substructuring by Lagrange Multipliers in Theory and Practice

Charbel Farhat and Jan Mandel

1 Introduction

The FETI (Finite Element Tearing and Interconnecting) method is a non-overlapping domain decomposition algorithm for the iterative solution of systems of equations arising from the finite element discretization of self-adjoint elliptic partial differential equations. It is based on using direct solvers in subdomains and enforcing continuity at subdomain interfaces by Lagrange multipliers. The dual problem for the Lagrange multipliers is solved by a preconditioned conjugate gradient (PCG) algorithm. The FETI method was developed in [Far91, FR91, FR92], and discussed in detail in the monograph [FR94]. Unlike other related domain decomposition methods using Lagrange multipliers as unknowns [GW88, Rou90], the FETI method uses the null spaces of the subdomain stiffness matrices (rigid body modes) to construct a small “coarse” problem that is solved in each PCG iteration. It was recognized in [FMR94] and proved mathematically in [MT96] that solving this coarse problem accomplishes a global exchange of information between the subdomains and results in a method which, for elasticity problems, has a condition number that grows only polylogarithmically with the number of elements per subdomain, and is bounded independently of the number of subdomains. For time-dependent problems, one has to solve a linear problem with positive definite subdomain matrices in each time step. The coarse space built from null spaces is lost, resulting in deteriorating convergence with growing number of subdomains. Quasi-optimal convergence properties were retained by introducing an artificial coarse space [FCM95]. For plate bending problems, the condition number was observed to grow fast with the number of elements per subdomain [FMR94]. This was resolved by adding to the coarse space Lagrange multipliers that enforce continuity at the corners [MTF]. A related idea has been employed in the Balancing Domain Decomposition (BDD) method for plates [LMV94], where approximate continuity of the iterates at crosspoints is enforced by adding new basis functions associated with corners to the original coarse space [Man93, MB96].
While the underlying ideas of FETI and BDD are in a way dual, FETI is not the BDD method applied to the dual problem. The distinguishing features of both FETI and the BDD method is that they are non-overlapping and work for standard plate and shell finite elements used in everyday engineering practice.

The formulation of the FETI method presented here is based on [MTF], where more details can be found. This formulation covers the original FETI for solids as well as extensions to time-dependent problems and plates and shells. The extension to shells and practical results draw partially on [FCMR95, FM95].

2 Abstract Formulation of FETI

Let $\Omega$ be a domain in $\mathbb{R}^2$ decomposed into $N_s$ non-overlapping subdomains $\Omega_1, \Omega_2, \ldots, \Omega_{N_s}$. We assume that there is a conforming finite element discretization defined on $\Omega$, such that each subdomain is a union of some of the elements. The discrete problem arising from this discretization can be formulated as the minimization of the energy subject to intersubdomain continuity conditions,

$$\mathcal{E}(u) = \frac{1}{2} u^T K u - f^T u \to \min \quad \text{subject to } Bu = 0. \quad (1)$$

Here,

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N_s} \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N_s} \end{bmatrix}, \quad K = \begin{bmatrix} K_1 & 0 & \cdots & 0 \\ 0 & K_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_{N_s} \end{bmatrix},$$

with $u_s$, $K_s$, and $f_s$ being the vector of degrees of freedom, the local stiffness matrix, and the load vector, respectively, associated with the subdomain $\Omega_s$, and $B = [B_1, B_2, \ldots, B_{N_s}]$ a given matrix such that $Bu = 0$ expresses the condition that the values of the degrees of freedom associated with two or more subdomains coincide.

The local stiffness matrices $K_s$ and hence $K$ are positive semidefinite. The algorithm will use a given full rank matrix

$$Z = \begin{bmatrix} Z_1 & 0 & \cdots & 0 \\ 0 & Z_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Z_{N_s} \end{bmatrix}, \quad \text{Range } Z = \text{Ker } K.$$  

We assume that the global structure is not floating, that is, the solution of (1) is unique, which is equivalent to $\text{Ker } K \cap \text{Ker } B = \{0\}$.

Introducing Lagrange multipliers $\lambda$ for the constraint $Bu = 0$, the problem (1) becomes

$$Ku + B^T \lambda = f \quad \text{subject to } Bu = 0$$

(2)

A solution $u$ of the first equation in (2) exists if and only if $f - B^T \lambda \in \text{Range } K$, and

$$u = K^\dagger (f - B^T \lambda) + Z\alpha \quad \text{if } f - B^T \lambda \perp \text{Ker } K,$$

(3)
where \( \alpha \) is to be determined. Substituting \( u \) from (3) into the second equation of (2) yields \( BK^1(f - B^T \lambda) + BZ \alpha = 0 \). It follows that \( \lambda \) satisfies the system of equations

\[
\begin{align*}
  P(F \lambda - d) &= 0, \\
  G^T \lambda &= e,
\end{align*}
\]

where \( G = BZ, F = BK^1B^T, d = BK^1f, P = I - G(G^T G)^{-1} G^T, e = Z^T f \). Note that \( P \) is the orthogonal projection onto \( \text{Ker} \ G^T \). It can be proved that \( (G^T G)^{-1} \) exists [FR94, MTF].

It is easy to see that any two solutions \( \lambda \) of (4), (5) can differ only by a vector from \( \text{Ker} \ B^T \), and that any solution \( \lambda \) of (4), (5) yields the same solution \( u \) of (1) by (3) with \( \alpha = -(G^T G)^{-1} G^T (d - F \lambda) \).

The physical interpretation is that the Lagrange multipliers \( \lambda \) are interface forces and moments. From (3) and the definition of \( F \), the residual \( P(F \lambda - d) = -Bu \) has the interpretation of jumps in the values of the degrees of freedom between subdomains. The condition \( f - B^T \lambda \perp \text{Ker} \ K \) means that the action of the loads and inter-subdomain forces and moments does not excite rigid body motions.

To obtain more flexibility in the algorithm design, we add to the system (4), (5) a redundant weighted residual condition, and require that all iterates satisfy along with (5) a weighted residual condition

\[
C^T P(F \lambda - d) = 0,
\]

where \( C \) is another given matrix. The conditions (5), (6) will be enforced throughout the iterations by projecting the increments. For applications to static problems with solid elements, the additional constraint (6) is not necessary, but a proper choice of \( C \) is essential for time-dependent problems as well as plate and shell problems.

Increments that preserve (5), (6) form the subspace

\[
V' = \{ \mu | G^T \mu = 0, C^T Pf \mu = 0 \}
\]

The operator \( PF \) is symmetric on \( \text{Ker} \ G^T \) in the sense that

\[
\langle PF \lambda, \lambda' \rangle = \langle \lambda, PF \lambda' \rangle, \quad \text{for all } \lambda, \lambda' \in \text{Ker} \ G^T,
\]

and positive definite on the factorspace \( \text{Ker} G^T / \text{Ker} B^T \), cf., [MTF].

To get an initial approximation \( \lambda_0 \) that satisfies (5), (6), we solve a system of equations for a given \( \lambda_0 \)

\[
\begin{align*}
  G^T F(\lambda_0 + G \alpha + C \beta) + G^T G \mu &= G^T d, \\
  C^T F(\lambda_0 + G \alpha + C \beta) + C^T G \mu &= C^T d \\
  G^T (\lambda_0 + G \alpha + C \beta) &= e
\end{align*}
\]

for unknowns \( \alpha, \beta, \mu, \) and set \( \lambda_0 = \lambda_0 + G \alpha + C \beta \). We will use an analogous process to update a tentative search direction so that it satisfies (6); given \( \lambda_0 \), one finds a projected search direction \( \lambda = \lambda_0 + G \alpha + C \beta \), with \( \alpha, \beta \) determined from

\[
\begin{align*}
  G^T F(\lambda + G \alpha + C \beta) + G^T G \mu &= 0, \\
  C^T F(\lambda + G \alpha + C \beta) + C^T G \mu &= 0 \\
  G^T (\lambda + G \alpha + C \beta) &= 0
\end{align*}
\]
Then \( \lambda = Q\tilde{\lambda} \), with \( Q \) given by

\[
Q = I - \begin{bmatrix} G & C & 0 \end{bmatrix} \begin{bmatrix} G^T F G & G^T F C & G^T G \end{bmatrix}^{-1} \begin{bmatrix} G^T F \\ G^T F C \\ G^T C \\ 0 \end{bmatrix},
\]

where the superscript \(^\dagger\) denotes a generalized inverse. It can be proved that \([MTF]\)

\[
Q^2 = Q, \quad \text{Range } Q^T + \text{Ker } B^T = \text{Range } PF + \text{Ker } B^T.
\]

Our formulation of the generalized FETI method is now the method of conjugate gradients in the space \( V' \) for the operator \( PF \), preconditioned by \( QDQ^T \), where \( D \) is symmetric positive semidefinite. It follows from (8) that the preconditioner \( QDQ^T \) can be replaced by \( QD \) without changing the method. Therefore, the following algorithm is obtained.

Algorithm 1 (Generalized FETI) Given an initial \( \tilde{\lambda}_0 \), compute the initial \( \lambda_0 \) using (7), and compute the initial residual by

\[
r_0 = P(F\lambda_0 - d).
\]

Repeat for \( k = 1, 2, \ldots \) until convergence:

\[
z_{k-1} = Dr_{k-1}, \\
y_{k-1} = Qz_{k-1}, \\
\xi_k = r_{k-1}^T y_{k-1}, \\
p_k = y_{k-1} + \xi_k p_{k-1}, \quad (p_1 = y_0) \\
\nu_k = \frac{\xi_k}{p_k^T PF p_k}, \\
\lambda_k = \lambda_{k-1} + \nu_k p_k, \\
r_k = r_{k-1} - \nu_k PF p_k.
\]

3 Selection of Common Algorithm Components

Continuity Constraint \( Bu = 0 \)

For a node \( x_i \) at the intersection of two subdomains \( \partial \Omega_r \cap \partial \Omega_s \), we define the continuity constraint on the displacement degrees of freedom by

\[
(Bw)_{rs}(x_i) = \sigma_{rs}(w_r(x_i) - w_s(x_i)) = 0.
\]

We use a similar condition for derivative or rotation degrees of freedom, if present. Here, \( \sigma_{rs} = 1 \) or \( \sigma_{rs} = -1 \) is a constant assigned to the edge (in 2D) or side (in 3D). In particular, the entries of \( B \) are \(-1, 0, +1\), and they are constant along an edge or side between subdomains. Note that this construction of \( B \) results in redundant constraints at all degrees of freedom that belong to more than two subdomain. This slightly increases the number of the Lagrange multipliers and complicates the analysis, but makes a simpler parallel implementation possible.
Dirichlet Preconditioner

Decompose the space of all the degrees of freedom into the space of the degrees of freedom lying on the subdomain interfaces, and the degrees of freedom internal to the subdomains

\[ W = W_b \times W_i, \]

where the subscript \( b \) denotes the block of degrees of freedom on subdomain boundaries, and the subscript \( i \) denotes degrees of freedom internal to the subdomains. Then,

\[ B = [B_b, 0], \]

since \( B \) has nonzero entries for the subdomain interface degrees of freedom only. Also,

\[ Z = \begin{bmatrix} Z_b \\ Z_i \end{bmatrix}, \quad G = BZ = B_b Z_b, \quad \text{Ker} B^T = \text{Ker} B_b^T. \]

Let \( S \) be the Schur complement of \( K \) obtained by elimination of the degrees of freedom internal to all subdomains:

\[ S = K_{ib} - K_{bi} K_{ii}^{-1} K_{ib}. \tag{9} \]

It is easy to see that

\[ F = BK^T + B^T = B_b S^T B_b^T, \tag{10} \]

and that \( \text{Ker} S = \text{Range} Z_b \). It is well known that the evaluation of the matrix-vector product \( S^T u \) reduces to the solution of independent Neumann problems on all subdomains. Analogously to (10), we choose \( D = B_b S B_b^T \), giving the preconditioner

\[ QD = Q B_b S B_b^T. \tag{11} \]

This preconditioner is called the Dirichlet preconditioner, since evaluating the matrix-vector product \( Sr \) is equivalent to solving independent Dirichlet problems on all subdomains.

Lumped Preconditioner

This is a simplified version of the Dirichlet preconditioner (11), which trades mathematical quasi-optimality for a lower cost per PCG iteration. The Schur complement \( S \) of \( K \) obtained from (9) is replaced simply by its leading term \( K_{ib} \). This is equivalent to “lumping” each subdomain stiffness on its interface boundary. The resulting preconditioner is given by

\[ QD = Q B_b K_{ib} B_b^T. \tag{12} \]

4 Special Instances of FETI

FETI for Solid Mechanics (Second-Order Elasticity)

The original FETI algorithm [Far91, FR91, FR92] is obtained by omitting the condition (6). Then, \( Q \) becomes the identity, and an initial approximation \( \lambda_0 \) is only
required to satisfy $G^T\lambda_0 = e$. It was proved in [MT96] that for the Laplace equation, P1 conforming elements, and the Dirichlet preconditioner both in 2D and 3D, and under the usual technical assumptions about the shape regularity of the elements and the subdomains, one has the following upper bound on the condition number

$$\kappa = \frac{\lambda_{\text{max}}(QDPF)}{\lambda_{\text{min}}(QDPF)} \leq C \left( 1 + \log \frac{H}{h} \right)^\gamma$$  \hspace{1cm} (13)

where $h$ is the characteristic element size, $H$ the characteristic subdomain size, and $\gamma = 3$. If there are no nodes shared between more than two subdomains, and in some other special cases, (13) holds with $\gamma = 2$.

The bound (13) no longer holds for the lumped preconditioner, but one observes a superconvergence effect instead [FMR94]. Because the operator $PF$ is a discretization of the inverse of a differential operator, which is compact, the eigenvalues are clustered around zero. Since the convergence of conjugate gradients after $k$ steps is determined by the spectrum left after removing $k$ extremal eigenvalues, this distribution of eigenvalues results in fast convergence. Unfortunately, as the number of subdomains increases, the spectrum fills in and the superconvergence effect is observed to disappear.

**FETI for Time-dependent Problems**

The solution of time-dependent problems by an implicit method calls for the repeated solution of linear systems with the subdomain matrices $K_s$ of the form

$$K_s = \tilde{K}_s + (\Delta t)^{-1} M_s,$$  \hspace{1cm} (14)

where $\tilde{K}_s$ now denotes the subdomain stiffness matrix, $M_s$ is the subdomain mass matrix and $\Delta t$ is the time step. Because the mass matrix is positive definite, $\text{Ker} \ K = \{0\}$, $Z$ is void. Therefore, the natural coarse problem for the unknowns $\alpha$ is lost and the number of iterations increases with the number of subdomains. This can be corrected by the selection $C = B\tilde{Z}$, where $\tilde{Z}$ is chosen so that $\tilde{Z} = \text{diag} \tilde{Z}_s$, $\text{Range} \ \tilde{Z}_s = \text{Ker} \ \tilde{K}_s$. Then, it was again observed that the number of iterations is independent on the number of subdomains. It was proved that the iterates approach the static case in the following sense. Consider the FETI iterative process on a linear system with the matrices $K_s$ from (14) with $0 < \Delta t \leq +\infty$, and a fixed right hand side. Let $\lambda^k(\Delta t)$ denote the approximate solution after $k$ iterations of FETI for a given $\Delta t$. Then, for all $k$,

$$\lim_{\Delta t \to +\infty} \lambda^k(\Delta t) = \lambda^k(+\infty).$$

For further details, see [FCM95].

**FETI for Plates**

Here, the columns of $C$ are chosen as vectors with a one at the position of the Lagrange multiplier that enforces the continuity of the transversal displacement at a crosspoint, and zeroes elsewhere. A crosspoint is an interface node adjacent to at least three subdomains or to two subdomains and the complement of $\Omega$. That is,
Figure 1  The domain splitting for a general operator $A$ ($n = 3$ & $N = 4$)

Lagrange multipliers that correspond to crosspoints are enforced exactly throughout the iterations.

The condition number bound (13) was proved in [MTF95, MTF] for a general class of plate bending elements that have the property that the local stiffness matrix of the element is spectrally equivalent to that of the HCT element for the biharmonic equation [LMV94]:

$$c_1 K_T^{HCT} \leq K_T \leq c_2 K_T^{HCT}$$

where $K_T^{HCT}$ is the reduced HCT element stiffness matrix of the biharmonic equation [CT66], with the rotations interpreted as derivatives of the transversal displacement, and $K_T$ is the element stiffness matrix for a triangular or rectangular element with one displacement and two rotation degrees of freedom per node. The spectral equivalence (15) was proved in [LMV] for the particular case of the DKT element [BBH80], and for a general class of non-locking $P1$ Reissner-Mindlin elements that have the element energy functional equivalent to

$$\int_T |\nabla \theta|^2 dx + \frac{1}{\ell^2 + h^2} \int_T |\theta - \nabla u|^2 dx$$

with $u \in P_1(T), \theta \in (P_1(T))^2, h = \text{diam}(T)$, $u$ the transversal displacement, and $\theta$ the rotation. This includes the DKT plate bending element as restated in [Pit87].

**FETI for Shells**

The ideas and theory governing the FETI method for plates [FCMR95, FM95] suggest that, for shell problems, the continuity of the component of the displacement field that is normal to the shell surface should be enforced at the substructure crosspoints throughout the PCG iterations. One approach for implementing this requirement and bypassing the difficulties associated with defining normals for non-smooth shell surfaces consists in enforcing the continuity of the displacement field at the substructure crosspoints in the direction of all three coordinate axes. Clearly,
Figure 2  A 30-substructure mesh partition

this would automatically enforce the continuity of the normal component of the
displacement field at the crosspoints, while requiring only a minor modification of the
implementation of the FETI method for plates. More precisely, only the construction
of the $C$ matrix needs to be modified to have a one at the position of each of the three
Lagrange multipliers that enforce the continuity of each of the three displacement
degrees of freedom at a crosspoint. In [FCMR95], the authors have shown numerically
that, even for irregular shell problems with juncures, such an extension of the FETI
method preserves the quasi-optimal convergence properties proved mathematically
in [MTF96, MTF] for plate problems.

However, the extension of the FETI method to shell problems summarized above
generates a coarse crosspoint problem that is three times larger than that for plate
problems, because the continuity of all three displacement degrees of freedom rather
than the transversal displacement is enforced at the substructure crosspoints. Hence,
wherever the shell structure has a smooth surface, one can enforce only the continuity
of the normal component of the displacement field at a crosspoint. This is done by
setting $C_{ij} = n_x$, $C_{i+1,j} = n_y$, $C_{i+2,j} = n_z$ at that crosspoint and $C_{ij} = 0$ elsewhere,
and incurs the same computational cost as for plate problems. Here, $n_x$, $n_y$, and $n_z$
denote the three components of the normal to a shell surface at a given crosspoint.

5 Parallel Implementation and Computational Results

The parallel implementation of the FETI method is straightforward, except for the
solution of the coarse problem, which has been discussed in detail in [FC94, Far96].
Because of space limitation, we focus here on illustrating only the scalability properties
of this method with respect to the number of substructures and processors. The
additional scalability of the FETI method with respect to the mesh size has already
been demonstrated and reported in all the FETI references cited in this paper.

For this purpose, we consider the stress analysis on a Paragon XP/S system of
a submarine structure loaded by a standing pressure wave (Fig. 1). The finite element
model contains 60332 nodes, 120064 three-noded shell elements, a total of 361735
active degrees of freedom, and many structural juncures. The mesh is partitioned
into 30, 40, 60, and 80 substructures with good aspect ratios [FMB95] for parallel
computations on a Paragon XP/S system (2).

Four structural analyses were performed using the FETI method for shells. The corresponding performance results are summarized in Table 1.

Clearly, scalability is well demonstrated for the solution of the coarse problems as well as the solution of the overall problem. The size of the coarse problem increases with the number of substructures and processors, but the CPU time elapsed in forming and solving iteratively the repeated coarse problems is shown to remain almost constant. Moreover, the convergence rate is observed to be almost independent of the number of substructures, and the measured total solution time decreases superlinearly with the number of processors.

**Acknowledgement**

This research was partially supported by ONR grant N-00014-95-1-0663, and NSF grants ASC-9217394 and ASC-9121431.

**REFERENCES**


Symposium on Domain Decomposition Methods, Penn State, November 1993.


