Journal of Computational Physics **171,** 394–417 (2001)
doi:10.1006/jcph.2001.6805, available online at http://www.idealibrary.com on IDEAL®

Low-Energy Basis Preconditioning for Elliptic Substructured Solvers Based on Unstructured Spectral/*hp* Element Discretization

Spencer J. Sherwin* and Mario Casarin†

**Imperial College of Science, Technology and Medicine, Prince Consort Road, London SW7 2BY, United Kingdom; and †Department of Mathematics, State University of Campinas, P.O. Box 6065, Campinas, SP-13081-970, Brazil*
E-mail: s.sherwin@ic.ac.uk; casarin@ime.unicamp.br

Received September 13, 2000; revised February 24, 2001

The development and application of three-dimensional unstructured hierarchical spectral/*hp* element algorithms has highlighted the need for efficient preconditioning for elliptic solvers. Building on the work of Bica (Ph.D. thesis, Courant Institute, New York University, 1997) we have developed an efficient preconditioning strategy for substructured solvers based on a transformation of the expansion basis to a low-energy basis. In this numerically derived basis the strong coupling between expansion modes in the original basis is reduced thus making it amenable to block diagonal preconditioning. The efficiency of the algorithm is maintained by developing the new basis on a symmetric reference element and ignoring, in the preconditioning step, the role of the Jacobian of the mapping from the reference to the global element. By applying an additive Schwarz block preconditioner to the low-energy basis combined with a coarse space linear vertex solver we have observed reductions in execution time of up to three times for tetrahedral elements and 10 times for prismatic elements when compared to a standard diagonal preconditioner. Full details of the implementation and validation of the tetrahedral and prismatic element preconditioning strategy are set out below.

1. INTRODUCTION

The development of unstructured solvers based on the spectral/*hp* element method has permitted a broader range of challenging problems to be addressed using automated mesh generation. In the context of computational fluid dynamics these methods provide high spatial accuracy and good phase properties when the solution is smooth; these properties make them very suitable for incompressible flow problems.

We consider an *hp* discretization where '*h*' denotes the size of an elemental subdomain and '*p*' represents the polynomial order within each element. This discretization has been

0021-9991/01 $35.00

applied to the incompressible Navier–Stokes equations in three dimensions using a splitting approach [2]. After we apply a time discretization which decouples the viscous and inviscid parts of the operator, the most computationally intensive parts of the solver are a series of elliptic solves, namely one Poisson solve and three Helmholtz solves, which are performed at each time step. Each of these elliptic solves is preconditioned with an iterative substructuring type domain decomposition method which takes advantage of the natural splitting of the basis into interior, face, edge, and vertex basis functions. Currently once a suitable computational mesh has been generated the limiting computational cost of the algorithm is the solution of the four elliptic problems.

As noted by several authors, special care must be taken in three dimensions in order to produce a method which is scalable with respect to h and p. In this paper we consider a preconditioning approach similar to that adopted by Mandel [3] for structured hexahedral domains and investigated for unstructured domains by Bica [1]. In the current investigation we develop a set of numerically evaluated low-energy basis functions calculated by solving a local elliptic problem on a regular reference element. Unlike the approach of Bica, the low-energy basis functions are then applied to the global elements that constitute the solution mesh. The resulting method is quasi-optimal in terms of iteration count and is computationally efficient due to its local construction. The method is then extended to treat hybrid domains of tetrahedral and prismatic elements. We have demonstrated by numerical experiments that for the solution of a standard Poisson equation the low-energy preconditioner significantly reduces the number of iteration counts as well as produces improvements of up to a factor of 10 in the computational solve time for typical polynomial orders.

In Section 2 we discuss the change of basis from an analytically defined expansion [4] to the new low-energy basis detailing the motivation behind the transformation and illustrate how to numerically implement the method. Section 3 then demonstrates some numerical tests on both regular and irregular regions. In Section 4 we provide a summary of our conclusions.

2. FORMULATION

We consider the elliptic boundary value problem

$$\nabla^2 u(x, y, z) + \lambda u(x, y, z) = f(x, y, z) \tag{1}$$

supplemented with appropriate Neumann or Dirichlet boundary conditions. As is standard practice in the spectral/*hp* element method approach [5] we approximate our solution as a piecewise polynomial $u^\delta(\boldsymbol{x})$ using the Galerkin discretization of Eq. (1); i.e., find $u^\delta \in V^\delta$ such that

$$\mathcal{L}(v, u) = \int_\Omega \nabla v^\delta \cdot \nabla u^\delta + \lambda v^\delta u^\delta \, d\boldsymbol{x} = \int_\Omega v^\delta f \, d\boldsymbol{x} \quad \forall v^\delta \in V^\delta. \tag{2}$$

The solution domain Ω is assumed to be tessellated into N_{el} nonoverlapping elemental regions Ω_e of polymorphic shapes consisting of hexagons, prisms, pyramids, and tetrahedrons such that

$$\bigcup_{e=1}^{e=N_{el}} \bar{\Omega}_e = \bar{\Omega}.$$

Each of the elemental regions is mapped to a standard region within which we construct a polynomial expansion.

The choice of the expansion base is relevant to the following discussion. Accordingly it is worth discussing the different approaches currently adopted in the spectral element and p-type finite element communities. The traditional approach within the spectral element community [6–8] has been to adopt a nodal tensorial expansion basis within structured subdomains (i.e., quadrilaterals or hexahedral elements). These bases are typically constructed from Lagrange polynomials through Gauss–Lobatto–Legendre quadrature points. Within the p-type finite element community the typical approach for structured subdomains has been to adopt a modal or hierarchical tensorial expansion basis. Typically these expansion bases, in one dimension, use the standard (nonhierarchical) linear, finite element modes which are then supplemented with hierarchical polynomial functions based on the integral of Legendre polynomials [9–11] or equivalently Jacobi polynomials with weights of (1, 1) [4, 12].

For unstructured subdomains, such as triangles and tetrahedrons, the use of a standard tensor product of one-dimensional expansions leads to a basis whose dimension is larger than the polynomial space supported by the expansion. Therefore such an approach would lead to a redundant system (in terms of polynomial space) which can cause undesirable time-step restrictions when treating convectively dominated problems [13]. To generate a nonredundant system within unstructured subdomains the use of modal expansions has been preferred [4, 10, 12]. However, more recently nontensorial nodal bases for triangular regions have also been proposed [14–16].

As previously mentioned the following paper is based upon the doctoral work of Bica who considered a modal expansion for tetrahedral domains [1]. A similar approach was also proposed by Mandel for structured subdomains [3].

2.1. *Change of Basis*

The discussion in this section is independent of the particular polynomial basis being used. We note however that in the remainder of this work we have adopted the basis defined in [5, 17, 18]; see Section 2.2.

Let Φ_1 and Φ_2 represent two complete bases in the discrete space, V^δ. We introduce the two Helmholtz matrices $\boldsymbol{H}_1, \boldsymbol{H}_2$ such that $\boldsymbol{H}_1[i, j] = \mathcal{L}(\varphi_{1i}, \varphi_{1j})$, where φ_{1i} and φ_{1j} are two arbitrary elements of basis Φ_1; $\boldsymbol{H}_2$ is defined analogously.

For $u^\delta \in V^\delta$, let $\hat{u}_{1i}$ and $\hat{u}_{2i}$ be the ith coefficients of u^δ when expanded in terms of the two bases; i.e.,

$$u^\delta(\boldsymbol{x}) = \sum_{i=1}^{\dim(V^\delta)} \hat{u}_{1i}\varphi_{1i}(\boldsymbol{x}) = \sum_{j=1}^{\dim(V^\delta)} \hat{u}_{2j}\varphi_{2j}(\boldsymbol{x}). \tag{3}$$

Since we have chosen Φ_1, Φ_2 both to lie in the same discrete space V^δ we can represent one basis in terms of the other

$$\varphi_{2j}(\boldsymbol{x}) = \sum_{i=1}^{\dim(V^\delta)} \hat{\varphi}_{ji}\varphi_{1i}(\boldsymbol{x}), \tag{4}$$

where $\hat{\varphi}_{ji}$ is the ith coefficient corresponding to the expansion mode $\varphi_{1i}(\boldsymbol{x})$ which when summed over i is identical to the expansion basis $\varphi_{2j}(\boldsymbol{x})$.

Defining $\boldsymbol{\varphi_1}(\boldsymbol{x})$ as the vector of bases functions (i.e., $\boldsymbol{\varphi_1}[i] = \varphi_{1i}(\boldsymbol{x})$) and similarly $\hat{\boldsymbol{\varphi}}_{\boldsymbol{j}}$ as a vector of expansion coefficients (i.e., $\hat{\boldsymbol{\varphi}}_{\boldsymbol{j}}[i] = \hat{\varphi}_{ji}$) then Eq. (4) can be written in vector form as

$$\varphi_{2j}(\boldsymbol{x}) = [\hat{\boldsymbol{\varphi}}_{\boldsymbol{j}}]^T \boldsymbol{\varphi_1}(\boldsymbol{x}).$$

We now introduce a matrix $\boldsymbol{C}$ whose rows are the vectors $[\hat{\boldsymbol{\varphi}}_{\boldsymbol{j}}]^T$. We note that this matrix transforms from basis Φ_1 to Φ_2 since

$$\boldsymbol{\varphi_2}(\boldsymbol{x}) = \boldsymbol{C}\boldsymbol{\varphi_1}(\boldsymbol{x}), \tag{5}$$

where $\boldsymbol{\varphi_2}(\boldsymbol{x})$ is defined analogously to $\boldsymbol{\varphi_1}(\boldsymbol{x})$. Similarly the matrix $\boldsymbol{C}^{-1}$ transforms the basis Φ_2 to Φ_1.

If we now write Eq. (3) in vector form to obtain

$$\boldsymbol{u} = [\hat{\boldsymbol{u}}_{\boldsymbol{1}}]^T \boldsymbol{\varphi_1}(\boldsymbol{x}) = [\hat{\boldsymbol{u}}_{\boldsymbol{2}}]^T \boldsymbol{\varphi_2}(\boldsymbol{x}), \tag{6}$$

where $\hat{\boldsymbol{u}}_{\boldsymbol{1}}[i] = \hat{u}_{1i}$, $\hat{\boldsymbol{u}}_{\boldsymbol{2}}[i] = \hat{u}_{2i}$ then substitute Eq. (5) into Eq. (6) we obtain

$$[\hat{\boldsymbol{u}}_{\boldsymbol{1}}]^T \boldsymbol{\varphi_1}(\boldsymbol{x}) = [\hat{\boldsymbol{u}}_{\boldsymbol{2}}]^T \boldsymbol{C}\boldsymbol{\varphi_1}(\boldsymbol{x})$$

and so

$$\hat{\boldsymbol{u}}_{\boldsymbol{1}} = \boldsymbol{C}^T \hat{\boldsymbol{u}}_{\boldsymbol{2}}. \tag{7}$$

The matrix $\boldsymbol{C}^T$ can therefore be interpreted as a transformation from the expansion coefficients $\hat{\boldsymbol{u}}_{\boldsymbol{2}}$ to $\hat{\boldsymbol{u}}_{\boldsymbol{1}}$.

Finally recalling the bilinear energy as $\mathcal{L}(\cdot, \cdot)$,

$$\mathcal{L}(u, u) = \int_\Omega \nabla u \cdot \nabla u + \lambda u_2 \, d\boldsymbol{x}$$

we can express $\mathcal{L}$ in terms of the Helmholtz matrices as

$$\mathcal{L}(u, u) = [\hat{\boldsymbol{u}}_{\boldsymbol{1}}]^T \boldsymbol{H}_1 \hat{\boldsymbol{u}}_{\boldsymbol{1}} = [\hat{\boldsymbol{u}}_{\boldsymbol{2}}]^T \boldsymbol{H}_2 \hat{\boldsymbol{u}}_{\boldsymbol{2}}.$$

Applying the transformation (7) we obtain

$$[\hat{\boldsymbol{u}}_{\boldsymbol{2}}]^T \boldsymbol{C}\boldsymbol{H}_1\boldsymbol{C}^T \hat{\boldsymbol{u}}_{\boldsymbol{2}} = [\hat{\boldsymbol{u}}_{\boldsymbol{2}}]^T \boldsymbol{H}_2 \hat{\boldsymbol{u}}_{\boldsymbol{2}},$$

and so $\boldsymbol{H}_2 = \boldsymbol{C}\boldsymbol{H}_1\boldsymbol{C}^T$ and equivalently $\boldsymbol{H}_1 = \boldsymbol{C}^{-1}\boldsymbol{H}_2\boldsymbol{C}^{-T}$.

Notice that Φ_1 and Φ_2 could be bases of two different spaces $V_2 \subset V_1$, where $V_2 \neq V_1$. The expression $\boldsymbol{H}_2 = \boldsymbol{C}\boldsymbol{H}_1\boldsymbol{C}^T$ would still be valid, but not $\boldsymbol{H}_1 = \boldsymbol{C}^{-1}\boldsymbol{H}_2\boldsymbol{C}^{-T}$, for rank deficiency reasons.

At this juncture it can be appreciated that although an initial choice of basis Φ_1 may lead to an undesirable Helmholtz matrix $\boldsymbol{H}_1$, we can obtain a more suitable Helmholtz matrix $\boldsymbol{H}_2$ by an appropriate choice of the matrix $\boldsymbol{C}$ or equivalently by transforming the basis. The suitability of the matrix is related to the ease at which it can be preconditioned. From the point of view of implementation, a block diagonal preconditioner can be an efficient

choice. We are therefore motivated to define $\boldsymbol{C}$ so that it results in a Helmholtz matrix $\boldsymbol{H}_2$ that is as near as possible to being block diagonal. In our case, we follow the work of Dryja *et al*. [19]; these authors have already observed, in the *h*-version of the finite element, that the theory points to a deterioration of the performance of the block diagonal preconditioner when applied to a problem in three dimensions. They propose adding a diagonal part to the preconditioner, which acts on functions with small energy associated to the vertices. This idea has been implemented for modal expansions in [20]. In this paper we adapt those ideas and those of Bica [1] to the basis used in the $\mathcal{N}\varepsilon\kappa\mathcal{T}\alpha\boldsymbol{r}$ code.

We are left with the question of how to determine the matrix $\boldsymbol{C}$. For convenience of implementation the choice of our original basis Φ_1 is usually based on an analytic definition of a polynomial set, in a mapped region. However, it is unlikely that an optimal choice of $\boldsymbol{C}$ can be expressed in terms of another simple closed form polynomial basis. We therefore accept that the new basis Φ_2 needs to be generated numerically. The implementational efficiency of the spectral/*hp* element method is derived from the decomposition of the global expansion into elemental regions. It is essential that this property is maintained in our new expansion basis Φ_2. The details of the numerical implementation will be discussed in Section 2.4.1.

2.2. *Decomposition of the Expansion Basis and Substructuring*

Figure 1 illustrates the modal decomposition adopted in the current work. We assume the solution domain is decomposed into contiguous nonoverlapping elemental domains which are tetrahedrons in Fig. 1a. Within each elemental domain the solution is then expanded in terms of a complete polynomial expansion of arbitrary order. The construction of a suitable expansion basis is essential to the efficient implementation of the numerical methods; see Section 2.1. As we are interested in solving second-order, partial differential equations when the elemental polynomial expansions are assembled into a global approximation we

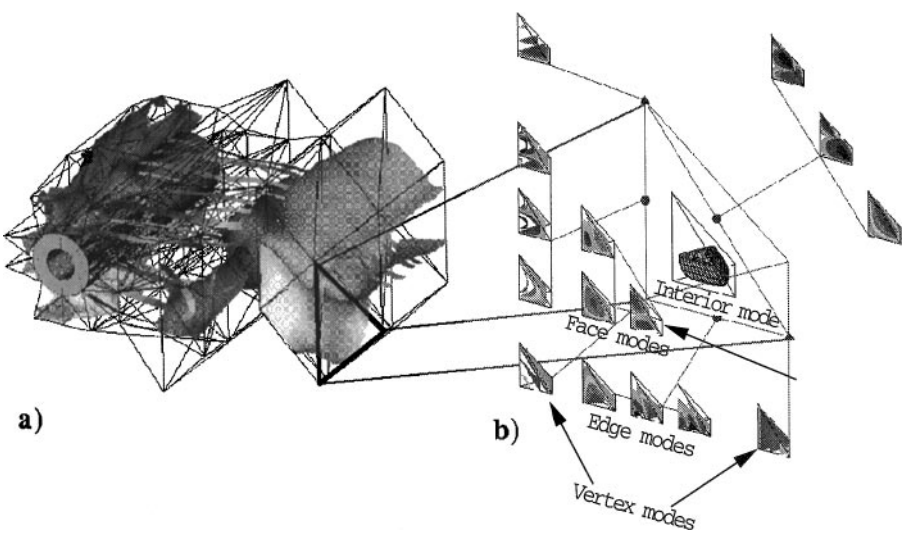


FIG. 1. Illustration of the modal decomposition: (a) decomposition of the solution domain into tetrahedral elements, (b) representative modes from a third-order polynomial expansion indicating vertex, edge, face, and interior modes.

require the resulting piecewise polynomial function to be in H^1. This can be guaranteed by using a C^0-continuous expansion, which is obtained by decomposing the expansion basis into interior and boundary modes, and matching the boundary modes across the interface of the elements.

The interior modes are defined as having support within the interior of the elemental region. The boundary modes are the remaining modes required to ensure a complete expansion basis. The boundary modes are decomposed into three different components comprising vertex, edge, and face modes. Face modes are defined as having support on a single face and the interior of the elemental domain and vanishing on all other faces, edges, and vertices. Edge modes are defined as having support on a single edge, the adjacent faces, and the interior of the elemental domain. An edge mode is therefore zero on all the other edges, nonadjacent faces, and at the vertices of the elemental domain. Finally, vertex modes are defined as having support at a single vertex and on the surrounding edges and faces, and the interior of the element. A vertex mode is therefore zero on all nonadjacent edges and faces as well as the remaining vertices. The combination of vertex and edge modes are collectively known as the wirebasket frame. Under the above definition it does not matter whether the expansion basis is modal or nodal although our motivation is to derive a good preconditioner for the modal expansion basis. For a full definition of the modal expansion basis adopted in this work see [4, 5, 17].

Having defined the boundary and interior modes we can use the property of the expansion basis to decompose our linear algebra problem using a manipulation known as static condensation or substructuring. The discrete form of our Galerkin problem (2) can be written in matrix form as

$$\boldsymbol{H}\hat{\boldsymbol{u}} = \boldsymbol{f},$$

where $\boldsymbol{H}$ is the Helmholtz matrix, $\hat{\boldsymbol{u}}$ is the vector of expansion coefficients of the polynomial approximation to the solution, and $\boldsymbol{f}$ is the inner product of the forcing function $f(\boldsymbol{x})$ with the expansion modes modified to incorporate the boundary conditions. If we now decompose the $\boldsymbol{H}\hat{\boldsymbol{u}}$ and $\boldsymbol{f}$ into contributions associated with the boundary and interior modes we obtain

$$\begin{bmatrix} \boldsymbol{H}_{bb} & \boldsymbol{H}_{bi} \\ \boldsymbol{H}_{ib} & \boldsymbol{H}_{ii} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{u}}_b \\ \hat{\boldsymbol{u}}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_b \\ \boldsymbol{f}_i \end{bmatrix}, \tag{8}$$

where the subscripts b and i refer to the boundary and interior degrees of freedom, respectively. The boundary and interior degrees of freedom can be solved in a decoupled manner if we statically condense the system by premultiplying Eq. (8) by

$$\begin{bmatrix} \boldsymbol{I} & -\boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1} \\ 0 & \boldsymbol{I} \end{bmatrix},$$

arriving at

$$\begin{bmatrix} \boldsymbol{H}_{bb} - \boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1}\boldsymbol{H}_{ib} & 0 \\ \boldsymbol{H}_{ib} & \boldsymbol{H}_{ii} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{u}}_b \\ \hat{\boldsymbol{u}}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_b - \boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1}\boldsymbol{f}_i \\ \boldsymbol{f}_i \end{bmatrix}. \tag{9}$$

This technique, otherwise known as substructuring, allows us initially to solve the top row of Eq. (9) for the boundary degrees of freedom. Having determined the boundary degrees of

freedom $\hat{\boldsymbol{u}}_b$ we can use the second row of Eq. (9) to solve for the interior degrees of freedom $\hat{\boldsymbol{u}}_i$. From the point of view of linear algebra it is not evident that such a manipulation is advantageous. However, if we recall the definition of the interior degrees of freedom, which vanish on the elemental domain boundaries and within all other elemental domains, we can conclude that the matrix $\boldsymbol{H}_{ii}$ is block diagonal. Therefore this matrix is easily inverted. Accordingly, the substructuring has reduced our problem to solving the boundary problems

$$\boldsymbol{S}\hat{\boldsymbol{u}}_b = [\boldsymbol{H}_{bb} - \boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1}\boldsymbol{H}_{ib}]\hat{\boldsymbol{u}}_b = \boldsymbol{f}_b - \boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1}\boldsymbol{f}_i,$$

where $\boldsymbol{S} = \boldsymbol{H}_{bb} - \boldsymbol{H}_{bi}[\boldsymbol{H}_{ii}]^{-1}\boldsymbol{H}_{ib}$ is known as the Schur complement matrix.

The decoupling of the interior blocks also has the advantage that the Schur complement matrix can be constructed at an elemental level. We can therefore iteratively solve the Schur complement system and it is this system that we wish to precondition. For the Helmholtz problem (1) the matrix $\boldsymbol{H}$ is positive definite provided $\lambda > 0$. Since the Schur complement is the stiffness matrix associated with a subspace of the space generated by the original basis, its condition number is bounded by the condition number of the full matrix $\boldsymbol{H}$ and is typically far better.

A disadvantage of this approach is the additional expense of constructing the Schur complement matrices. For a single matrix solve this can outweigh the advantages of solving a better conditioned system. However, for our problem of interest, we wish to solve the matrix system repeatedly as part of an unsteady solution to the Navier–Stokes equation. The computational cost of constructing and storing the Schur complement matrix is therefore outweighed by the number of solves. We remark that an alternative but equivalent interpretation of the substructuring approach is that we have numerically changed the basis so that the boundary degrees of freedom are orthogonal to the interior degrees of freedom in the energy inner product $\mathcal{L}(\cdot,\cdot)$.

We now wish to restrict our attention to preconditioning only the Schur complement system since the interior degrees of freedom are dealt with by a direct solver such as Cholesky factorization. If we define our transformation matrix $\boldsymbol{C}$ as

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{R} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix}$$

and statically condense the matrix system $\boldsymbol{H}_2 = \boldsymbol{C}\boldsymbol{H}_1\boldsymbol{C}^T$ we find that the Schur complement of $\boldsymbol{H}_2$ is related to the Schur complement of $\boldsymbol{H}_1$ by

$$\boldsymbol{S}_2 = \boldsymbol{R}\boldsymbol{S}_1\boldsymbol{R}^T,$$

where $\boldsymbol{S}_1, \boldsymbol{S}_2$ are the Schur complements of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$, respectively. Our task is now to determine an appropriate choice of $\boldsymbol{R}$.

2.3. *Study of the Couplings between the Blocks of the Helmholtz Matrix*

For a two-dimensional modal expansion, block diagonal preconditioning of the Schur complement system leads to a preconditioned system with a condition number that grows polylogarithmically with polynomial order [3, 21, 22]. This preconditioner uses the complete block corresponding to all vertex modes and the diagonal blocks corresponding to the modes along each individual edge. However, an analogous approach in three dimensions, where

the preconditioner is constructed from the blocks of vertex, edge, and face modes, does not produce favorable results. Even considering the complete wirebasket space of all vertex and edge modes as a single block still does not produce a very effective preconditioner. As we will see this is because the couplings between the face and the wirebasket parts of the Schur complement matrices are relatively strong.

In order to illustrate this point, we study the preconditioning of a single elemental tetrahedral region as previously studied by Bica [1]. A similar approach was also adopted by Babuška *et al.* for the quadrilateral element [23]. We consider the Helmholtz problem (1) with $\lambda = 0$ and construct the Schur complement matrix for a single reference element $\{-1 \le x, y, z; x + y + z \le -1\}$. To ensure that we do not have a singular problem we apply Dirichlet boundary conditions to all vertex modes. We denote the Schur complement with Dirichlet boundary conditions by

$$\boldsymbol{S}_{DV} = \begin{bmatrix} \boldsymbol{S}_{ww} & \boldsymbol{S}_{wf} \\ \boldsymbol{S}_{fw} & \boldsymbol{S}_{ff} \end{bmatrix},$$

where $\boldsymbol{S}_{ww}$ denotes all the couplings between the wirebasket modes which just contain the edges for this problem. $\boldsymbol{S}_{ff}$ contains all the couplings between the face modes and $\boldsymbol{S}_{wf} = \boldsymbol{S}_{fw}^T$ represents the couplings between the wirebasket and face modes.

The first test we consider is the growth with polynomial order of the condition number of the matrix

$$\boldsymbol{S}_{ww+ff}^{-1} \boldsymbol{S}_{DV},$$

where $\boldsymbol{S}_{ww+ff}$

$$\boldsymbol{S}_{ww+ff} = \begin{bmatrix} \boldsymbol{S}_{ww} & 0 \\ 0 & \boldsymbol{S}_{ff} \end{bmatrix}.$$

This test illustrates the relative importance of the coupling between the wirebasket and faces.

In the next test we consider a similar approach to determine the strength of the coupling between the four faces. If we let $\boldsymbol{S}_{f_1 f_1}, \boldsymbol{S}_{f_2 f_2}, \boldsymbol{S}_{f_3 f_3}, \boldsymbol{S}_{f_4 f_4}$ be the block diagonals of $\boldsymbol{S}_{DV}$, which correspond to the modes on each of the four faces, we can find the condition number of

$$\boldsymbol{S}_{fb}^{-1} \boldsymbol{S}_{ff},$$

where

$$\boldsymbol{S}_{fb} = \begin{bmatrix} \boldsymbol{S}_{f_1 f_1} & 0 & 0 & 0 \\ 0 & \boldsymbol{S}_{f_2 f_2} & 0 & 0 \\ 0 & 0 & \boldsymbol{S}_{f_3 f_3} & 0 \\ 0 & 0 & 0 & \boldsymbol{S}_{f_4 f_4} \end{bmatrix}.$$

Similarly, to determine the strength of the coupling between the edges we denote by $\boldsymbol{S}_{e_i e_i}$ a diagonal block of $\boldsymbol{S}_{DV}$ which corresponds to the modes within edge e_i and compute the

condition number of

$$\boldsymbol{S}_{wb}^{-1}\boldsymbol{S}_{ww},$$

where

$$\boldsymbol{S}_{wb} = \begin{bmatrix} \boldsymbol{S}_{e_1e_1} & 0 & 0 & 0 \\ 0 & \boldsymbol{S}_{e_2e_2} & 0 & 0 \\ 0 & 0 & \boldsymbol{S}_{e_3e_3} & 0 \\ 0 & 0 & 0 & \boldsymbol{S}_{e_4e_4} \end{bmatrix}.$$

A final test is to consider the full block diagonal decoupling, i.e.,

$$\boldsymbol{S}_{wb+fb}^{-1}\boldsymbol{S}_{DV},$$

where

$$\boldsymbol{S}_{wb+fb} = \begin{bmatrix} \boldsymbol{S}_{wb} & 0 \\ 0 & \boldsymbol{S}_{fb} \end{bmatrix}.$$

The growth rate of the L_2 condition number of $\boldsymbol{S}_{ww+ff}^{-1}\boldsymbol{S}_{DV}$ and $\boldsymbol{S}_{wb+fb}^{-1}\boldsymbol{S}_{DV}$ as a function of polynomial order is shown in Fig. 2 and the condition number of $\boldsymbol{S}_{wb}^{-1}\boldsymbol{S}_{ww}$ and $\boldsymbol{S}_{fb}^{-1}\boldsymbol{S}_{ff}$ are shown in Fig. 3. The condition number was evaluated as the ratio of the maximum and minimum eigenvalue which were computed using LAPACK [24]. From these figures it is evident that the strongest coupling is between the wirebasket and face modes and has an approximate asymptotic growth rate of $O(P^{2.5})$ as shown in Fig. 2.

In contrast the block diagonal preconditioning of the wirebasket and face modes is relatively well behaved and the numerical tests indicate a potentially sublinear growth rate.

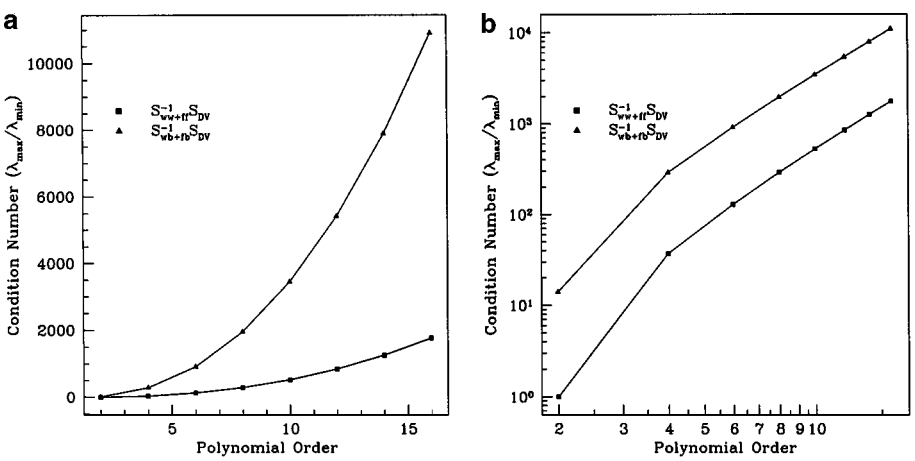


FIG. 2. Condition number growth of $\boldsymbol{S}_{ww+ff}^{-1}\boldsymbol{S}_{DV}$ and $\boldsymbol{S}_{wb+fb}^{-1}\boldsymbol{S}_{DV}$ versus polynomial order. (a) Lin–lin axis and (b) log–log axis.

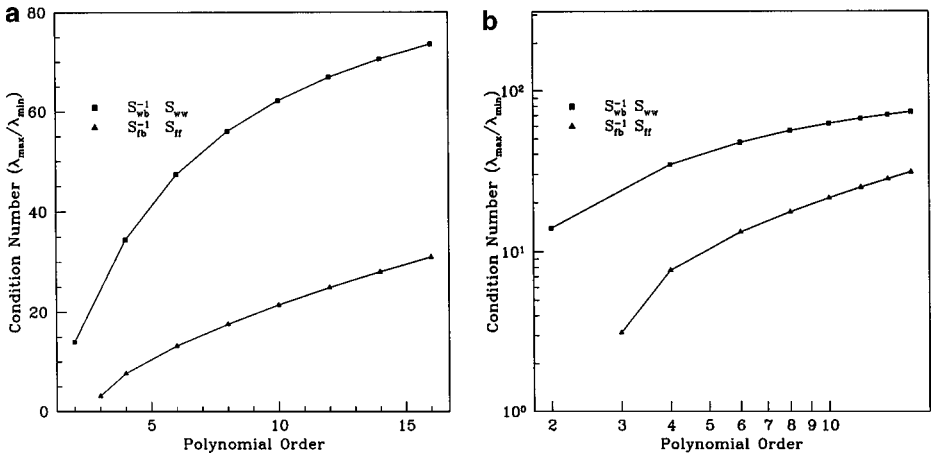


FIG. 3. Condition number growth of $\boldsymbol{S}_{wb}^{-1}\boldsymbol{S}_{ww}$ and $\boldsymbol{S}_{fb}^{-1}\boldsymbol{S}_{ff}$ versus polynomial order. (a) Lin–lin axis and (b) log–log axis.

Clearly the objective of the change of basis will be to weaken the coupling between the wirebasket and face modes and this is the focus of Section 2.4.

Finally we note that these experiments are dependent upon the expansion basis; the results shown in Figs. 2 and 3 have been obtained using the basis defined in the $\mathcal{N}\varepsilon\kappa\mathcal{T}\alpha\boldsymbol{r}$ code [5, 17].

2.4. *Construction of the Low-Energy Basis*

We recall that

$$\boldsymbol{S}_2 = \boldsymbol{R}\boldsymbol{S}_1\boldsymbol{R}^T.$$

Adopting the notation introduced in Sections 2.1 and 2.2, and considering the experiments referred to in Section 2.3, our goal is to find a basis Φ_2 spanning the same space as our initial basis Φ_1, which decouples the face mode contribution from the wirebasket modes. This decoupling would make $\boldsymbol{S}_2$ block diagonal, however, and to avoid a very expensive transformation we impose the additional requirement that only the edge and vertex functions are modified in the change of basis.

Consider a single elemental matrix and the transformation of basis which arises from a matrix $\boldsymbol{R}$ of the form

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{R}_{ve} & \boldsymbol{R}_{vf} \\ 0 & \boldsymbol{I} & \boldsymbol{R}_{ef} \\ 0 & 0 & \boldsymbol{I} \end{bmatrix},$$

where we have assumed that vertex modes are listed first followed by the edge and then the face modes. The matrices $\boldsymbol{R}_{ve}, \boldsymbol{R}_{vf}$ represent the modification of the vertex modes by the edges and face modes. Similarly the matrix $\boldsymbol{R}_{ef}$ represents the modification of the edge modes by the face modes. We note that this matrix has a very straightforward inverse due

to the upper triangular form of the matrix which is of the form

$$\boldsymbol{R}^{-1} = \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{R}_{ve} & -\boldsymbol{R}_{vf} + \boldsymbol{R}_{ve}\boldsymbol{R}_{ef} \\ 0 & \boldsymbol{I} & -\boldsymbol{R}_{ef} \\ 0 & 0 & \boldsymbol{I} \end{bmatrix}.$$

In the work of Bica [1] and Mandel [3] the submatrix $\boldsymbol{R}_{ve}$ was set to zero. For descriptive convenience we define

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{R}_v \\ 0 & \boldsymbol{A} \end{bmatrix},$$

where

$$\boldsymbol{R}_v = [\, \boldsymbol{R}_{ve} \quad \boldsymbol{R}_{vf} \,], \quad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{R}_{ef} \\ 0 & \boldsymbol{I} \end{bmatrix}.$$

Now if we denote the original Schur complement of the Helmholtz matrix as

$$\boldsymbol{S}_1 = \begin{bmatrix} \boldsymbol{S}_{vv} & \boldsymbol{S}_{v,ef} \\ \boldsymbol{S}_{v,ef}^T & \boldsymbol{S}_{ef,ef} \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}_{vv} & \boldsymbol{S}_{ve} & \boldsymbol{S}_{vf} \\ \boldsymbol{S}_{ve}^T & \boldsymbol{S}_{ee} & \boldsymbol{S}_{ef} \\ \boldsymbol{S}_{vf}^T & \boldsymbol{S}_{ef}^T & \boldsymbol{S}_{ff} \end{bmatrix}$$

then under the change of basis $\boldsymbol{S}_2 = \boldsymbol{R}\boldsymbol{S}_1\boldsymbol{R}^T$ we obtain

$$\boldsymbol{S}_2 = \begin{bmatrix} \boldsymbol{S}_{vv} + \boldsymbol{R}_v\boldsymbol{S}_{v,ef} + \boldsymbol{S}_{v,ef}\boldsymbol{R}_v^T + \boldsymbol{R}_v\boldsymbol{S}_{ef,ef}\boldsymbol{R}_v^T & \left[\boldsymbol{S}_{v,ef} + \boldsymbol{R}_v^T\boldsymbol{S}_{ef,ef}\right]\boldsymbol{A}^T \\ \boldsymbol{A}\left[\boldsymbol{S}_{v,ef} + \boldsymbol{S}_{ef,ef}\boldsymbol{R}_v^T\right] & \boldsymbol{A}\boldsymbol{S}_{ef,ef}\boldsymbol{A}^T \end{bmatrix}, \tag{10}$$

where

$$\boldsymbol{A}\boldsymbol{S}_{ef,ef}\boldsymbol{A}^T = \begin{bmatrix} \boldsymbol{S}_{ee} + \boldsymbol{R}_{vf}\boldsymbol{S}_{ef} + \boldsymbol{S}_{ef}\boldsymbol{R}_{vf}^T + \boldsymbol{R}_{vf}\boldsymbol{S}_{ff}\boldsymbol{R}_{vf}^T & \boldsymbol{S}_{ef} + \boldsymbol{R}_{vf}^T\boldsymbol{S}_{ff} \\ \boldsymbol{S}_{ef} + \boldsymbol{S}_{ff}\boldsymbol{R}_{vf}^T & \boldsymbol{S}_{ff} \end{bmatrix}. \tag{11}$$

If we consider the coupling between the vertex modes with the edge and face modes shown in Eq. (10) we see that to completely orthogonalize these modes we require that

$$\boldsymbol{S}_{v,ef}^T + \boldsymbol{S}_{ef,ef}\boldsymbol{R}_v^T = 0$$

or

$$\boldsymbol{R}_v^T = -\boldsymbol{S}_{ef,ef}^{-1}\boldsymbol{S}_{v,ef}^T. \tag{12}$$

Similarly to decouple the edge modes from the face modes we see from inspecting Eq. (11) that

$$\boldsymbol{R}_{ef}^T = -\boldsymbol{S}_{ff}^{-1}\boldsymbol{S}_{ef}^T. \tag{13}$$

We note that on an elemental region, the use of Eq. (13) corresponds to the use of edge modes with the same traces as the original ones, modified in the interior of the faces to have

the lowest possible energy within the polynomial space. However, decoupling the basis using Eqs. (12) and (13) to form $\boldsymbol{R}$ would destroy the boundary decomposition of the vertex and edges modes. We recall from Section 2.2 that the edge modes are defined as having support along a single edge and the two adjacent faces. The edge modes are therefore zero at all vertices and along all other edges and faces. The use of $\boldsymbol{S}_{ff}^{-1}$ in the Eq. (11) would create a new basis where an edge mode would have support along all faces. This is not a problem when treating a single element. The decomposition of the boundary modes is, however, important when generating a piecewise continuous global expansion from an elemental region. If we alter the support of the boundary modes, it becomes considerably harder to generate a global expansion from the elemental definitions. Apart from the issue of global assembly a change of basis of this form would also be extremely difficult to implement for a general mesh. This complexity arises from the fact that there can be a high multiplicity of face modes around any specific edge and so assembling the local matrices $\boldsymbol{S}_{ff}^{-1}$ would not be significantly easier than directly inverting the problem.

To overcome these problems we need to reconsider the most appropriate decomposition for construction of our new basis. Reviewing the form of the edge modes we find that we can decouple a specific edge only from its two adjacent faces if we do not wish to alter its support. This type of local decoupling, however, does not circumvent the problem of multiple faces being adjacent to any edge in a global mesh. Nevertheless if we consider the decoupling of the local faces from the edges within a standard symmetrical region we can then apply the new, local basis to all global elements. This does, however, require that the shape of the new edge and vertex modes within adjacent faces maintains the same rotational symmetry as the original basis. This can be ensured by using a rotationally symmetric standard region provided the operator under consideration is isotropic. Using a standard region to construct the basis means that we do not take account of the Jacobian of the mapping between the global element and the standard region. Accordingly the edges in the global mesh will not be completely decoupled from the surrounding elements. However, as we shall demonstrate from numerical tests, the new low-energy basis generates a Schur complement matrix which can be spectrally approximated by its block diagonal contribution. Using the block diagonal as a preconditioner therefore leads to a preconditioned system with a favorable condition number which grows polylogarithmically with the polynomial order P.

Following a similar argument for the vertex modes we can locally decouple each vertex mode from the edges and faces which are adjacent to the vertex without destroying the assembly properties of the expansion. In other words, we orthogonalize the edge and vertex modes with respect to the face modes using an inner product based on the geometry of the standard region and disregarding the Jacobian of the mapping. The new functions will not be exactly orthogonal in the inner product induced by the global Helmholtz matrix, but it will retain enough orthogonality for the new basis to be useful for block diagonal preconditioning. The remainder of this section will therefore focus on the construction of the locally decoupled edge and vertex modes.

2.4.1. Numerical construction of the low-energy transformation matrix $\boldsymbol{R}$. Perhaps the most abstract aspect of the formulation so far is the determination of the transformation matrix $\boldsymbol{R}$ in a numerical implementation. To illustrate this process we start from the assumption that we have already calculated the local Schur complement matrix $\boldsymbol{S}_1$ (see [5] for further details on this construction). As before we assume that this matrix is ordered so that the vertex modes are listed before the edge modes which are followed by the face modes.

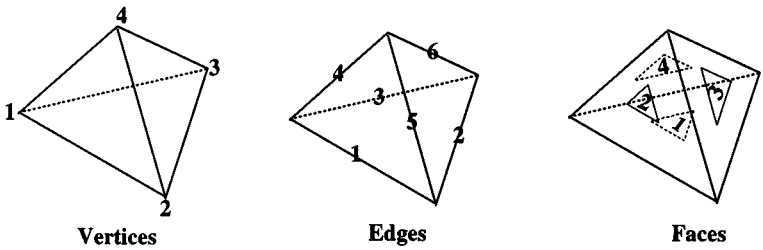


FIG. 4. Numbering of vertices, edges, and faces within a standard region of an equilateral tetrahedron.

We introduce an edge and face numbering scheme for the standard rotationally symmetric region of an equilateral tetrahedron as shown in Fig. 4. The construction of the submatrix $\boldsymbol{R}_{ef}^T$ to orthgonalize the local edge 2 from faces 1 and 3 is illustrated in Fig. 5. Initially we extract the submatrices from $\boldsymbol{S}_{ff}$ corresponding to inner products of modes on faces 1 and 3 within themselves as well as the coupling matrix between the two faces. This matrix of face coupling is then inverted and multiplied by the submatrices of $\boldsymbol{S}_{ef}^T$ which correspond to the coupling between edge 2 and faces 1 and 3. The nonsquare resultant matrix then forms a submatrix of transformation matrix $\boldsymbol{R}_{ef}^T$. Performing the same operation for all six edges and their corresponding adjacent faces leads to the full construction of $\boldsymbol{R}_{ef}^T$ and involves only relatively straightforward operations once the matrix $\boldsymbol{S}_1$ has been generated.

The assembly of the components $\boldsymbol{R}_v^T$ of the transformation matrix as schematically illustrated in Fig. 6, follows a similar construction. In this figure we have considered the construction of the submatrices corresponding to the coupling between vertex 1 with edges 1, 2, 4 and faces 1, 2, 4. For every vertex mode we have to construct and invert the submatrix corresponding to the coupling between edges 1, 2, 4 and faces 1, 2, 4 and then multiply this by the submatrix representing the coupling between vertex 1 and the adjacent edges and faces.

One way of interpreting the low-energy edge modes is as the solution to the Schur complement problem within the symmetric standard region. The boundary conditions to the problem are zero Dirichlet boundary conditions on the nonadjacent vertices, faces, and edges and a unit Dirichlet boundary condition on the mode under consideration. The solution

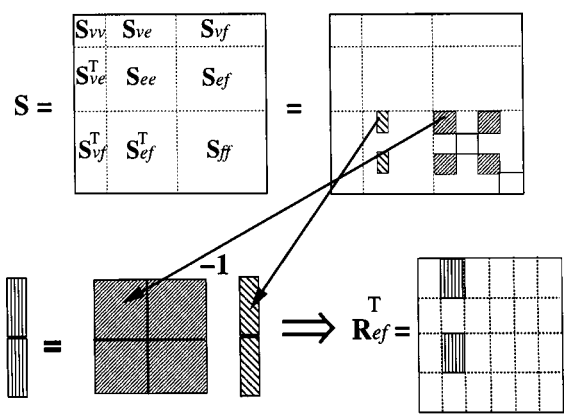


FIG. 5. Schematic representation of the construction of a submatrix of $\boldsymbol{R}_{ef}^T$ corresponding to the coupling of edge 2 with faces 1 and 3 as defined in Fig. 4.

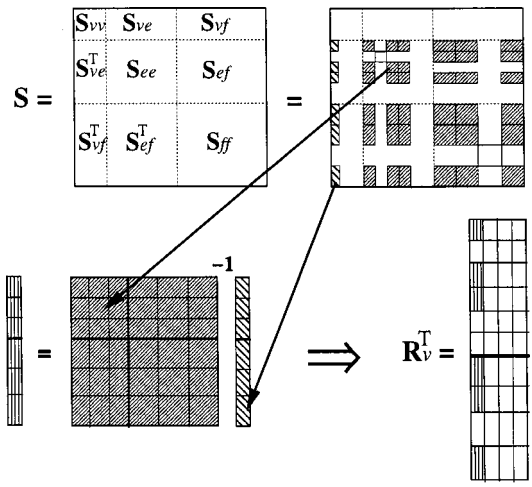


FIG. 6. Schematic representation of the construction of a submatrix of $\boldsymbol{R}_{ve}^T$ and $\boldsymbol{R}_{vf}^T$ corresponding to the coupling of vertex 1 with edges 1, 3, 4 and faces 1, 2, and 3 as defined in Fig. 4.

to the problem or equivalently the shape of two low-energy modes is shown in Fig. 7. In Fig. 7a, zero Dirichlet boundary conditions have been imposed on face 1, edges 1, 2, and 3 and vertices 1, 2, and 3 while a unit Dirichlet condition has been imposed on vertex 4. The change in shape should be compared with the original projected vertex mode shown in Fig. 7b. In Fig. 8 we show the scatter plot of the magnitude of the Schur complement of the original expansion basis $\boldsymbol{S}_1$ and the low-energy basis $\boldsymbol{S}_2 = \boldsymbol{R}\boldsymbol{S}_1\boldsymbol{R}^T$ plotted on the same scale. From this plot we see that the energy of the vertex modes in the original basis is noticeably higher than that of the rest of the diagonal contribution. Another strong contribution, however, exists between the edge modes and vertex modes as denoted by the energetic contributions at the edge of the plot. In contrast the low-energy basis in Fig. 8b, which has been scaled by a factor of 4, is far more diagonally dominated. We note that the diagonal face contribution is the same in both matrices. Both of these examples were evaluated at a polynomial order of $P = 5$.

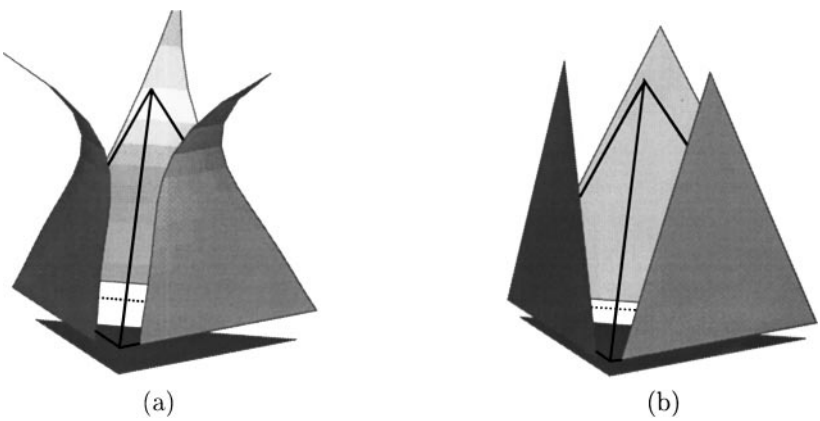


FIG. 7. Projected mode shape of vertex 4 (a) low-energy and (b) original basis. The polynomial order was $P = 5$.

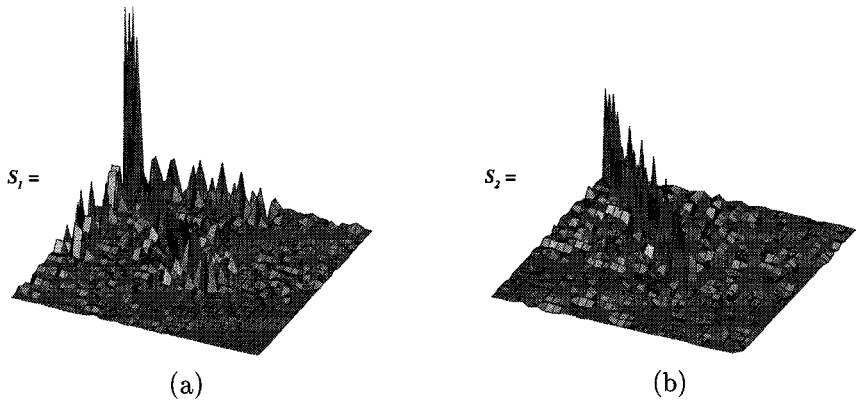


FIG. 8. Scatter plot of Schur complement matrices of a $P = 5$ polynomial expansion: (a) Original basis and (b) low-energy basis (scaled by a factor of 4).

2.5. *Extension to Hybrid Subdomains*

Although tetrahedral subdomains permit a greater flexibility in automatically generating meshes, there are regions, such as the boundary layer of a viscous flow, where other geometric shapes are more attractive. If we use prismatic subdomains in the boundary layer where the triangular faces of the prisms touch the surface of the body the flexibility of automated surface mesh generation is maintained but better resolution in the surface normal direction is provided. We therefore require low-energy bases for hybrid elemental regions consisting of tetrahedrons, prisms, pyramids, and hexahedrons for which conforming hierarchical bases already exist [17, 18].

For an elemental region of any tetrahedral, pyramidic, prismatic, or hexahedral shape the boundary transformation matrix $\boldsymbol{R}$ can be constructed in a way similar to the technique described for tetrahedral elements in Section 2.4. The analytically defined boundary modes applied within hybrid subdomains [17, 18] have similar modal shapes along edges and faces. This is necessary to enforce C^0 continuity in the global expansion with minimal effort. We have seen that when developing the transformation matrix $\boldsymbol{R}$ for the tetrahedron, the choice of a rotationally symmetric standard region is sufficient to ensure that the low-energy expansion basis maintains shape similarity between edges and faces. For example, the shape of the low-energy mode on vertex 1 will have a shape similar to the low-energy mode on vertex 2 along all edges. Accordingly, these vertices in different elements can be assembled together.

When developing the low-energy transformation for the prismatic and pyramidic elemental domains a rotationally symmetric standard domain is clearly not possible. For the prismatic element we adopt a standard region with two equilateral triangles connected by edges of the same length as shown in Fig. 9. Developing the transformation matrix as discussed in Section 2.4 does not, however, guarantee that the shape of the transformed vertex mode along the edge of a triangular face will be similar to the shape of the transformed vertex mode along an edge which lies within a square face. Similarly we cannot guarantee that the shape of a low-energy edge mode within a triangular face of the prism will be similar to the shape of the low-energy edge mode within a triangular face of the tetrahedron. It is perhaps not surprising that forming independent transformation matrices for tetrahedral and prismatic regions leads to low-energy bases which do not have similar edge and face shapes.

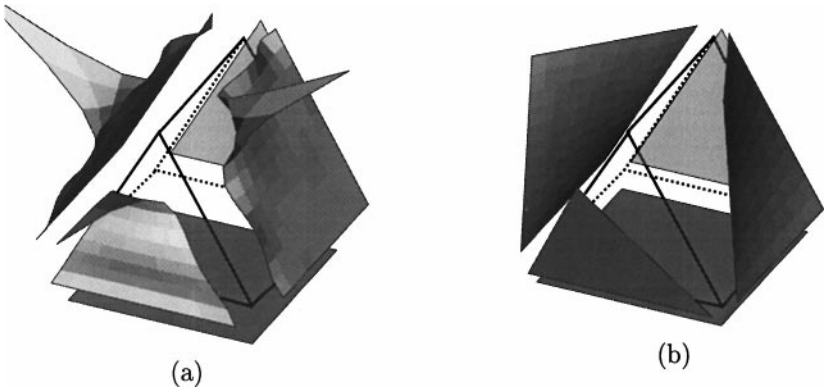


FIG. 9. Projected mode shapes within the standard prismatic region. The low-energy vertex function is shown in (a) and can be compared with the original vertex basis shown in (b). The polynomial order was $P = 5$.

These bases therefore could not be assembled into a C^0 basis using the same connectivity as the original analytic basis.

The similarity of the low-energy basis therefore needs to be preserved. Recognizing that the original basis does have shape similarity we can maintain this similarity between different elemental regions by using the information from the tetrahedral low-energy transformation within the prismatic transformation. We therefore replace the edge and triangular face components of the prismatic vertex transformation matrices ($\boldsymbol{R}_{ve}$ & $\boldsymbol{R}_{vf}$) with the corresponding components from the tetrahedral transformation matrices. We also modify all the triangular face components in the prismatic edge transformation matrix $\boldsymbol{R}_{ef}$ with the corresponding component from the tetrahedral transformation matrix. These operations require us to identify edges and faces with similar local coordinates in the tetrahedral and prismatic regions. Since, by design, the edge and face components of the analytic basis has similar shapes this operation ensures that the low-energy basis also have similar shape symmetries. Under this modification the only remaining components of the original prismatic low-energy basis are those corresponding to the quadrilateral faces of the prism. We note that these components are also not rotationally symmetric but the restrictions on prism orientation within the computational meshes considered means that this lack of symmetry does not cause any problems. A more general transformation matrix can be constructed from a family of transformation matrices based upon the rotationally symmetric tetrahedral and hexahedral elements.

In Fig. 9 we see the low-energy vertex modes shapes for the analytic and unmodified low-energy prismatic basis. Clearly, modifying the low-energy prismatic basis to impose continuity when using hybrid domains introduces another suboptimality operation. This adds to the fact that we have ignored the mapping from a general element to the standard region. In Section 3 we shall, however, demonstrate that it is still possible to achieve favorable numerical conditioning.

2.6. *Construction of the Preconditioner*

To complete our discussion of the low-energy preconditioner we describe how the additive Schwarz preconditioner is adopted in this work.

The additive Schwarz preconditioner $\boldsymbol{S}_{prec}$ can be defined as

$$\boldsymbol{S}_{prec} = \Pi(\boldsymbol{S}_1)_{vv}^{-1}\Pi^T + \boldsymbol{R}^T \begin{bmatrix} \mathrm{Diag}[(\boldsymbol{S}_2)_{vv}] & 0 & 0 \\ 0 & (\boldsymbol{S}_2)_{eb} & 0 \\ 0 & 0 & (\boldsymbol{S}_2)_{fb} \end{bmatrix}^{-1} \boldsymbol{R},$$

where Π is the vertex restriction operator, $\mathrm{Diag}[(\boldsymbol{S}_2)_{vv}]$ is the diagonal of the $\boldsymbol{S}_2$ vertex modes, $(\boldsymbol{S}_2)_{eb}$ is the block diagonal of the edge components and $(\boldsymbol{S}_2)_{fb}$ is the block diagonal of the face components.

The initial component of the preconditioner requires the assembly and inversion of the piecewise linear vertex block $(\boldsymbol{S}_1)_{vv}$. This is readily available since $\boldsymbol{S}_1$ must be generated before constructing $\boldsymbol{S}_2$. The elemental transformation matrix $\boldsymbol{R}$ is then generated as discussed in Sections 2.4 and 2.5 from which we can generate the elemental contributions $(\boldsymbol{S}_2)_{eb}$ and $(\boldsymbol{S}_2)_{fb}$.

We note that the construction of the low-energy basis discussed in Section 2.4 does not ensure that the space of the wirebasket functions contains the space of constants. This is undesirable from the point of view of preconditioning because of the dependence of the condition number on the number of elements in the whole mesh. Nevertheless, since the space of constants is already included in the space of original vertex functions, represented by $(\boldsymbol{S}_1)_{vv}$, the algorithm is scalable with respect to the number of elements.

3. RESULTS

3.1. *Model Tests*

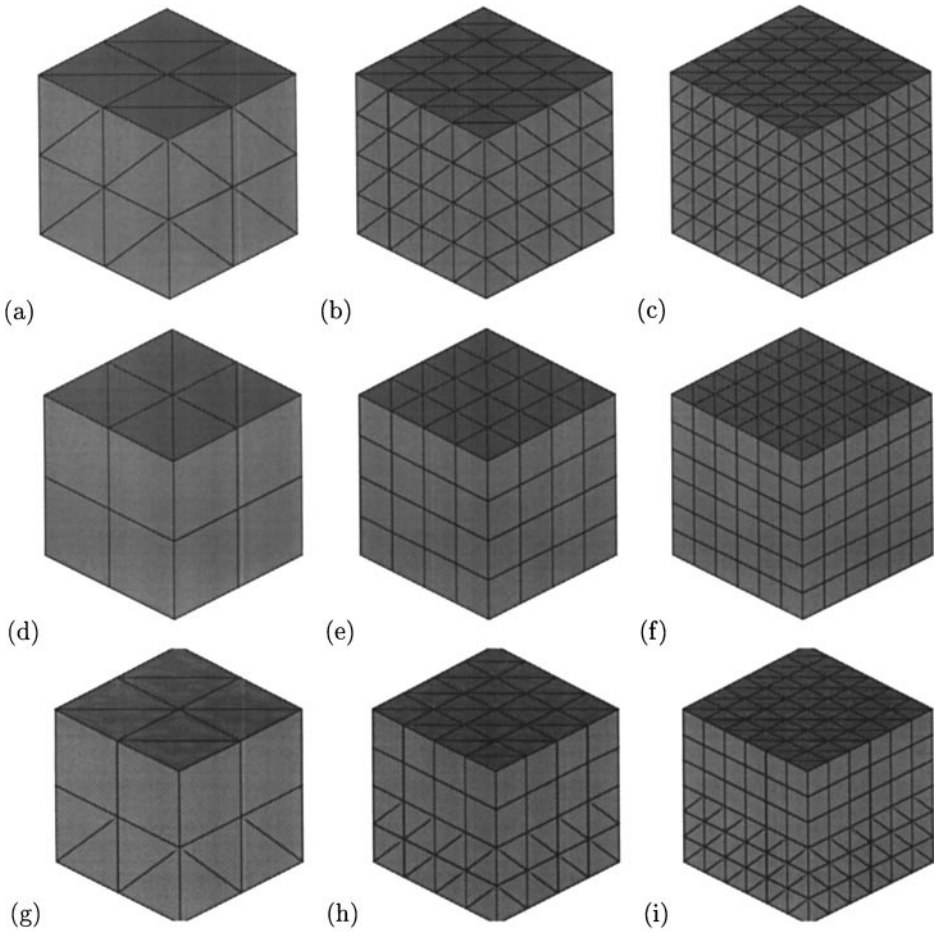
In the first series of tests we consider uniform refinement in terms of h and p on model computational regions. We have considered the region $-1 \leq x, y, z \leq 1$ and subdivided the region into two, four, and six similar cuboids. These have then been divided into six tetrahedrons or two prisms as shown in Fig. 10. We have also considered a hybrid region of tetrahedrons and prisms as shown in Fig. 10g–i. We note that the global degrees of freedom are identical in the tetrahedral, prismatic, and hybrid meshes for a given polynomial order if the domains contain a similar number of cuboid regions. However, the number of boundary degrees of freedom and therefore the rank of the Schur complement system is not similar. The rank is approximately 50% larger in the full tetrahedral mesh as compared with the full prismatic mesh at higher polynomial orders.

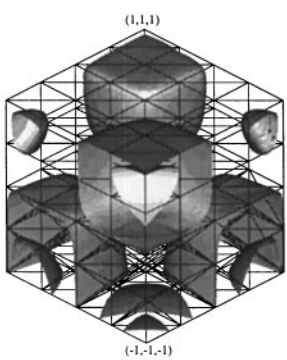
We have chosen to solve the Dirichlet Poisson equation with an analytic solution of the form

$$u(x, y, z) = xyz\left(1 - e^{-10(1-x)}\right)\left(1 - e^{-10(1-y)}\right)\left(1 - e^{-10(1-z)}\right).$$

Isocontours of the solution is shown in Fig. 10j for the $4 \times 4 \times 4$ cube domains split into tetrahedral, prismatic, and hybrid elements. Finally, the convergence for the tetrahedral, prism, and hybrid element regions is shown in Fig. 10k on a semilog axis, where we see that an exponential rate of convergence is achieved on all meshes. The polynomial order for this convergence test and all subsequent tests in this section ranged over $2 \leq p \leq 8$, which is the current practical range used in our simulations.

For each of the computational domains shown in Fig. 10 we have considered the conditioning of the low-energy preconditioned system in terms of polynomial order and uniform





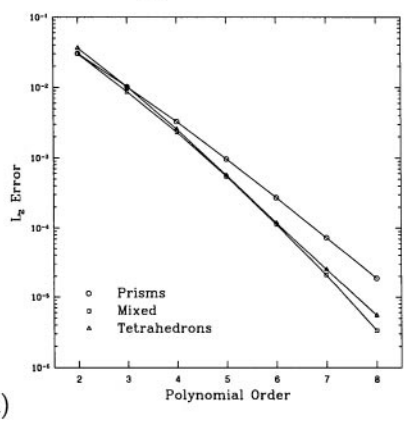


FIG. 10. Computational meshes. Tetrahedral domains: (a) $N_{el} = 48$ (b) $N_{el} = 384$, and (c) $N_{el} = 1296$. Prismatic domains: (d) $N_{el} = 16$, (e) $N_{el} = 128$, and (f) $N_{el} = 432$. Hybrid domains (g) $N_{el} = 32$, (h) $N_{el} = 256$, and (i) $N_{el} = 864$. (j) Isocontours of solution. (k) Convergence of L_2 error with respect to polynomial order P.

mesh refinement. Each of the following tests were performed using the Lanczos technique with a standard preconditioned conjugate gradient solver [25] which generates a tridiagonal matrix with a spectral distribution similar to the preconditioned system. The eigenvalues of the tridiagonal matrix were then determined using a LAPACK routine. For low polynomial

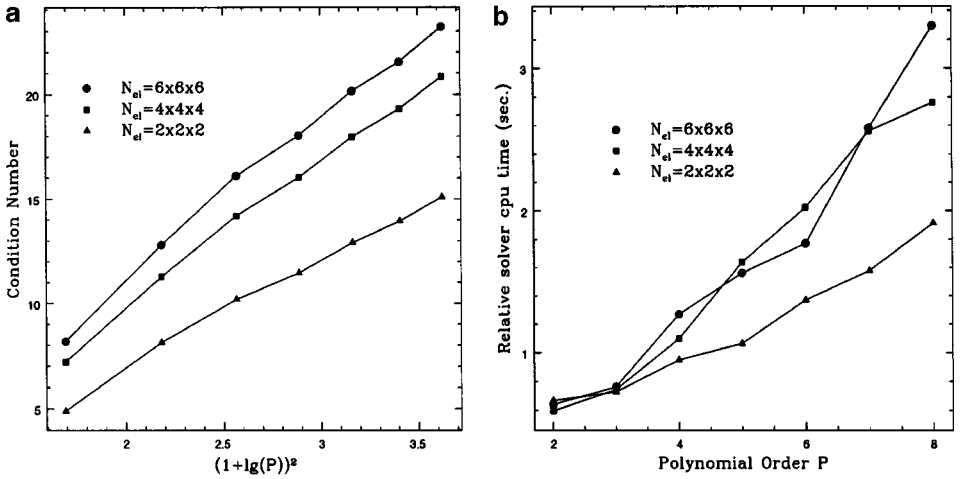


FIG. 11. (a) Scaling of the tetrahedral mesh condition number as a function of the square of the logarithm of polynomial order. (b) Ratio of solver time using low-energy preconditioning versus diagonal preconditioning.

orders on small meshes a minimum of 50 iterations were performed. The routine was also validated against the full eigenvalue evaluation using a QR algorithm in LAPACK. In all other tests the solution was iterated until the L_2 residual was 10^{10} times smaller than the vector on the right-hand side. All tests were performed on a dedicated SGI R10000 195 MHz processor.

In Fig. 11a we see the L_2 condition number of the low-energy preconditioned Schur complement matrix as a function of the polylogarithmic scaling $(1 + lg(P)^2)$. From the work of Pavarino and Widlund [20] we know that a polylogarithmic scaling of this form is possible in a substructured solver in three-dimensional spectral elements (i.e., using a Lagrange basis). Further from the work of Bica [1] we would also expect a polylogarithmic scaling of this form since the role of the mapping, not considered in our low-energy basis, is unlikely to play a significant role in such a regular domain. The polynomial fit to the condition number of the diagonally preconditioned system based on the three highest polynomial orders was $O(P^{3.3})$. Figure 11a also demonstrates that the uniform mesh refinement increase in condition number for a fixed polynomial order is very slow. We expect it be asymptotically independent of the mesh size h.

Also shown in Fig. 11b is a comparison of the relative CPU time to solve a diagonally preconditioned system versus a low-energy preconditioned system where the CPU time does not include matrix setup costs. The setup costs have not been considered as we are interested in unsteady fluid dynamics solvers, where a solve can be called thousands of times during a single simulation making the setup cost negligible for practical polynomial orders. From Fig. 11b we observe that the low-energy basis breaks even at a polynomial order of $P = 4$. By a polynomial order of $P = 8$ the low-energy basis solve is three times as fast as the diagonally preconditioned solver. This saving is a direct consequence of the lower iteration count as a result of the significantly improved scaling of the condition number. It should also be noted that part of the efficiency of the preconditioner results from the addition of the coarse linear space preconditioning and not only the low-energy basis.

Figures 12a and 12b demonstrate similar tests for the unmodified low-energy prismatic regions where shape symmetry has not been enforced along the edges. In Fig. 12a we again

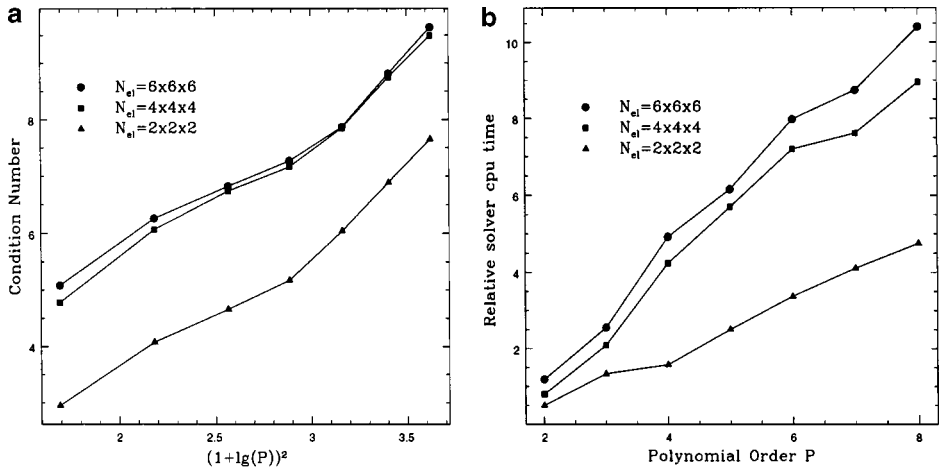


FIG. 12. (a) Scaling of the prismatic mesh condition number as a function of the square of the logarithm of polynomial order. (b) Ratio of solver time using low-energy preconditioning versus diagonal preconditioning.

observe an asymptotic polylogarithmic trend in the conditioning of the low-energy system as a function of polynomial order. The condition number at $P = 8$ is significantly lower than the tetrahedral case although the rank of the matrix is also lower for a given polynomial order. Further, when the mesh is refined from $4 \times 4 \times 4$ cuboid blocks to $6 \times 6 \times 6$ cuboid blocks the *h*-scaling is independent of the mesh size. For this case the condition number diagonally preconditioned system scaled as $O(P^{3.2})$ is based on the three highest polynomial orders. Considering the CPU speedup of the low-energy preconditioning as compared with the diagonal preconditioning for this case shown in Fig. 12b we see that the low-energy basis breaks even at a polynomial order of $P = 3$ and demonstrates a factor of 10 speedup at $P = 8$.

Finally in Figs. 13a and 13b we apply the same test to the hybrid mesh of tetrahedral and prismatic elements. For this test the prismatic low-energy transformation matrix

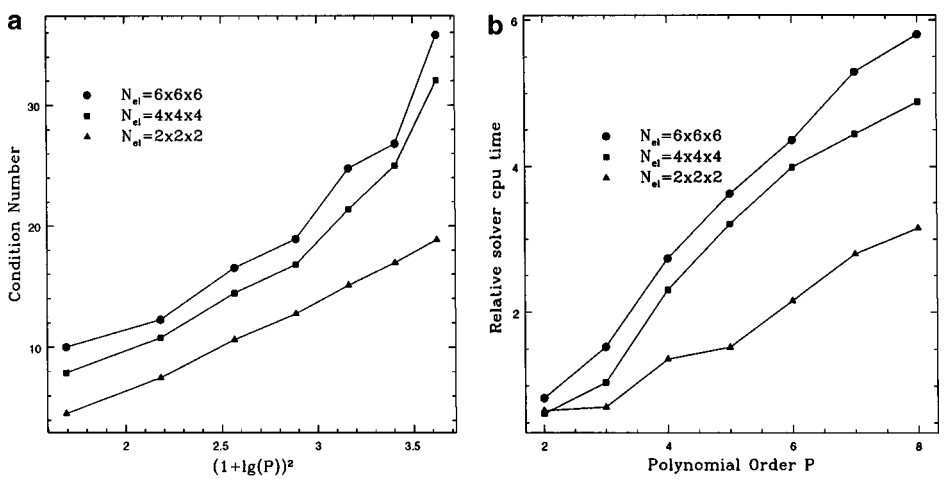


FIG. 13. (a) Scaling of the hybrid mesh condition number as a function of the square of the logarithm of polynomial order. (b) Ratio of solver time using low-energy preconditioning versus diagonal preconditioning.

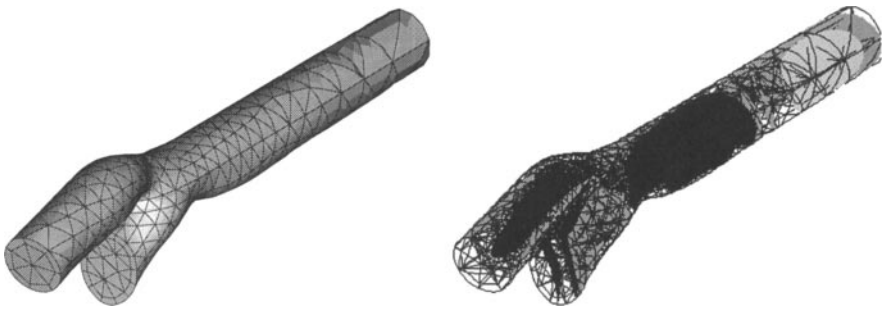


FIG. 14. Geometrically complex hybrid domain of a distal arterial bypass graft. The domain is constructed from a prismatic boundary layer region adjacent to the surface within which an unstructured tetrahedral mesh is constructed. Also shown is the solution to a Poisson equation $u(x, y, z) = \sin(x)\sin(y)\sin(z)$.

was modified to maintain edge and face similarity with the tetrahedral basis. Unlike the unmodified test cases shown in Figs. 11 and 12 the condition number only appears to follow a polylogarithmic scaling of the form $(1 + lg(P)^2)$ in the $2 \times 2 \times 2$ cuboid mesh. At finer mesh resolution the condition number grows at a faster rate and the absolute values of the condition number exceed the previous tetrahedral and prismatic tests for all values of P. However, Fig. 13b demonstrates that the speedup of the low-energy preconditioner over the diagonal preconditioner still breaks even at a polynomial order of $P = 4$ and even shows a sixfold speedup at a polynomial order of $P = 8$ on the finest mesh.

3.2. *Geometrically Complex Computational Domains*

To finish our results section we shall consider a geometrically complex hybrid computational domain of practical interest as shown in Fig. 14. This figure illustrates the computational reconstruction of a porcine arterial bypass graft at the downstream, or distal, end of the graft. The domain consists of an unstructured triangular surface discretization from which prismatic elements are constructed by extruding the triangular surface elements in the surface normal direction. The interior region is then discretized using tetrahedral subdomains. The discretization shown in Fig. 14 consists of 749 prismatic and 1720 tetrahedral elements.

In our first test we again consider a Dirichlet Poisson equation with the solution $u(x, y, z) = \sin(x)\sin(y)\sin(z)$. This is also shown in Fig. 14. The Lanczos technique was applied to determine the condition number of the diagonal and low-energy preconditioned systems. The results of this are shown in Fig. 15a. From Fig. 15a we see that the diagonal and low-energy preconditioned systems scale approximately as $O(P^3)$ and $O(P)$ respectively in this polynomial range. The departure from the polylogarithmic scaling of the low-energy preconditioner observed in the previous tests is presumably due to the Jacobian of the mapping between the local and global element which was ignored in our low-energy transformation. The improvement in condition number is also reflected in the speedup of the back solve of the low-energy preconditioner over the diagonal preconditioner as shown in Fig. 15b.

The tests were performed on eight processors of a SGI origin 2000 system and eight processors of a PC Pentium III 500 MHz Beowulf system. From Fig. 15b we observe that a speedup of approximately 6 was achieved on both systems at a polynomial order of $P = 8$ and the breakeven polynomial order was approximately $P = 3$.

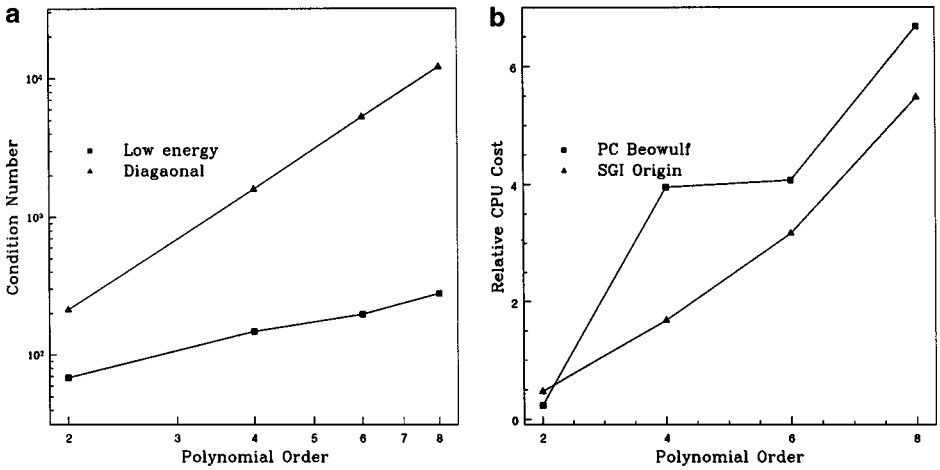


FIG. 15. (a) Growth of the condition number of a diagonal and low-energy preconditioned problem as a function of polynomial order on the computational domain shown in Fig. 14. (b) Relative CPU cost of the diagonal preconditioner versus the low-energy preconditioner.

As previously discussed the motivation behind the development of the low-energy basis was for application in a unsteady incompressible Navier–Stokes solver using a high-order splitting scheme [2]. This algorithm requires the solution of three Helmholtz equations and one Poisson equation. Once again considering the computational domain shown in Fig. 14 we compared the CPU time for 20 time steps of the solver with the low-energy and diagonal preconditioning starting with zero initial conditions. These tests were performed on 16 processors of a Fujitsu AP3000. From Table I we see that at a polynomial order of $P = 3$ that the low-energy preconditioner drops the average number of iterations to 30 from 157 iterations using a diagonal preconditioner. However, at this low polynomial order the speedup is only 1.7. Nevertheless as we increase the polynomial order to $P = 7$ the average number of iterations drops from 324 to 43 with an associated speedup of 5.3 as the average time per step drops from 174 to 32 s. The tolerance for these tests was set at 1×10^{-8} and the timings included all other operations necessary for the time integration of the Navier–Stokes solver using the splitting scheme [2].

TABLE I

Average Iteration Count and CPU Time over 20 Times Steps for the Diagonal and Low-Energy Preconditioners Applied to the High-Order Splitting Scheme for the Solution to the Navier–Stokes Equations

	Diagonal preconditioner		Low-energy preconditioner	
Poly order	Avg. iter.	Avg. CPU time	Avg. iter	Avg. CPU time
3	157	5.38	30	3.19
4	194	13.16	32	5.6
5	237	40.37	33	10.22
6	278	99.64	35	18.74
7	324	174.36	43	32.60

4. CONCLUSIONS

The transformation from an analytic unstructured expansion to a low-energy basis has produced an efficient preconditioner for hierarchical spectral/*hp* elements. The efficiency of the new basis is maintained by developing the transformation in a suitable, preferably rotationally symmetric, standard region. This approach is quasi-optimal since it ignores the Jacobian of the mapping to the physical elemental region. However, all of the computational properties of the expansion are developed at the elemental level maintaining the local nature of the algorithm. Further the connectivity information for the global assembly of the original expansion is maintained.

The low-energy basis is amenable to block diagonal preconditioning provided the linear subspace containing the space of constants is also included. Numerical tests performed in model computational domains illustrate that a condition number which is polylogarithmic with polynomial order can be achieved for tetrahedral and prismatic expansions. Modifications to the prismatic low-energy expansion, required to maintain the shape similarity of the two elements regions, leads to an increase in the asymptotic rate of the condition number scaling with polynomial order. Nevertheless a significant improvement in the conditioning and CPU cost as compared with diagonal preconditioned was observed for polynomial orders greater than $P > 4$.

Numerical validation in geometrically complex domains were also performed and demonstrated a similar improvement in conditioning and CPU times. Future improvements on the strategy should identify areas where the Jacobian of the local to global mapping becomes important.

ACKNOWLEDGMENTS

The authors acknowledge Rupad Darekar and Philipp Frauenfelder for the aid in generating the data. We also acknowledge the computational support from the Biomedical Visualisation Centre and the Imperial College Parallel Computing Centre. The first author also acknowledges partial support under the EPSRC Grant GR/L48096. The second author acknowledges partial support from CNPq—Brazil, under Project 300446/96-9.

REFERENCES

1. I. Bica, *Iterative Substructuring Algorithms for the p-version Finite Element Method for Elliptic Problems*, Ph.D. thesis (Courant Institute, New York University, 1997).
2. G. E. Karniadakis, M. Israeli, and S. A. Orszag, High-order splitting methods for incompressible Navier–Stokes equations, *J. Comput. Phys.* **97**, 414 (1991).
3. J. Mandel, Two-level domain decomposition preconditioning for the *p*-version finite element method in three-dimensions, *Int. J. Numer. Meth. Eng.* **29**, 1095 (1990).
4. S. J. Sherwin and G. E. Karniadakis, A new triangular and tetrahedral basis for high-order finite element methods, *Int. J. Numer. Meth. Eng.* **38**, 3775 (1995).
5. G. E. Karniadakis and S. J. Sherwin, *Spectral/hp Element Methods for CFD* (Oxford Univ. Press, London, 1999).
6. A. T. Patera, A spectral method for fluid dynamics: Laminar flow in a channel expansion, *J. Comput. Phys.* **54**, 468 (1984).
7. E. Rønquist, *Optimal Spectral Element Methods for the Unsteady Three-Dimensional Incompressible Navier–Stokes Equations*, Ph.D. thesis (Massachusetts Institute of Technology, 1988).
8. C. Canuto and A. Quarteroni, Approximation results for orthogonal polynomials in Sobolov spaces, *Math. Comput.* **38**, 55 (1982).

9. A. G. Peano, Hierarchies of conforming finite elements for plate elasticity and plate bending, *Comput. Math. Appl.* **2**, 211 (1976).
10. B. Szabo and I. Babuška, *Finite Element Analysis* (Wiley, New York, 1991).
11. J. T. Oden, *Optimal hp-Finite Element Methods*, Technical Report TICOM Report 92-09 (University of Texas at Austin, 1992).
12. M. Dubiner, Spectral methods on triangles and other domains, *J. Sci. Comput.* **6**, 345 (1991).
13. S. J. Sherwin and G. E. Karniadakis, A triangular spectral element method; applications to the incompressible Navier–Stokes equations, *Comput. Meth. Appl. Mech. Eng.* **123**, 189 (1995).
14. J. S. Hesthaven, From electostatics to almost optimal nodal sets for polynomial interpolation in a simplex, *SIAM J. Numer. Anal.* **35**, 655 (1998).
15. B. A. Wingate and M. A. Taylor, *The Koornwinder Polynomials Are the Eigenfunctions of a Sturm-Liouville Problem in the Simplex*, submitted for publication.
16. T. Warburton, L. Pavarino, and J. S. Hesthaven, A pseudo-spectral scheme for the incompressible Navier–Stokes equations using unstructured nodal elements, *J. Comput. Phys.* **164**, 1 (2001).
17. S. J. Sherwin, Hierarchical *hp* finite elements in hybrid domains, *Finite Elem. Anal. Des.* **27**, 109 (1997).
18. T. C. E. Warburton, *Spectral/hp Methods on Polymorphic Multi-Domains: Algorithms and Applications*, Ph.D. thesis (Brown University, 1998).
19. Maksymilian Dryja, Barry F. Smith, and Olof B. Widlund, Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions, *SIAM J. Numer. Anal.* **31**(6), 1662 (1994).
20. Luca F. Pavarino and Olof B. Widlund, A polylogarithmic bound for an iterative substructuring method for spectral elements in three dimensions, *SIAM J. Numer. Anal.* **33**(4), 1303 (1996).
21. I. Babuška, A. Craig, J. Mandel, and J. Pitkäranta, Efficient preconditioning for the *p*-version finite element method in two-dimensions, *SIAM J. Numer. Anal.* **28**(3), 624 (1991).
22. Mario A. Casarin, Quasi-optimal Schwarz methods for the conforming spectral element discretization, *SIAM J. Numer. Anal.* **34**(6), 2482 (1997).
23. I. Babuška, M. Griebel, and J. Pitkäranta, The problem of selecting the shape functions for a *p*-type finite element, *Int. J. Numer. Meth. Eng.* **28**, 1891 (1989).
24. E. Anderson, A. Bai, C. Bischof, J. Dennel, J. Dongarra, J. Du Croz, A. Greebaum, S. Hammarling, A. Mckenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide* (SIAM, Philadelphia, 1992).
25. G. H. Golub and C. F. van Loan, *Matrix Computations*, 2nd ed. (Johns Hopkins Press, Baltimore, 1989).