Journal of Computational Physics **160,** 298–317 (2000)
doi:10.1006/jcph.2000.6463, available online at http://www.idealibrary.com on IDEAL®

Overlapping Schwarz Methods for Unstructured Spectral Elements

Luca F. Pavarino*[,1] and Timothy Warburton†[,2]

**Department of Mathematics, Università di Milano, Via Saldini 50, 20133 Milan, Italy; and †Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom*
E-mail: pavarino@ares.mat.unimi.it, timw@comlab.ox.ac.uk

Received July 29, 1999; revised January 24, 2000

A parallel and scalable domain decomposition method for unstructured and hybrid spectral element discretizations of elliptic problems is introduced and studied. The spectral elements are affine images of the reference triangle or square in two dimensions and of the reference tetrahedron, pyramid, prism, or cube in three dimensions. The method is based on overlapping Schwarz techniques applied to the Schur complement of the discrete system and is implemented as a preconditioner for a Krylov space method. Numerical results in two and three dimensions show that the iteration counts of our method are bounded by a constant independent of the spectral degree and the number of subdomains. The resulting elliptic solver can be used in Navier–Stokes simulations using the spectral element code NekTar. © 2000 Academic Press

Key Words: spectral elements; unstructured meshes; domain decomposition; overlapping Schwarz methods.

1. INTRODUCTION

In this paper, we introduce and study a parallel and scalable domain decomposition method for unstructured and hybrid spectral element discretizations of elliptic problems. The spectral elements are affine images of the reference triangle or square in two dimensions and of the reference tetrahedron, pyramid, prism, or cube in three dimensions. The method is based on overlapping Schwarz techniques, consisting in dividing the domain of the given elliptic problem into overlapping subdomains and solving smaller instances of the elliptic problem on these subdomains. An additional coarse problem with few degrees of freedom per subdomain is also solved in order to obtain scalability. The method is implemented

[1] This work was supported by Murst and by the National Science Foundation under Grant NSF CTS-9417520.

[2] This work was supported by DARPA Grant 98-240 and by the Naval Undersea Warfare Center Grant N66604-98-C-1266.

as a preconditioner for a Krylov space method such as the conjugate gradient method for symmetric problems and GMRES or QMR for nonsymmetric problems. This elliptic solver can be used in Navier–Stokes simulations using the spectral element code NekTar described in Section 4.

Structured spectral elements employing hexahedral elements, tensor product basis functions, and Gauss–Lobatto–Legendre quadrature rules have been developed extensively both theoretically and numerically; see, e.g., Bernardi and Maday [2] and the references therein. Domain decomposition for hexahedral spectral and *hp* elements has been developed mainly using nonoverlapping techniques, also known as iterative substructuring methods; see, e.g., Mandel [21, 20], Fischer and Rønquist [12], Rønquist [28], Pavarino and Widlund [27], Pavarino [25], Guo and Cao [13], and Oden *et al.* [22]. A few works have proposed and studied overlapping methods; see Pavarino [24], Casarin [4], Fischer [11], and Rønquist [29]. Multi-p methods, analogous to multigrid methods for h-version finite elements, can be found in Katz and Hu [19].

Unstructured spectral and *hp* elements have been studied by Szabo and Babuška [35], and Babuška and Suri [1] and more recently by Karniadakis' group; see Karniadakis and Sherwin [17] and the references therein. The choice of basis functions and quadrature rules is a more difficult issue for unstructured spectral elements and the theoretical analysis of these methods still presents some basic open questions. Different choices of interpolation points on triangles and tetrahedra can be found in Chen and Babuška [7, 8], Hesthaven [16], Wingate and Taylor [40, 39], and Heinrichs [14]. We follow in this paper the approach of [17] based on Dubiner's basis function [10], described in the next section.

Nonoverlapping domain decomposition methods for triangular and tetrahedral spectral elements can be found in Bica's Ph.D. Thesis [3], Sherwin *et al.* [30], and Casarin and Sherwin [5]. As it is already well known for standard h-version finite elements, the lack of overlap among subdomains requires the construction of complex coarse solvers in order to ensure scalability. The method we propose in this paper, on the other hand, is based on overlapping Schwarz techniques which allow greater freedom in the choice of local and coarse solvers. In order to reduce the computational cost of our preconditioner, we implicitly eliminate the interior degrees of freedom in each element (as in nonoverlapping methods) and we apply the overlapping Schwarz technique to the resulting Schur complement involving only the interface degrees of freedom. The same technique could, of course, be applied to the whole discrete system involving both interior and interface degrees of freedom. Due to the unstructured spectral elements used, we employ generous overlap consisting of a layer at least one-element wide around each subdomain. Numerical results in two and three dimensions show that the iteration counts of our method are bounded by a constant independent of the spectral degree, the number of elements, and the number of subdomains. At this moment, we are able to prove this result only in the case of a structured quadrilateral or hexahedral mesh, using the theory developed in [24].

The paper is organized as follows. In Section 2, we briefly describe the unstructured spectral element discretization used in this paper. In particular, we illustrate in separate sections the coordinate system, the quadrature rules, the basis functions, the polynomial spaces, and the continuous and discrete elliptic problems. In Section 3, we introduce our overlapping Schwarz preconditioner in its additive form and we present in Section 4 the results of several numerical experiments performed with the code NekTar showing the scalability of the proposed method.

2. UNSTRUCTURED SPECTRAL ELEMENTS

In this section, we briefly recall the basic facts about the spectral element discretization considered, including the standard element mappings, quadrature rules, polynomial bases, and spaces. A more complete treatment can be found in the book by Karniadakis and Sherwin [17].

2.1. *Coordinate Systems*

For triangular and quadrilateral elements the reference element is a square $(a, b) \in [-1, 1]^2$. For tetrahedral, pyramidal, prismatic, and hexahedral elements the reference element is a cube $(a, b, c) \in [-1, 1]^3$. The reference element is mapped to a standard element specific to each type of element, which is described by a set of coordinates (r, s) in two dimensions and (r, s, t) in three dimensions. Subsequently each standard element is continuously mapped to its physical element.

For example, the triangle standard element is the set $\{(r, s) \mid -1 < r, s; r + s < 0\}$. The reference element is mapped to the standard triangle element by the mapping:

$$\begin{aligned} r &= \frac{1}{2}(1 + a)(1 - b) - 1, \\ s &= b. \end{aligned}$$

The standard triangle element is mapped to the physical triangle element by the mapping

$$\mathbf{x} = -\frac{(r + s)}{2}\mathbf{v^1} + \frac{(1 + r)}{2}\mathbf{v^2} + \frac{(1 + s)}{2}\mathbf{v^3},$$

where $\mathbf{v^0}, \mathbf{v^0}, \mathbf{v^0}$ are the vector coordinates of the vertices of the physical triangle.

The standard finite element mappings are used for the quadrilateral and hexahedral elements. The mappings for the tetrahedron, pyramid, and prism can be found in Sherwin [31] and Warburton [36]. Details of algorithms that align the coordinate systems of the three-dimensional unstructured elements so that they conform can be found in Warburton [36] and Sherwin *et al.* [30].

2.2. *Quadrature*

We take advantage of the tensor product element coordinate systems to perform integration. The integrations over each element can be performed as a set of one-dimensional integrals using Gauss quadrature. If we used the reference coordinate systems this would be very expensive since the limits of the "collapsed" elements are not constant.

We first describe the choice of quadrature type for integrating each direction. We will then motivate the inclusion of quadrature with nonconstant weights in order to reduce the number of points we use.

In two dimensions we consider integrals of the form:

$$\begin{aligned} \int_{\text{Physical}} f(\mathbf{x})\, dx\, dy &= \int_{\text{Reference}} f(\mathbf{x}(\mathbf{r}))\frac{\partial(\mathbf{x})}{\partial(\mathbf{r})}\, dr\, ds, \\ &= \int_{\text{Tensor}} f(\mathbf{x}(\mathbf{r}(\mathbf{a})))\frac{\partial(\mathbf{x})}{\partial(\mathbf{r})}\frac{\partial(\mathbf{r})}{\partial(\mathbf{a})}\, da\, db. \end{aligned}$$

TABLE I

Element	a	b	c
Triangle	*GLL*	$GRJ_{1,0}$	—
Quadrilateral	*GLL*	*GLL*	—
Tetrahedron	*GLL*	$GRJ_{1,0}$	$GRJ_{2,0}$
Pyramid	*GLL*	*GLL*	$GRJ_{2,0}$
Prism	*GLL*	*GLL*	$GRJ_{1,0}$
Hexahedron	*GLL*	*GLL*	*GLL*

Note. GLL Implies Gauss–Lobatto–Legendre which is the Gauss quadrature for a constant weight function with both $x = \pm 1$ points endpoints included. $GRJ_{\alpha,\beta}$ implies Gauss–Radau–Jacobi quadrature with (α, β) weights and the endpoint $x = -1$ included.

In three dimensions:

$$\int_{\text{Physical}} f(\mathbf{x})\, dx\, dy\, dz = \int_{\text{Reference}} f(\mathbf{x}(\mathbf{r}))\frac{\partial(\mathbf{x})}{\partial(\mathbf{r})}\, dr\, ds\, dt,$$
$$= \int_{\text{Tensor}} f(\mathbf{x}(\mathbf{r}(\mathbf{a})))\frac{\partial(\mathbf{x})}{\partial(\mathbf{r})}\frac{\partial(\mathbf{r})}{\partial(\mathbf{a})}\, da\, db\, dc.$$

We use the Gauss weights that will perform the discrete integral of a function as a sum:

$$\int_{-1}^{1}(1-z)^{\alpha}(1+z)^{\beta} f(z)\, dz = \sum_{i=0}^{N-1} f\left(z_i^{\alpha,\beta}\right) w_i^{\alpha,\beta}.$$

This will be used in each of the d directions in the d-dimensional elements. In Table I we show the type of Gaussian quadrature we use in each of the a, b, and c directions.

2.3. *Polynomial Bases*

2.3.1. Modal basis. A modal basis for triangle elements has been presented by Dubiner [10] and Sherwin and Karniadakis [32], and it was applied to fluid dynamics problems by Sherwin and Karniadakis [33] and to geophysical fluid dynamics problems by Wingate and Boyd [38]. We include it here as an example of the type of basis we use for the unstructured elements. Further details of the bases we use for quadrilaterals, tetrahedra, prisms, pyramids, and hexahedra can be found in Sherwin [31]. We remark that these bases are tensorial and therefore the sum factorization technique (also known as tensor product factorization) can be applied for the efficient evaluation of matrix–vector products; see [17, Section 4.1.5, pp. 124–132]. There are also alternative nodal bases for the unstructured elements discussed by Chen and Babuška [7, 8], Hesthaven [16], and Warburton *et al.* [37].

2.3.2. Triangle basis. We present here a basis which is a set of tensor products with respect to the tensor product coordinates for the triangle and polynomials with respect to the reference element coordinates. It maintains numerical linear independence up to high orders due to the construction of the interior modes from Jacobi polynomials $P_n^{\alpha,\beta}$ with carefully chosen (α, β) coefficients to ensure that mode shapes do not become too similar. Increasing α shifts the roots of the Jacobi polynomials away from the coordinate singularity at $b = 1$

as demonstrated in Sherwin and Karniadakis [32] and hence the modes are prevented from having the same shape at this vertex.

The form of the basis is:

Vertex modes:

$$\phi^{\text{vertex}_1} = \left(\frac{1-a}{2}\right)\left(\frac{1-b}{2}\right),$$
$$\phi^{\text{vertex}_2} = \left(\frac{1+a}{2}\right)\left(\frac{1-b}{2}\right),$$
$$\phi^{\text{vertex}_3} = \left(\frac{1+b}{2}\right).$$

Edge modes ($2 \leq m; 1 \leq n, m < N; m+n < N$):

$$\phi_m^{\text{edge}_1} = \left(\frac{1+a}{2}\right)\left(\frac{1-a}{2}\right)P_{m-2}^{1,1}(a)\left(\frac{1-b}{2}\right)^m,$$
$$\phi_n^{\text{edge}_2} = \left(\frac{1+a}{2}\right)\left(\frac{1-b}{2}\right)\left(\frac{1+b}{2}\right)P_{n-1}^{1,1}(b),$$
$$\phi_n^{\text{edge}_3} = \left(\frac{1-a}{2}\right)\left(\frac{1-b}{2}\right)\left(\frac{1+b}{2}\right)P_{n-1}^{1,1}(b).$$

Interior modes ($2 \leq m; 1 \leq n, m < N; m+n < N$):

$$\phi_{mn}^{\text{interior}} = \left(\frac{1+a}{2}\right)\left(\frac{1-a}{2}\right)P_{m-2}^{1,1}(a)\left(\frac{1-b}{2}\right)^m\left(\frac{1+b}{2}\right)P_{n-1}^{2m-1,1}(b).$$

We represent this basis graphically for $N = 5$ in Fig. 1; the highest mode is quartic.

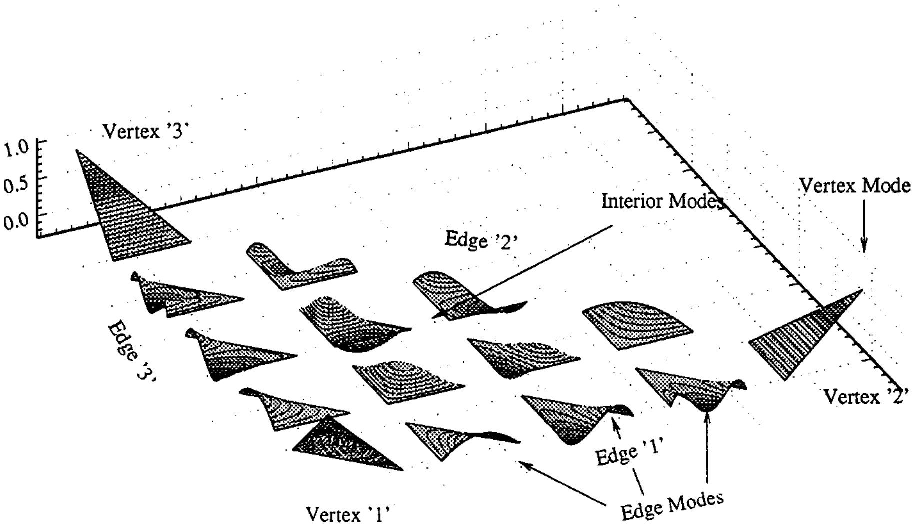

FIG. 1. Mode shapes for the triangle modal basis with $N = 5$.

2.4. *Polynomial Support*

The polynomial bases represented in Sherwin [31] for the hybrid elements have the following support in terms of the polynomial order N and the local coordinate systems of the standard element coordinate systems (r, s) for the two-dimensional elements and (r, s, t) for the three-dimensional elements (the bases here are only used to describe the polynomial spaces; the actual bases used in the method have been described in the previous section). For more general polynomial bases with different degrees in each direction, see Karniadakis and Sherwin [17, Section 3.2].

Element	Polynomial space P_N
Triangle	$\text{Span}\{r^i s^j \mid 0 \le i + j \le N\}$
Quadrilateral	$\text{Span}\{r^i s^j \mid 0 \le i, j \le N\}$
Tetrahedron	$\text{Span}\{r^i s^j t^k \mid 0 \le i + j + k \le N\}$
Pyramid	$\text{Span}\{r^i s^j t^k \mid 0 \le i + j + k \le N\}$
Prism	$\text{Span}\{r^i s^j t^k \mid 0 \le i + k \le N, 0 \le j \le N\}$
Hexahedron	$\text{Span}\{r^i s^j t^k \mid 0 \le i \le N, 0 \le j \le N, 0 \le k \le N\}$

2.5. *Continuous and Discrete Elliptic Problems*

We consider the following model elliptic problem on a bounded Lipschitz region $\Omega \subset R^d$ with boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$. Dirichlet boundary conditions are imposed on Γ_D, a closed subset of $\partial\Omega$ with positive measure, and Neumann conditions on Γ_N.

$$\begin{cases} (-\nabla^2 + \lambda)u = f \ (\lambda \ge 0) & \text{in } \Omega, \\ u = u_0 & \text{on } \Gamma_D, \\ \frac{\partial u}{\partial n} = g & \text{on } \Gamma_N. \end{cases}$$

More general linear, self adjoint, second order elliptic problems and boundary conditions could be considered as well. The standard variational formulation of this problem is: Find $u \in V = H_D^1(\Omega) = \{v \in H^1(\Omega) : v = 0 \ \text{ on } \ \Gamma_D\}$ such that

$$a(u, v) = F(v), \quad \forall v \in V, \tag{1}$$

where

$$a(u, v) = \int_\Omega (\nabla u \cdot \nabla v + \lambda u v)\, dx \quad \text{and} \quad F(v) = \int_\Omega f v\, dx + \int_{\Gamma_N} g v\, ds.$$

We assume that the domain Ω is a union of the spectral elements described previously,

$$\Omega = \bigcup_{k=1}^{K} \Omega_k,$$

where each Ω_k is the affine image of the reference triangle or square in two dimensions and of the reference tetrahedron, hexahedron, pyramid, or prism in three dimensions. Let τ_K be the finite element mesh defined by the spectral elements Ω_k. The spectral element space is defined as

$$V^{N,K} = \left\{ v \in V : v\big|_{\Omega_k} \in P_N, k = 1, \dots, K \right\},$$

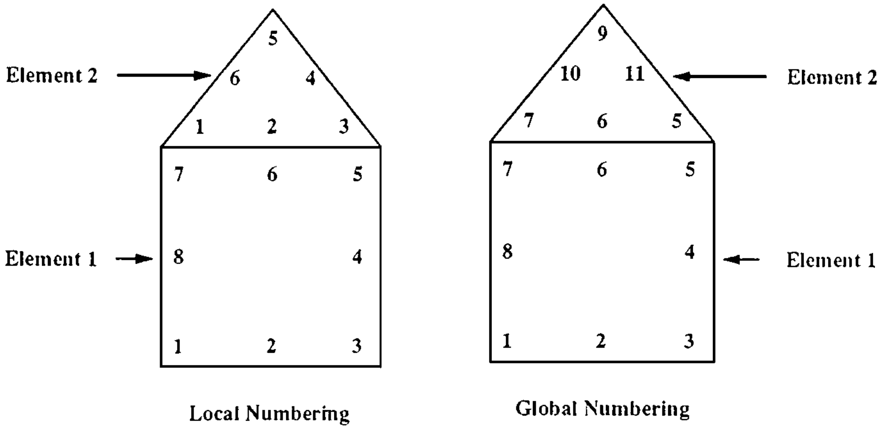

FIG. 2. Illustration of local and global numbering for a domain containing one quadrilateral and one triangular element. Here the expansion order is $N = 3$, and we only show the boundary modes.

where P_N is the proper polynomial space on each element; see Section 2.4. The standard Galerkin formulation of (1) is: Find $u \in V^{N,K}$ such that

$$a_{N,K}(u, v) = F_{N,K}(v) \quad \forall v \in V^{N,K}, \tag{2}$$

where $a_{N,K}(\cdot, \cdot)$ and $F_{N,K}(\cdot)$ are obtained from $a(\cdot, \cdot)$ and $F(\cdot)$ by using the numerical quadrature rules described in Subsection 2.2. The stiffness matrix and load vector of this discrete system are assembled from their elemental contributions on each Ω_k by means of the Z operator, described in detail in Henderson [15], that assembles the local coefficients into the global coefficients and ensures C^0 continuity.

In order to illustrate this global assembly procedure, we consider a global domain made up of two elements as shown in Fig. 2. Only the boundary modes of each element are involved, as the interior modes are completely decoupled. The expansion order shown here is $N = 3$ which means there are six boundary modes on the triangle and eight boundary modes on the quadrilateral. The total number of local boundary modes is therefore $N_{\text{local}} = 14$. Since three modes meet along the connecting edge the number of global boundary modes is eleven and so for this case Z is a 14 × 11 matrix as shown in Fig. 3.

The superscripts denote the local or global nodal number and the subscripts denote the element number. The absolute column sum gives the multiplicity of a mode and we see that columns 5, 6, and 7 all have a multiplicity of 2. We also note that the absolute row sum is always 1 since there is only one value of each local mode. The Z matrix groups together the degrees of freedom for the boundary mode shapes for all the elements. The remaining interior degrees of freedom are collected in one group per element. A typical matrix structure of the resulting discrete system (2) is shown in Fig. 4, where $A = Z^t \tilde{A} Z$ and $\tilde{A}_{i,j} = a_{N,K}(\phi_i, \phi_j)$, with ϕ_i and ϕ_j boundary modes.

We note that the global stiffness matrix is never formed; only the elemental stiffness matrices are formed and factored. The action of the global stiffness matrix on a vector is computed by subassembly using the elemental stiffness matrices.

We can take advantage of the bandedness of the A boundary–boundary matrix and the decoupled C matrix by using the well-known substructuring (or static condensation) technique; see Chap. 4 in Smith *et al.* [34]. By implicitly eliminating the interior degrees of

$$u_l = \begin{bmatrix} u_1^1 \\ u_2^1 \\ u_3^1 \\ u_4^1 \\ u_5^1 \\ u_6^1 \\ u_7^1 \\ u_8^1 \\ \cdots \\ u_1^2 \\ u_2^2 \\ u_3^2 \\ u_4^2 \\ u_5^2 \\ u_6^2 \end{bmatrix} = \begin{bmatrix} 1 & & & & & & & & & & \\ & 1 & & & & & & & & & \\ & & 1 & & & & & & & & \\ & & & 1 & & & & & & & \\ & & & & 1 & & & & & & \\ & & & & & 1 & & & & & \\ & & & & & & 1 & & & & \\ & & & & & & & 1 & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ & & & & & & 1 & & & & \\ & & & & & 1 & & & & & \\ & & & & 1 & & & & & & \\ & & & & & & & & & & 1 \\ & & & & & & & & 1 & & \\ & & & & & & & & & 1 & \end{bmatrix} \begin{bmatrix} u_1^g \\ u_2^g \\ u_3^g \\ u_4^g \\ u_5^g \\ u_6^g \\ u_7^g \\ u_8^g \\ u_9^g \\ u_{10}^g \\ u_{11}^g \end{bmatrix}$$

FIG. 3. Z-matrix map from global to local degrees of freedom.

freedom, the system can be rewritten as

$$\begin{bmatrix} A - BC^{-1}B^T & 0 \\ B^T & C \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_b \\ \boldsymbol{u}_I \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_b - BC^{-1}\boldsymbol{f}_I \\ \boldsymbol{f}_I \end{bmatrix},$$

where bold letters denote vectors of coefficients, $\boldsymbol{u}_b$ contains the unknown global boundary coefficients, and $\boldsymbol{u}_I$ contains the unknown interior coefficients.

Once the boundary coefficients $\boldsymbol{u}_b$ are known, the interior coefficients $\boldsymbol{u}_I$ are easily found by solving K decoupled interior problems. The matrix $S = A - BC^{-1}B^T$ is known as the Schur complement of the original linear system. We determine $\boldsymbol{u}_b$ by solving iteratively the Schur complement system

$$S\boldsymbol{u}_b = \tilde{\boldsymbol{f}}_b = \boldsymbol{f}_b - BC^{-1}\boldsymbol{f}_I \tag{3}$$

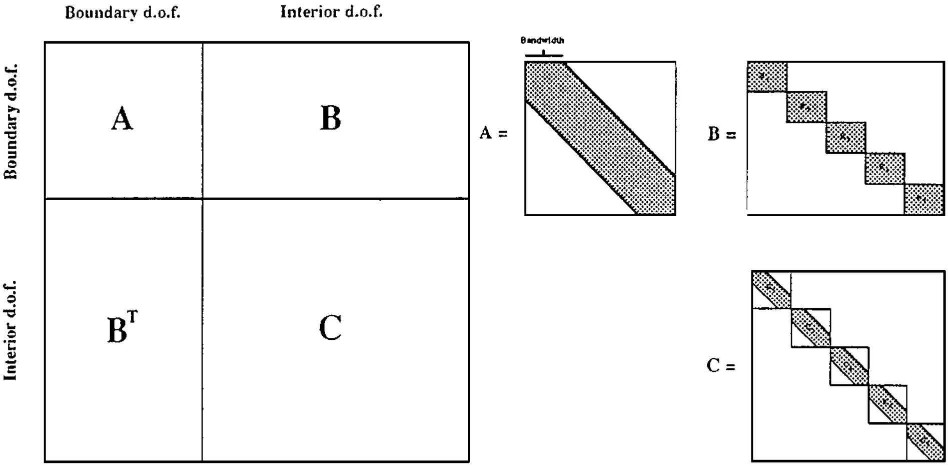

FIG. 4. Form of the global operator matrix. Notice the sparsity of A and the block nature of B and C.

by a preconditioned conjugate gradient method (PCG). The preconditioner is based on overlapping Schwarz techniques, described in the next section. We remark that the iterative solution of the Schur complement system by PCG does not require one to explicitly form S, but requires only the action of S on a given vector. This is computed using the definition of S by solving K uncoupled Dirichlet problems corresponding to the diagonal blocks of C. Each block can be factored before the iteration so that each Dirichlet solve during the iteration can be accomplished by a simple back-substitution.

It is easy to see that the linear system (3) is the matrix representation of the following Galerkin problem (see Chapter 4.6 in Smith *et al.* [34]): Find $u_b \in \tilde{V}^{N,K}$ such that

$$a_{N,K}(u_b, v) = F_{N,K}(v) \quad \forall v \in \tilde{V}^{N,K}, \tag{4}$$

where $\tilde{V}^{N,K}$ is the subspace of the discrete harmonic functions of $V^{N,K}$. We recall that a function $v \in V^{N,K}$ is discrete harmonic if

$$a_{N,K}(v, \phi_I) = 0$$

for every interior mode $\phi_I \in V^{N,K}$, i.e., for every mode that vanishes on the interface $\Gamma = \bigcup_{k=1}^{K} \partial\Omega_k$. In matrix terms, $\boldsymbol{v} = (\boldsymbol{v}_b, \boldsymbol{v}_I)$ is discrete harmonic if

$$B^T \boldsymbol{v}_b + C\boldsymbol{v}_I = 0.$$

3. OVERLAPPING SCHWARZ PRECONDITIONERS

For simplicity, we present here the basic additive variant of the preconditioner. More general multiplicative or hybrid variants can be considered as well; see Smith *et al.* [34].

We recall that τ_K is the finite element triangulation defined by $\Omega = \bigcup_{k=1}^{K} \Omega_k$. We partition τ_K into K_s disjoint groups D_i of adjacent elements, called subdomains: $\Omega = \bigcup_{i=1}^{K_s} D_i$. In order to have an overlapping partition of Ω, each subdomain D_i is extended to a larger subdomain D_i', consisting of all elements of τ_K within a distance δ from D_i. The minimal overlap consists of one layer of elements outside D_i.

3.1. *Matrix Form of the Preconditioner*

Our overlapping additive Schwarz preconditioner $\hat{S}^{-1}$ for the Schur complement matrix S is based on the solution of a coarse problem and on the solutions of local problems on the subdomains D_i', defined as follows.

(a) Let R_0 be the restriction matrix returning only the vertex modes of a global vector ϕ_b of boundary coefficients and let S_0 be the vertex block of the Schur complement S. Then $R_0^T S_0^{-1} R_0$ represents the solution of a coarse problem with discrete harmonic piecewise linear elements on the mesh τ_K (i.e., involving only the discrete harmonic extensions of vertex modes).

(b) Let R_i be restriction matrices associated with the subdomains D_i'; given a global vector ϕ_b, R_i returns only the local boundary coefficients ϕ_b^i of basis functions with support in D_i'. Let S_i be the local Schur complement associated with the subdomain D_i'. Then $R_i^T S_i^{-1} R_i$ represents the solution of the ith local problem on D_i' involving only the local boundary coefficients ϕ_b^i and satisfying zero Dirichlet boundary conditions on $\partial D_i' \backslash \partial\Omega$.

The matrix form of our preconditioner is then:

$$\hat{S}^{-1} = R_0^T S_0^{-1} R_0 + \sum_{i=1}^{K_s} R_i^T S_i^{-1} R_i. \tag{5}$$

We remark that, in practice, the action of S_i^{-1} can be calculated without explicitly forming S_i by solving a problem on D_i' (with zero Dirichlet boundary conditions on $\partial D_i' \backslash \partial\Omega$ and right-hand side different from zero only for the local boundary modes associated with ϕ_b^i); see [34, Chap. 4.2.1] for this well-known technique. The same technique can be applied to compute the action of S_0^{-1} without explicitly forming S_0. We also remark that actions of both S_i^{-1} and S_0^{-1} can be approximated by the solution of local and coarse problems with approximate Schur complements; see [34, Chap. 4].

3.2. *Space Decomposition and Convergence Rate Estimates*

Schwarz preconditioners can also be described in functional terms by introducing a decomposition of the discrete space of the problem. This is usually done in order to obtain a theoretical analysis of the preconditioner by using the abstract Schwarz framework (see [34]). This preconditioner is associated with the following decomposition of the discrete space $\tilde{V}^{N,K}$, defined in the previous section, into a coarse space V_0 and local spaces V_i, associated with the subdomains D_i':

$$\tilde{V}^{N,K} = V_0 + \sum_{i=1}^{K_s} V_i.$$

(a) The coarse space is defined as

$$V_0 = \tilde{V}^{1,K};$$

i.e., it consists of discrete harmonic piecewise linear functions on the mesh τ_K. Computationally cheaper choices are possible, such as piecewise linear function on a coarser mesh associated with the subdomains D_i. The associated coarse stiffness matrix is denoted by S_0.

(b) The local spaces V_i consist of piecewise *hp* polynomials satisfying zero Dirichlet boundary conditions on the internal subdomain boundaries $\partial D_i' \backslash \partial\Omega$ and the original boundary conditions on $\partial D_i' \cap \partial\Omega$. Hence, for an internal subdomain D_i' the associated local space is

$$V_i = \tilde{V}^{N,K} \cap H_0^1(D_i').$$

If we define a coarse projection $T_0 : \tilde{V}^{N,K} \to V_0$ and K_s local projections $T_i : \tilde{V}^{N,K} \to V_i$ by

$$a_{N,K}(T_i u, v) = a_{N,K}(u, v) \quad \forall v \in V_i,\ i = 0, 1, \ldots, K_s,$$

then it is easy to see that our preconditioned system

$$\hat{S}^{-1} S = R_0^T S_0^{-1} R_0 S + \sum_{i=1}^{K_s} R_i^T S_i^{-1} R_i S$$

is exactly the matrix form of the additive Schwarz operator

$$T = T_0 + T_1 + \cdots + T_{K_s}.$$

Pavarino [24] analyzed an overlapping preconditioner for p-version finite elements on quadrilateral and hexahedral meshes. The method studied in [24] has generous overlap since the subdomains consist of the union of the elements sharing each vertex. If the technical tools developed in [24] (in particular Section 5, pp. 510–512) are used, it is straightforward to extend to our preconditioner the same uniform bound on the condition number of the iteration operator in case of quadrilateral meshes in two dimensions and hexahedral meshes in three dimensions:

THEOREM 1. *If τ_K is a quadrilateral or a hexahedral mesh, then*

$$cond(T) = cond(\hat{S}^{-1}S) \leq C,$$

with C constant independent of N, K, and K_s.

The extension of the analysis to triangular or nonhexahedral meshes is still an open problem. The numerical results reported in the next section indicate that a uniform bound as in Theorem 1 holds for hybrid meshes as well. Some progress for nonoverlapping domain decomposition methods and tetrahedral meshes can be found in Bica [3].

An additional open problem is the extension of our method based on generous overlap to a more flexible method based on small overlap techniques. This has been done for hexahedral meshes and GLL nodal basis, where a small overlap of a few GLL points outside each spectral element led to a successful algorithm; see, e.g., Pahl [23], Casarin [4], Fischer [11], Pavarino [26], and Rønquist [29].

3.3. *Complexity Estimates*

We present here some simple complexity estimates of the computational costs of the PCG algorithm with our overlapping Schwarz preconditioner. For simplicity, we only consider the serial case. More complete estimates should consider distributed memory parallel architectures and estimate not just operation counts but also memory and interprocessor communication costs. For a study of the parallel complexity of overlapping Schwarz methods for h-version finite elements and optimal choices of the coarse problem, we refer to Chan and Shao [6].

The cost of solving the Schur complement system $S\boldsymbol{u}_b = \tilde{\boldsymbol{f}}_b$ by CG or PCG is given by

$$\#_{\text{it}} \times C_{\text{it}},$$

where $\#_{\text{it}}$ is the number of iterations required and C_{it} is the cost per iteration (floating point operation count).

(a) Cost C_{CG} of CG.

(a1) $\#_{\text{it}}$ is proportional to $\sqrt{cond(S)}$ but we do not have a theoretical estimate of $cond(S)$ for our unstructured spectral elements. For h-version finite elements, $cond(S) = O(h^{-1})$, an improvement over the conditioning of the full Laplacian operator L, $cond(L) = O(h^{-2})$. Spectral elements are more ill-conditioned than h-version finite elements: for example, for GLL spectral elements on hexahedral meshes we have $cond(L) = O(KN^3)$

(see Couzy and Deville [9]). Numerical results by Sherwin *et al.* [30] indicate that for our unstructured spectral elements $cond(S) = O(KN)$ in two dimensions; hence

$$\#_{\text{it}} = O\big(K^{1/2}N^{1/2}\big).$$

(a2) The cost per iteration C_{it} is dominated by the cost of matrix–vector products $S\boldsymbol{v}$. These products can be computed either by subassembly of the local Schur complements S_i for each element Ω_i (when the S_i are formed in a preprocessing stage before the CG iteration) or by the definition, i.e., by K interior solves, one for each element Ω_i as explained in Section 2.5. The cost C_i of the ith interior solve depends on the solver employed. Let n_i be the dimension of the ith interior problem on Ω_i; we have $n_i = O(N^2)$ in 2D and $n_i = O(N^3)$ in 3D. If we factor the local interior stiffness matrices before the iteration then each local solve has the cost of a back substitution: $C_i = O(n_i^2)$. I.e., $C_i = O(N^4)$ in 2D and $C_i = O(N^6)$ in 3D. If we use a fast interior solver such as the fast diagonalization methods (FDM), available for hexahedral elements, then the cost reduces to $O(N^4)$ in 3D; see Couzy and Deville [9]. We can also use a spectral version of the multigrid method, the multi-p method (see Katz and Hu [19]), or a few steps of an iterative method to further reduce the cost to the optimal value $O(N^3)$ in 3D. Of course, in this last case the use of inexact interior solvers might increase the number of CG iterations required.

In conclusion, we have $C_i = O(N^{3\alpha})$, with $\alpha \geq 1$ depending on the interior solver employed, and in 2D

$$C_{\text{CG}} = O\left(K^{1/2}N^{1/2}\sum_{i=1}^{K} C_i\right) = O\big(K^{3/2}N^{3\alpha+1/2}\big).$$

(b) Cost C_{PCG} of PCG with overlapping Schwarz preconditioner.

(b1) In this case, from Theorem 1 for hexahedral elements and from the numerical experiments presented in the next section for general unstructured elements,

$$\#_{\text{it}} = \sqrt{cond(\hat{S}^{-1}S)} = O(1).$$

(b2) C_{it} is now dominated by the cost of matrix–vector products $S\boldsymbol{v}$ and the cost of the overlapping preconditioner $\hat{S}^{-1}\boldsymbol{v}$ (the restriction and interpolation operations with the matrices R_i and R_i^T have lower order complexity as the other vector operations in each CG iteration). The cost of $\hat{S}^{-1}\boldsymbol{v}$ is the sum of the cost of the local solvers on each subdomain D_i' and the cost of the coarse solver. As described in Subsection 3.1, each subdomain solve requires the solution of a Dirichlet problem on D_i', which has $O(K_iN^3)$ unknowns, where K_i is the number of spectral elements in subdomain D_i'. If we use the same solver we used for the interior problems, the cost of each subdomain solve is then $O(K_i^\alpha N^{3\alpha})$ and the sum of the subdomain solves has cost $O(\sum_{i=1}^{K_s} K_i^\alpha N^{3\alpha}) = O(N^{3\alpha}\sum_{i=1}^{K_s} K_i^\alpha)$. The cost of the coarse solver is proportional to the dimension n_0 of the coarse space: if we use a direct solver the cost is $O(n_0^3)$ and if we use an iterative (inexact) solver the cost could be reduced to $O(n_0)$. For simplicity, we took $n_0 = K$ in our implementation that uses all vertex modes, but a more sophisticated implementation requiring only one degree of freedom per subdomain would yield $n_0 = K_s$. The cost of the coarse solve is then $O(n_0^\beta)$, where $\beta \geq 1$

depends on the coarse solver employed. In conclusion, we have

$$C_{\text{it}} = cost(S\mathbf{v}) + cost(\hat{S}^{-1}\mathbf{v}) = O\left(KN^{3\alpha} + N^{3\alpha}\sum_{i=1}^{K_s} K_i^{\alpha} + n_0^{\beta}\right)$$

$$= O\left(N^{3\alpha}\left(K + \sum_{i=1}^{K_s} K_i^{\alpha}\right) + n_0^{\beta}\right),$$

and since $\#_{\text{it}} = O(1)$, we have

$$C_{\text{PCG}} = O\left(N^{3\alpha}\left(K + \sum_{i=1}^{K_s} K_i^{\alpha}\right) + n_0^{\beta}\right).$$

If the coarse problem is not too large, C_{PCG} is asymptotically better than C_{CG} with respect to N, while it is more difficult to estimate the exponent of K. Supposing subdomains of approximately the same size, we have $K_i = K/K_s + K_\delta$, where K_δ is the number of elements in the overlapping layer of each subdomain. If we use an optimal solver with $\alpha = 1$, then $\sum_{i=1}^{K_s} K_i^{\alpha} = \sum_{i=1}^{K_s}(K/K_s + K_\delta) = K + K_s K_\delta \leq const.K$, because each element belongs to a fixed maximal number of subdomains. Then, in this case PCG has a better complexity estimate than CG also with respect to K. We could also use as local solver a fixed number of iterations of a linear iterative method (this will have an optimal cost with $\alpha = 1$, but of course it will be no longer guaranteed that $\#_{\text{it}} = O(1)$). We remark that using an inner CG iteration as local subdomain solver would lead to worse complexity estimates because the cost of the local solver would then be $O(K_i^{3/2}N^{3\alpha+1/2})$, as estimated in point (a).

4. NUMERICAL EXPERIMENTS

In this section, we report on the results of numerical experiments with the overlapping Schwarz preconditioner $\hat{S}^{-1}$ described in Section 3. The elliptic problem considered is the one described in Section 2.5 with $\lambda = 1$, domain Ω specified below and right-hand side, and boundary condition prescribed such that the exact solution is $\sin(\pi x)\sin(\pi y)$. The spectral element discretization of this problem has been described in Section 2. The results have been obtained using the unstructured spectral element code NekTar [32, 33, 31, 37, 17, 30] (see also http://www.cfm.brown.edu/people/tcew/nektar.html) running on Silicon Graphics workstations. The NekTar code is constructed using an object-oriented programming model implemented in C++ with matrix operations computed using fortran LAPACK/BLAS routines.

The domain has been partitioned into subdomains using the mesh partitioning code PMETIS [18]. The preconditioner is accelerated by the conjugate gradient method, since the Schur complement is symmetric and positive definite. For these computations the initial guess is always the null vector and the iteration is stopped when the norm of the relative residual is less than 10^{-10} (or 10^{-15} in the last test in Table VI). The overlap is kept fixed to the minimal one-element layer around each subdomain. In all figures, the subdomains are shown without overlap. The coarse and local problems are solved exactly by a direct method, except in the 3D runs of Tables V and VI, where for memory reasons we have used a diagonally preconditioned conjugate gradient method. Computationally cheaper inexact solvers, such as ILU(k), multi-p, or other iterative methods, could be used instead.

TABLE II

2D Block Mesh, 128 = $8^2 \cdot 2$ Elements, 4 Subdomains

N	Iter.	Rel. err.	Cond. n.	$\lambda_{\max}$	$\lambda_{\min}$
2	18	1.02e-10	5.11	5.00	0.9791
3	19	2.92e-11	5.11	5.00	0.9791
4	19	3.97e-11	5.11	5.00	0.9791
5	19	9.59e-11	5.11	5.00	0.9792
6	19	9.28e-11	5.11	5.00	0.9792
7	19	3.15e-11	5.11	5.00	0.9791
8	19	3.20e-11	5.11	5.00	0.9791
9	19	3.08e-11	5.11	5.00	0.9791
10	19	2.96e-11	5.11	5.00	0.9791
11	19	2.93e-11	5.11	5.00	0.9791
12	19	3.37e-11	5.11	5.00	0.9791

We consider the following four test problems.

(1) $\Omega = [-1, 1]^2$ is a square domain in the plane;

(a) τ_K is a structured mesh with $K = 128 = 8^2 \cdot 2$ triangular elements, $K_s = 4$ subdomains, and N varies from 2 to 12; see Table II.

(b) τ_K is a structured mesh with $K = 512 = 16^2 \cdot 2$ triangular elements, $K_s = 16$ subdomains, and N varies from 2 to 12; see Table III.

(2) Ω is a rectangular domain in the plane, τ_K is an unstructured mesh with $K = 780$ triangular and quadrilateral elements, $K_s = 4, 8, 16$ subdomains, and N varies from 2 to 12; see Figs. 5–7 and Table IV.

(3) $\Omega = [-1, 1] \times [-1, 1] \times [0, 2]$ is a cubic domain, τ_K is a structured mesh with $K = 3072 = 8^3 \cdot 6$ tetrahedral elements, $K_s = 8$ subdomains, and N varies from 2 to 8; see Fig. 9 and Table V.

(4) Ω is the three-dimensional domain shown in Fig. 10, representing a one-element thick face mask, τ_K is a mesh with $K = 3295$ prismatic elements, $K_s = 8$ subdomains, and N varies from 2 to 6; see Table VI.

TABLE III

2D Block Mesh, 512 = $16^2 \cdot 2$ Elements, 16 Subdomains

N	Iter.	Rel. err.	Cond. n.	$\lambda_{\max}$	$\lambda_{\min}$
2	24	1.31e-11	5.11	5.00	0.9786
3	24	1.46e-11	5.11	5.00	0.9790
4	24	1.42e-11	5.11	5.00	0.9791
5	24	1.40e-11	5.11	5.00	0.9791
6	24	1.41e-11	5.11	5.00	0.9791
7	24	1.43e-11	5.11	5.00	0.9791
8	24	1.43e-11	5.11	5.00	0.9791
9	24	1.43e-11	5.11	5.00	0.9791
10	24	1.42e-11	5.11	5.00	0.9791
11	24	1.39e-11	5.11	5.00	0.9791
12	24	1.37e-11	5.11	5.00	0.9791

TABLE IV
qt2 Mesh, 780 Elements

K_s	N	Iter.	Rel. err.	Cond. n.	$\lambda_{\max}$	$\lambda_{\min}$
4	2	16	1.05e-9	3.02	3.00	0.9930
	3	16	2.20e-9	3.00	3.00	0.9993
	4	16	2.04e-9	3.01	3.00	0.9968
	5	16	2.09e-9	3.01	3.00	0.9969
	6	16	2.13e-9	3.01	3.00	0.9969
	7	16	2.15e-9	3.01	3.00	0.9969
	8	16	2.16e-9	3.01	3.00	0.9969
	9	16	2.17e-9	3.01	3.00	0.9969
	10	16	2.18e-9	3.01	3.00	0.9969
	11	16	2.18e-9	3.01	3.00	0.9969
	12	16	2.18e-9	3.01	3.00	0.9969
8	2	21	2.87e-9	4.49	4.46	0.9924
	3	22	1.01e-9	4.51	4.47	0.9924
	4	22	1.02e-9	4.51	4.47	0.9922
	5	22	1.05e-9	4.51	4.47	0.9932
	6	22	1.06e-9	4.50	4.47	0.9937
	7	22	1.07e-9	4.50	4.47	0.9940
	8	22	1.07e-9	4.50	4.47	0.9942
	9	22	1.07e-9	4.50	4.47	0.9943
	10	22	1.08e-9	4.50	4.47	0.9944
	11	22	1.08e-9	4.50	4.47	0.9945
	12	22	1.08e-9	4.50	4.47	0.9945
16	2	23	2.37e-9	5.05	5.00	0.9901
	3	23	2.56e-9	5.04	5.00	0.9924
	4	23	2.52e-9	5.04	5.00	0.9918
	5	23	2.47e-9	5.04	5.00	0.9918
	6	24	9.83e-10	4.98	4.93	0.9902
	7	24	1.03e-9	4.92	4.87	0.9902
	8	23	1.79e-9	4.59	4.55	0.9914
	9	23	1.78e-9	4.59	4.55	0.9914
	10	23	1.78e-9	4.59	4.55	0.9914
	11	23	1.78e-9	4.59	4.55	0.9914
	12	23	1.77e-9	4.59	4.55	0.9914

TABLE V
3D Block Mesh, $3072 = 8^3 \cdot 6$ Elements, 8 Subdomains

N	Iter.	Rel. err.	Cond. n.	$\lambda_{\max}$	$\lambda_{\min}$
2	29	1.38e-11	5.74	5.38	0.9380
3	29	1.15e-11	5.75	5.41	0.9418
4	29	1.62e-11	5.75	5.42	0.9430
5	29	1.63e-11	5.75	5.42	0.9430
6	29	1.20e-11	5.75	5.42	0.9430
7	29	1.67e-11	5.75	5.42	0.9428
8	29	1.36e-11	5.76	5.42	0.9415

TABLE VI
Head Mesh, 3295 Elements, 8 Subdomains

N	Iter.	Rel. err.	Cond. n.	$\lambda_{\max}$	$\lambda_{\min}$
2	17	4.47e-15	3.08	3.00	0.9756
3	18	3.78e-15	3.08	3.00	0.9756
4	19	3.28e-15	3.08	3.00	0.9750
5	20	1.81e-15	3.08	3.00	0.9753
6	20	3.30e-15	3.08	3.00	0.9756

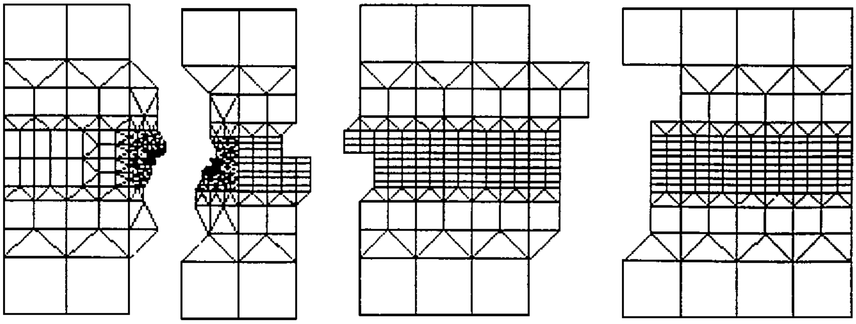

FIG. 5. qt2 mesh, 4 subdomains.

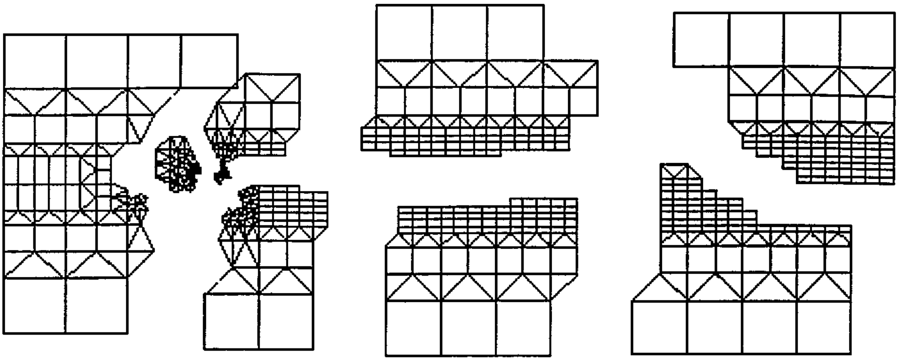

FIG. 6. qt2 mesh, 8 subdomains.

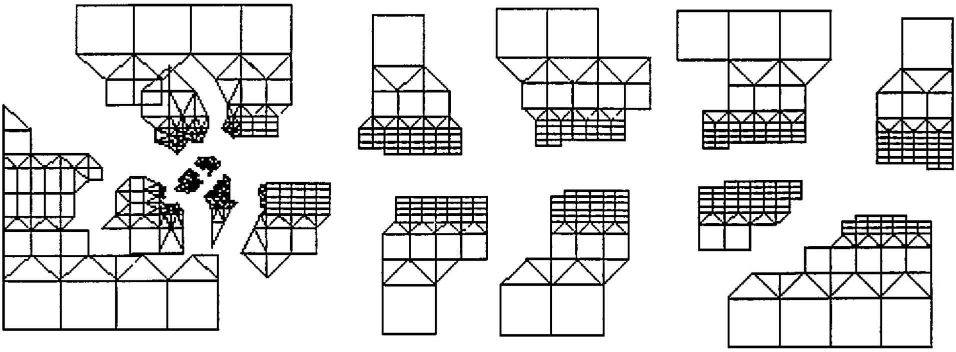

FIG. 7. qt2 mesh, 16 subdomains.

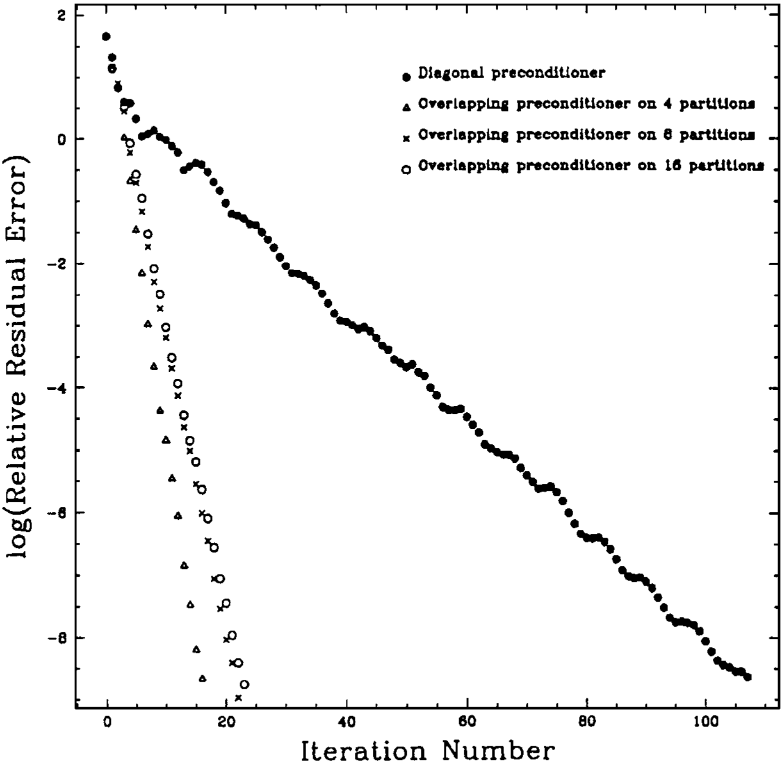

FIG. 8. PCG convergence history from Table IV, mesh qt2, $N = 10$, for diagonal preconditioner and for overlapping additive Schwarz preconditioner on 4, 8, and 16 subdomains.

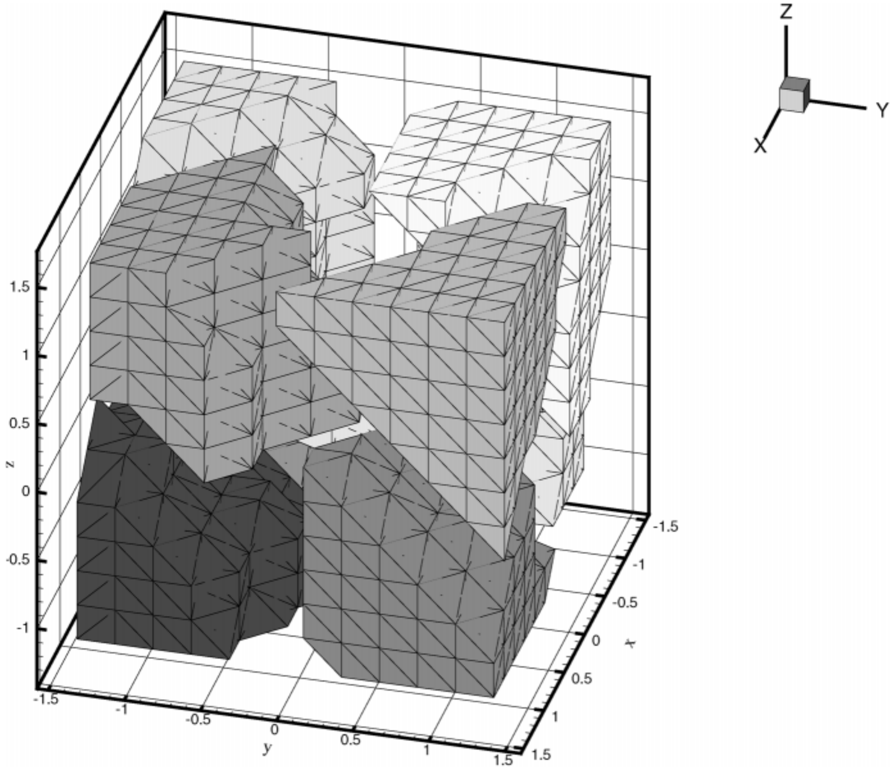

FIG. 9. 3D block mesh, 8 subdomains.

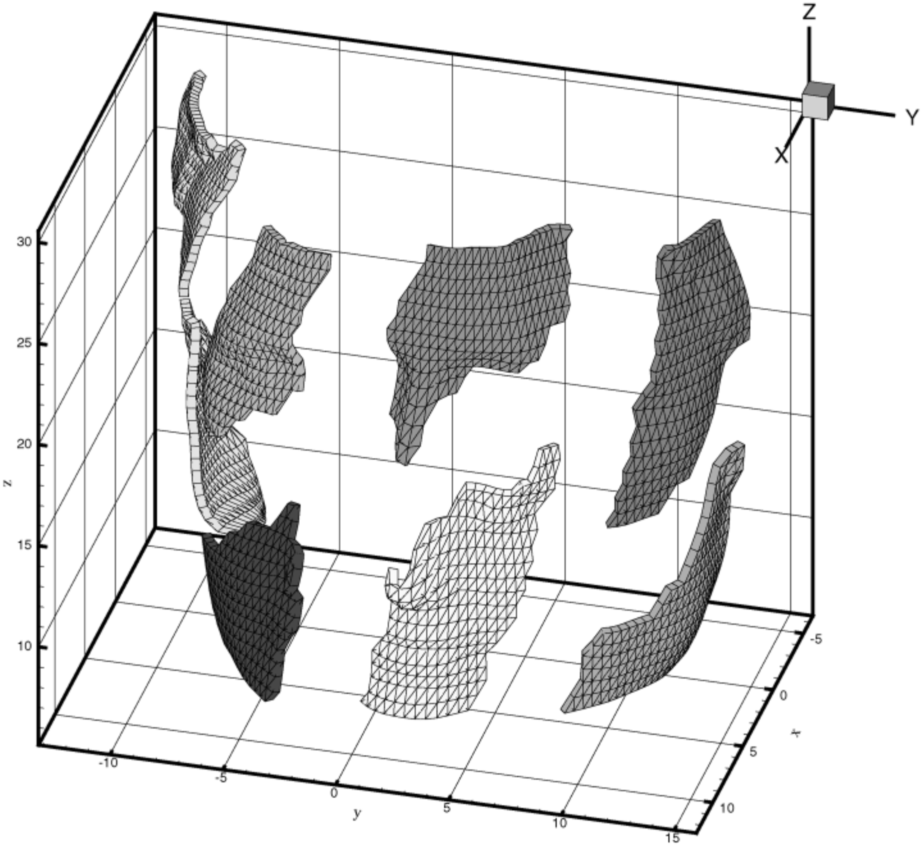

FIG. 10. Head mesh, 8 subdomains.

In all tables, we report the number of conjugate gradient iterations, the relative error with the direct solution, the Lanczos approximation of the condition number, and the extreme eigenvalues of the preconditioned operator $\hat{S}^{-1}S$.

It is clear from the results of the tables that the iteration counts and spectral properties of the proposed method are independent of the spectral degree N. The results with the hybrid mesh reported in Table IV also show that the iteration counts are bounded independent of the number of subdomains K_s (this is less evident in Tables II, III). This bound seems to be the same for both hybrid (Table IV) and block meshes (Tables II, III). In Fig. 8, we have plotted the PCG convergence history for the runs with $N = 10$ and $K_s = 4$, 8, and 16 in Table IV. These three runs with the overlapping Schwarz preconditioner are contrasted with the diagonal preconditioner. For the three-dimensional tests, we could only run cases with smaller values of N, but the results point to the same conclusions: as is the case for standard h-version finite elements, the overlapping Schwarz method proposed is clearly a parallel and scalable solver also for unstructured spectral element discretizations. An efficient and competitive implementation on parallel architectures was beyond the scope of this paper and will require a study of efficient approximate solvers for the local and coarse problems.

REFERENCES

1. I. Babuška and M. Suri, The p and h-p versions of the finite element method, basic principles and properties, *SIAM Rev.* **36**, 578 (1994).

2. C. Bernardi and Y. Maday, Spectral methods, in *Handbook of Numerical Analysis, Volume V: Techniques of Scientific Computing (Part 2)* (North-Holland, Amsterdam, 1997), p. 209.
3. I. Bică, *Iterative Substructuring Algorithms for the p-version Finite Element Method for Elliptic Problems,* PhD thesis, Dept. of Mathematics, Courant Institute of Mathematical Sciences, New York University, September 1997.
4. M. Casarin, Quasi-optimal Schwarz methods for the conforming spectral element discretization, *SIAM J. Numer. Anal.* **34**, 2482 (1997).
5. M. Casarin and S. Sherwin, Low energy bases preconditioning for elliptic substructured solvers based on spectral/hp element discretisations, in preparation.
6. T. F. Chan and J. P. Shao, Parallel complexity of domain decomposition methods and optimal coarse grid size, *Parallel Comput.* **21**, 1033 (1995).
7. Q. Chen and I. Babuška, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle, *Comp. Meths. Appl. Mech. Eng.* **128**, 405 (1995).
8. Q. Chen and I. Babuška, The optimal symmetrical points for polynomial interpolation of real functions in the tetrahedron, *Comp. Meths. Appl. Mech. Eng.* **137**, 89 (1996).
9. W. Couzy and M. Deville, A fast Schur complement method for the spectral element discretization of the incompressible navier-stokes equations, *J. Comp. Phys.* **116**, 135 (1995).
10. M. Dubiner, Spectral methods on triangles and other domains, *J. Sci. Comp.* **6**, 345 (1991).
11. P. F. Fischer, An overlapping schwarz method for spectral element solution of the incompressible Navier-Stokes equations, *J. Comp. Phys.* **133**, 84 (1997).
12. P. F. Fischer and E. Rønquist, Spectral element methods for large scale parallel Navier-Stokes calculations, *Comp. Meths. Appl. Mech. Eng.* **116**, 69 (1994).
13. B. Guo and W. Cao, An additive Schwarz method for the *h-b* version of the finite element method in three dimensions, *SIAM J. Numer. Anal.* **35**, 632 (1998).
14. W. Heinrichs, Spectral collocation on triangular elements, *J. Comp. Phys.* **145**, 743 (1998).
15. R. D. Henderson, *Unstructured Spectral Element Methods: Parallel Algorithms and Simulations,* PhD thesis, Princeton University, 1994.
16. J. Hesthaven, From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, *SIAM J. Numer. Anal.* **35**, 655 (1998).
17. G. E. Karniadakis and S. J. Sherwin, *Spectral/hp Element Methods for CFD* (Oxford University Press, Oxford, 1999).
18. G. Karypis and V. Kumar, Parallel multilevel k-way partitioning scheme for irregular graphs, *SIAM Rev.* **41**, 278 (1999).
19. I. N. Katz and N. Hu, Multi-p methods: Iterative algorithms for the p-version of the finite element analysis, *SIAM J. Sci. Comp.* **16**, 1308 (1995).
20. J. Mandel, Iterative solvers for p-version finite element method in three dimensions, *Comp. Meths. Appl. Mech. Eng.* **116**, 175 (1994). [ICOSAHOM'92, Montpellier, France, June 1992]
21. J. Mandel, Iterative methods for p-version finite elements: Preconditioning thin solids, *Comp. Meths. Appl. Mech. Eng.* **133**, 247 (1996).
22. J. Oden, A. Patra, and Y. Feng, Parallel domain decomposition solver for adaptive *hp* finite element methods, *SIAM J. Numer. Anal.* **34**, 2090 (1997).
23. S. S. Pahl, *Schwarz Type Domain Decomposition Methods for Spectral Element Discretizations,* Master's thesis, Department of Computational and Applied Mathematics, University of the Witwatersrand, Johannesburg, South Africa, December 1993.
24. L. F. Pavarino, Additive Schwarz methods for the p-version finite element method, *Numer. Math.* **66**, 493 (1994).
25. L. F. Pavarino, Domain decomposition algorithms for saddle point problems, in *Domain Decomposition Methods 10*, editd by Mandel, Farhat, and Cai, Contemporary Mathematics (Amer. Math. Soc., Providence, RI, 1998), p. 138.
26. L. F. Pavarino, Indefinite overlapping Schwarz methods for time-dependent Stokes problems, *Comp. Meths. Appl. Mech. Eng.*, (2000), to appear. [Technical Report 99/8, Department of Mathematics, University of Milan, Italy]

27. L. F. Pavarino and O. B. Widlund, A polylogarithmic bound for an iterative substructuring method for spectral elements in three dimensions, *SIAM J. Numer. Anal.* **33**, 1303 (1996).

28. E. Rønquist, A domain decomposition method for elliptic boundary value problems: Application to unsteady incompressible fluid flow, in *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, edited by Chan, Keyes, Meurant, Scroggs, and Voigt (SIAM, Philadelphia, 1992).

29. E. Rønquist, Domain decomposition methods for the steady Stokes equations, in *Proc. of DD11*, edited by Lai, Bjorstad, Cross, and Widlund, DDM.org, 1999, to appear.

30. J. Sherwin, T. Warburton, and G. Karniadakis, Spectral/*hp* methods for elliptic problems on hybrid grids, in *Domain Decomposition Methods 10*, edited by Mandel, Farhat, and Cai, Contemporary Mathematics (Amer. Math. Soc., Providence, RI, 1998), Vol. 218, p. 191.

31. S. J. Sherwin, Hierarchical *hp* finite elements in hybrid domains, *Finite Elements Anal. Design* **27**, 109 (1997).

32. S. J. Sherwin and G. E. Karniadakis, A New triangular and tetrahedral basis for high-order hp finite element methods, *Int. J. Numer. Meth. Eng.* **38**, 3775 (1995).

33. S. J. Sherwin and G. E. Karniadakis, A triangular spectral element method; applications to the incompressible Navier-Stokes equations, *Comp. Meths. Appl. Mech. Eng.* **123**, 189 (1995).

34. B. F. Smith, P. Bjørstad, and W. D. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations* (Cambridge University Press, Cambridge, UK, 1996).

35. B. Szabo and I. Babuška, *Finite Element Analysis* (Wiley-Interscience, New York, 1991).

36. T. Warburton, *Spectral/hp Methods on Polymorphic Multi-Domains: Algorithms and Applications*, PhD thesis, Brown University, Division of Applied Mathematics, 1999.

37. T. Warburton, S. Sherwin, and G. Karniadakis, Spectral basis functions for 2D hybrid/hp elements, *SIAM J. Sci. Comp.* (1999), to appear.

38. B. A. Wingate and J. P. Boyd, Triangular spectral element methods for geophysical fluid dynamics applications, *Houston J. Math.* 305 (1996).

39. B. A. Wingate and M. A. Taylor, Fekete collocation points for triangular spectral elements, manuscript submitted for publication.

40. B. A. Wingate and M. A. Taylor, The natural function space for triangular and tetrahedral spectral elements, manuscript submitted for publication.