Contemporary Mathematics
Volume 180, 1994

Domain Decomposition for Adaptive *hp* Finite Element Methods

J. TINSLEY ODEN, ABANI PATRA and YUSHENG FENG

ABSTRACT. A highly parallelizable domain decomposition solution technique for adaptive *hp* finite element methods is developed. The technique uses good partitioning strategies and a subspace decomposition based preconditioned iterative solver. Two level orthogonalization is used to obtain a reduced system which is preconditioned by a coarse grid operator. Numerical results show fast convergence for the iterative solver and good control of the condition number (less than 16 for meshes with spectral orders up to 8).

1. Introduction

Adaptive *hp* finite element methods [1], in which both the mesh size and spectral order are independently varied over the whole mesh,. produce exponential convergence rates in the discretization error even in the presence of singularities. Together with domain decomposition and parallel algorithms they have the potential to produce dramatic improvements in finite element modeling of problems in computational mechanics. However, the complex mesh and data structures involved in adaptive *hp* finite element methods raise interesting issues in domain partitioning and the parallel solution process. Domain decomposition for *h*–version and *p*–version finite element methods have been investigated by several authors [5][6], but there have been no studies on domain decomposition for the *hp*–version finite element methods.

Two issues that immediately arise are – automatic partitioning of *hp* adaptive finite element meshes and efficient parallel solution of the resulting algebraic systems using iterative solvers. In partitioning an *hp* mesh, difficulties may be encountered due to non-uniform computational load across the elements, non-uniform communication patterns and constraints between partitions. The usual choice of finite

1991 *Mathematics Subject Classification.* Primary 41A10; Secondary 65N30. The authors are supported by DARPA contract no. DABT63-92-C-0042 This paper is the final form. No revision will be submitted elsewhere.

element basis functions will also lead to high condition number of the associated algebraic system (possibly $O(10^{p_{max}})$, where p_{max} is the highest spectral order of the approximation) and slow convergence of almost all iterative solvers.

In this study, a highly parallelizable domain decomposition method is developed for adaptive *hp* finite element methods. The remainder of the paper is organized as follows. In section 2, three families of partitioning algorithms are presented. In section 3, a decomposition of the *hp* finite element space is formulated. The algorithm is tested on a two–dimensional elliptic boundary value problem. Numerical results and conclusions are given in section 4 and section 5, respectively.

2. Automatic Partitioning Algorithms

Efficient parallel computing requires decomposition of the problem into "load balanced" sub-problems with minimal communication and synchronization among them. The twin objectives in a partitioning algorithm thus are equi-distribution of computational effort among the sub–domains, which are eventually assigned to different processors and minimization of the interfaces between the sub-domains. These two objectives are necessary to maximize utilization of the processors, minimize communication among them and reduce the size of the interface problem. However, one may often need to accept trade-off between the two goals.

Unlike *h*-version methods, choice of *a-priori* computational effort measure is a non-trivial issue. In adaptive *hp* finite element approximation, good candidate measures of computational effort appear to be 1) error distribution in a coarse mesh solution 2) degrees of freedom distribution in the mesh and 3) element conditioning estimation.

Although degrees of freedom seems to be a natural choice, it does not reflect the computational effort very accurately. The motivation for error as a partitioning measure is in the *hp* adaptive strategy developed in a previous study [2].

In this section, three partitioning algorithms for adaptive *hp* meshes are presented. Each algorithm is implemented with different choices of computational effort measure. Due to the page limit of these proceedings,only one of the algorithms is discussed in detail. More details will be provided in future publications.

2.1. Mesh Traversal Based Decomposition (MTBD). In this family of partitioning algorithms, the mesh is traversed in some fashion and then elements are accumulated into partitions based on some estimate of computational effort. In the first choice of ordering implemented, the mesh is traversed in a nearest neighbor with lowest load fashion. This ensures some amount of locality in the decomposition as each element has at least one neighboring element in the decomposition. However this ordering often results in disconnected partitions. This drawback can be somewhat overcome by using an ordering created by mapping the centroids of the elements onto a Peano–Hilbert curve[4].

2.2. Recursive Load Based Bisection of Coordinate (RLBBC) This family of algorithms uses an explicit choice of interface to create the partition. The advantage of these methods is that both objectives, load balance and minimum interface are explicitly addressed. However, the cost of doing so inhibits the method.

In principle, the methods are comprised of selecting candidate separator surfaces and then selecting a separator based on maximum load balance and minimum interface. The selection involves assigning to each candidate separator surface a number indicative of load balance and interface size associated with the resulting domain partitions. This approach is a generalization of the work of Miller, Vavasis and their coworkers [3] on partitioning two and three dimensional finite element grids for the h version.

2.3. Recursive Load Based Bisection of Ordering (RLBBO). In this family of algorithms, an attempt is made to combine the advantages of the mesh traversal algorithms with that of the interface partitioning. The elements are ordered using the "Peano-Hilbert curve" ordering, and then a recursive splitting is applied to the resulting one–dimensional ordering of the elements. The basic algorithm is outlined below:

Algorithm

1. Create an ordering of the elements by mapping the centroids of the elements onto a Peano-Hilbert curve.
2. Let t_K be the distance of the centroid of element K along this curve.
3. Compute maximum and minimum of t_K.
4. Compute n trial separator levels as

$$t_i = t_{min} + \frac{t_{max} - t_{min}}{n}$$

5. For each t_i compute q_i

$$q_i = abs(\frac{dof_{left}}{dof_{right}} - 1) \cdot dof_{tot} + dof_{inter}$$

   Replace dof by error or other load estimate as appropriate

6. Choose as interface t_i corresponding to lowest q_i
7. Apply 1-6 recursively.

One particularly demanding *hp* mesh (p ranges from 1 to 7) and corresponding partitions are shown in the figure 1(a) and figure 1(b). The resulting partitions seem to have balanced load and nice interfaces.

3. Domain Decomposition Solver for *hp* FEM

The solver will be discussed with respect to the model problem defined below:

$$\text{Find } u \in \mathcal{V} \quad \text{such that} \quad \mathcal{B}(u,v) = \mathcal{L}(v) \quad \forall\ v \in \mathcal{V}$$

where $\mathcal{V} = \{v \in H^1(\Omega): v = 0 \text{ on } \partial\Omega\}$, and $\mathcal{B}(u,v)$ is the bilinear form resulting from the weak formulation of a two-dimensional second order elliptic PDE with Dirichlet boundary conditions on the boundary $\partial\Omega$.

Define $\mathcal{V}^h \subset \mathcal{V}$ as a finite dimensional subspace constructed by a series of affine mappings of the following functions defined on a 'master element'.

Vertices (Nodes):

$$\widehat{\psi}_i(\xi,\eta) = \frac{1}{4}(1 \pm \xi)(1 \pm \eta) \qquad i = 1,2,3,4$$

Edges:

$$\widehat{\zeta}_i^k(\xi,\eta) = \begin{cases} \frac{(-1)^i}{2}(1 \pm \eta)\rho_k(\xi) & i = 1,3 \\ \frac{(-1)^i}{2}(1 \pm \xi)\rho_k(\eta) & i = 2,4 \end{cases}$$

Interior (Bubble):

$$\widehat{b}_{ij}(\xi,\eta) = \rho_i(\xi)\rho_j(\eta) \qquad 2 \le i,\, j \le p$$

$$\rho_k(\xi) = \sqrt{\frac{2k-1}{2}} \int_{-1}^{\xi} P_{k-1}(s)ds$$

with P_{k-1} the Legendre polynomial of degree $k-1$

Assume domain Ω is decomposed into N_D sub-domains. Each sub-domain Ω_i is associated with a subspace $\mathcal{V}_i^h$. Then

$$\mathcal{V}^h = \sum_{i=1}^{N_D} \mathcal{V}_i^h, \qquad \mathcal{V}_i^h \subset \mathcal{V}^h \subset \mathcal{V} \quad \forall\, i$$

Each subspace is further decomposed into

$$\mathcal{V}_i^h = \mathcal{X}_i^N + \mathcal{X}_i^S + \mathcal{X}_i^V + \mathcal{X}_i^E + \mathcal{X}_i^B$$

where $\mathcal{X}_i^N$ and $\mathcal{X}_i^S$ are spaces spanned by vertex and edge functions on subdomain interfaces. $\mathcal{X}_i^V$, $\mathcal{X}_i^E$ and $\mathcal{X}_i^B$ are spaces spanned by vertex, edge and bubble functions in subdomain interiors. Then, the bilinear form can be written as:

$$\mathcal{B}(u_{hp},u_{hp}) = \sum_{i=1}^{N_D} \mathcal{B}_i(u_{hp}^N + u_{hp}^S + u_{hp}^V + u_{hp}^E + u_{hp}^B, u_{hp}^N + u_{hp}^S + u_{hp}^V + u_{hp}^E + u_{hp}^B)$$

Now if the local trial functions are chosen to satisfy the orthogonality condition

$$(1) \qquad \mathcal{B}_{i,K}(\gamma_j, b_k) = 0 \quad \forall \gamma_j \in \mathcal{X}_i^V + \mathcal{X}_i^E, b_k \in \mathcal{X}_i^B$$

the element stiffness matrix reduces to the form

$$K_{elt} = \begin{bmatrix} \widetilde{VV} & \widetilde{VE} & 0 \\ \widetilde{EV} & \widetilde{EE} & 0 \\ 0 & 0 & BB \end{bmatrix}$$

where $\widetilde{VV}, \widetilde{EE}, \widetilde{VE}$ represent modified blocks of the original element stiffness matrix.

If in addition the trial functions satisfy the orthogonality condition

$$\mathcal{B}_{i,K}(\tau_j, \varphi_k) = 0 \quad \forall \tau_j \in \mathcal{X}_i^N + \mathcal{X}_i^S, \varphi_k \in \mathcal{X}_i^V + \mathcal{X}_i^E + \mathcal{X}_i^B \tag{2}$$

the resulting subdomain stiffness matrices reduce to

$$K_i = \begin{bmatrix} \widetilde{NN} & \widetilde{NS} & 0 \\ \widetilde{SN} & \widetilde{SS} & 0 \\ 0 & 0 & II \end{bmatrix} \qquad II = \begin{bmatrix} \widetilde{II_{VV}} & \widetilde{II_{VE}} & 0 \\ \widetilde{II_{EV}} & \widetilde{II_{EE}} & 0 \\ 0 & 0 & BB \end{bmatrix}$$

where NN, NS, SN, SS are shared degrees of freedom among subdomains. Note that the first condition causes the orthogonalization of the bubbles with respect to the edges and vertices while the second causes the orthogonalization of the interfaces with the interiors. The subdomain problems are now independent of the interface problem.

Remark 1: Implementation of the first condition can be done at the element level and is thus completely parallelizable.

Remark 2: The modifications of NN etc. to $\widetilde{NN}$ etc. are of the form

$$\widetilde{NN} = \sum_{i=1}^{N_D} (NN_i + \widehat{NN}_i)$$

Remark 3: If an iterative solver (e.g. PCG,GMRES) is used, these modifications can then directly participate in the parallel matrix-vector product

$$K \cdot p = \sum_{i=1}^{N_D} (K_i + \widehat{K}_i) \cdot p$$

and there is no need for assembly of these components.

The parallel domain decomposition algorithm is summarized as follows:

Parallel Domain Decomposition Solution Algorithm

1. Partition the mesh into subdomains using any of the decomposition algorithms.
2. Create subdomain approximations transforming the algebraic system at the element level to satisfy orthogonality conditions (1) and (2).
3. Solve the reduced preconditioned system by an iterative method (e.g. PCG,GMRES) using coarser grid operator preconditioning.
4. Solve subdomain problems in parallel.

5. Transform the solution of the reduced system to the original system for condition (2) at subdomain level and condition (1) at element level.

4. Numerical Results And Conclusions

Poisson's equation in two dimensions is chosen as a test problem. In figure 2(a), both iteration counts and condition number estimation are plotted against p (ranging from 2 to 8). The condition number is controled under 16. Figure 2(b) shows the residual and condition number estimation against iteration count for *hp* DD and Conventional Jacobi. A variant of Lanczos connection is used to estimate conditioning of the preconditioned operator[7].

In this study, several partitioning algorithms were developed and tested for adaptive *hp* meshes. Performance of RLBBO appears to superior. Two level orthogonalization of finite element basis functions produces good control of conditioning for the algebraic system generated by *hp* finite element approximation. The resulting domain decomposition solution algorithm is highly parallelizable.

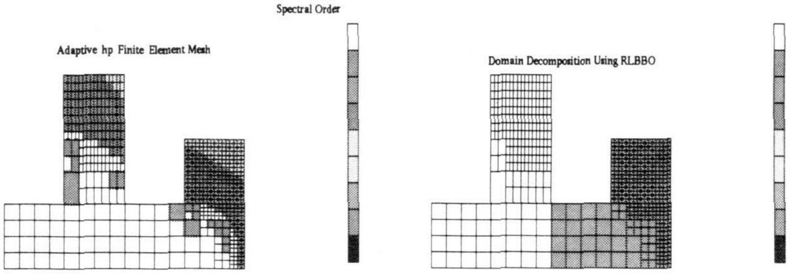
Spectral Order
Adaptive hp Finite Element Mesh
Domain Decomposition Using RLBBO

Figure 1 (a) Adaptive *hp* mesh (b) Partitioning generated by RLBBO

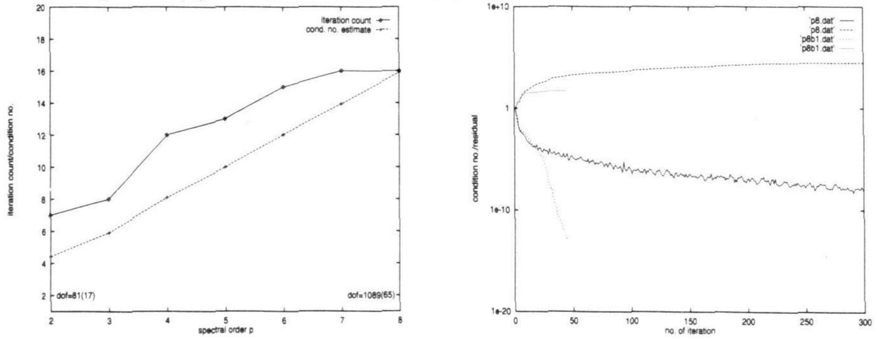

Figure 2.(a) *hp* Domain Decomposition Solver (4 sub-domain) (b) Comparison of *hp* DD solver with conventional Jacobi Iterative Solver

References

1. L. Demkowicz, J. T. Oden, W. Rachowicz and O. Hardy "Toward A Universal *hp* Adaptive Finite Element Strategy, Part 1. Constrained Approximation and Data Structure" *Comput. Methods. Appl. Mech. and Engg.*, 77(1989), pp.79-112

2. J. T. Oden, Abani Patra and Y. S. Feng, " An *hp* Adaptive Strategy", *Adaptive, Multilevel and Hierarchical Computational Strategies*, A. K. Noor(ed)., AMD-Vol. 157, 1992, pp. 23-46.

3. G. L. Miller, S.Teng, W. Thurston, S. A. Vavasis, "Automatic Mesh Partitioning", CTC92TR112 , Advanced Computing Research Institute, Department of Computer Science, Cornell University, Ithaca, 1992.

4. E. A. Patrick, D. R. Anderson and F. K. Bechtel, "Mapping Multidimensional Space To One Dimension For Computer Output Display", *IEEE Transactions on Computers*, vol C-17, No. 10, October, 1968

5.J. Bramble, J. Pasciak, A. Schatz, "The Construction of Preconditioners for Elliptic Problems by Substructuring I", *Math. Comp.* 47, N 175 (1986), pp. 103-134

6. I. Babuska, A. Craig, J. Mandel, and J. Pitkaranta, "Efficient Preconditioning For The *p* version finite element method in Two Dimensions", *SIAM J. Numer. Anal*, Vol 28, No. 3, pp. 624-661, June 1991.

7. J. Tinsley Oden and Yusheng Feng, "A CG Extension for Nonsymmetric Linear Systems and Lanczos' Connection to PCG", Cornelius Lanczos International Centenary Conference, Raleigh, North Carolina , December 1993.

Texas Institute For Computational and Applied Mathematics,
The University Of Texas at Austin,
3500, West Balcones Ctr. Drive,
Austin, TX-78759
U.S.A

oden@ticom.ae.utexas.edu
abani@ticom.ae.utexas.edu
feng@ticom.ae.utexas.edu