An overlapping Schwarz method for spectral element simulation of three-dimensional incompressible flows

P. F. Fischer *
N. I. Miller
H. M. Tufo

November 21, 1997

Abstract

As the characteristic propagation speed is infinite for unsteady incompressible flows, solving for the pressure operator is the most computationally challenging phase of flow simulation. Efficient solution is dependent upon the availability of fast solvers for sparse linear systems. In this paper we develop an Schwarz preconditioner using overlapping subdomains for the pressure. These local subdomain problems are derived from tensor products of one-dimensional finite element discretizations and admit use of fast diagonalization methods based upon matrix-matrix products. In addition, we use a coarse grid projection operator applied directly to the pressure on the interior Gauss points whose solution is computed via a fast parallel direct solver. The combination of overlapping Schwarz preconditioning and coarse grid solver provides as much as a four-fold reduction in simulation time over previously employed methods based upon deflation for parallel solution of multi-million grid point flow problems.

1 Introduction

We consider the problems encountered in large-scale spectral element simulations of unsteady incompressible flows. For semi-implicit time discretization of the incompressible Navier-Stokes equations based upon operator splitting, the linear subproblem associated with the pressure/divergence-free constraint can become very ill-conditioned at elevated resolutions, and consequently tends to be the most expensive phase of the simulation when iterative solvers are employed. This problem can be exacerbated by the presence of high-aspect ratio elements or widely varying scales of resolution, both of which are frequently encountered in practice. Therefore, a robust parallel preconditioning strategy is required.

We present a preconditioner for the pressure problem which derives from a low-order finite element Laplacian (with appropriate boundary conditions) and is well suited for application to three-dimensional problems. The low-order operator defines a system to which additive overlapping Schwarz methods, as proposed by Dryja and Widlund (e.g.[11]), can be readily applied. The combination of spectral methods and finite element preconditioning was first proposed by Orszag [27] and has been studied extensively by Deville, Mund, and coworkers, e.g., [9, 10]. For the case of the discrete Laplacian, the combination of spectral methods, finite element preconditioning, and additive Schwarz methods has been investigated by Pahl [28],Pavarino and Widlund [30], and Casarin [5]. Rønquist [33] and Casarin

*Present address: Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: *pff@cfm.brown.edu*

[6] have studied iterative substructuring methods for spectral element solution of the fully-coupled steady Navier-Stokes equations. Rønquist also proposed a block-Jacobi/deflation based scheme applied to the consistent Poisson operator governing the pressure for the unsteady case [15, 32].

The present scheme is closely related to our earlier two-dimensional work in which local subdomain problems were based upon linear finite elements (FEM) [17]. Here we abandon the flexible unstructured FEM approach in favor of tensor-product forms for the local operators on the overlapping regions. which, when employing the fast diagonalization method (FDM), require $O(N^d)$ storage and $O(N^{d+1})$ work per solve [7, 8, 31]. Moreover, this formulation obviates the need to tetrahedralize the Gauss points in $\mathbb{R}^3$. Consequently, we have been able to extend our earlier work to three dimensions and, for several large runs, have found the method to yield a four-fold reduction in simulation time over our previous deflation based production code [15, 16, 32].

The outline of the paper is as follows. In Section 2, we review the spectral element formulation for the unsteady Navier-Stokes equations and derive the system governing the pressure. In Section 3, we examine the use of low-order discrete Laplacians as a basis for pressure preconditioners. In Section 4 we extend this to develop an efficient overlapping Schwarz method based upon the FDM. In Section 5 we discuss the coarse grid problem and our direct solver. We present numerical results comparing the new method with earlier solution techniques in Section 6 and close with conclusions in Section 7.

2 Navier-Stokes Discretization

As the nature of the pressure operator is quite different from discrete Laplacians based upon standard weighted residual techniques, we briefly review the temporal and spatial discretization for the spectral element method.

We consider solution of the incompressible Navier-Stokes equations in $\mathbb{R}^d$, $d = 2$ or 3:

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} &= -\nabla p + \tfrac{1}{Re}\nabla^2\mathbf{u} && \text{in } \Omega, \\ \nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega, \end{aligned}$$

where $\mathbf{u} = (u_1, \ldots, u_d)$ is the velocity vector, p the pressure, and $Re = \frac{UL}{\nu}$ the Reynolds number based on a characteristic velocity, length scale, and kinematic viscosity. We have associated initial and boundary conditions

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}^0(\mathbf{x}) \,, \quad \mathbf{u} = \mathbf{u}_v \text{ on } \partial\Omega_v \,, \quad \nabla u_i \cdot \hat{\mathbf{n}} = 0 \text{ on } \partial\Omega_o \,,$$

where $\hat{\mathbf{n}}$ is the outward pointing normal on the boundary, and subscripts v and o refer to boundary regions where either "velocity" or "outflow" boundary conditions are specified.

2.1 Temporal discretization

Temporal discretization is based upon a semi-implicit scheme in which the nonlinear convective terms are treated explicitly either via a third-order Adams-Bashforth scheme or via a stable characteristics-based scheme which allows for time step sizes exceeding standard Courant limited time step sizes [7, 24]. Such a splitting leads to an unsteady Stokes problem to be solved at each time step:

$$\begin{aligned} \mathcal{H}\,\mathbf{u}^n + \nabla p^n &= \mathbf{f}^n && \text{in } \Omega \,, \\ \nabla\cdot\mathbf{u}^n &= 0 && \text{in } \Omega \,. \end{aligned} \tag{1}$$

Here $\mathcal{H}$ is the Helmholtz operator, $\mathcal{H} = \left(-\frac{1}{Re}\nabla^2 + \frac{c_0}{\Delta t}\right)$, and c_0 is an order unity constant. The inhomogeneous term, $\mathbf{f}^n$, and c_0 are determined by the choice of the nonlinear treatment. For the following derivation we assume that $c_0 = 1$ and drop the superscript n in (1). We also assume, without loss of generality, that $\mathbf{u}_v \equiv \mathbf{0}$ on $\partial\Omega_D$.

2.2 Spatial discretization

The Stokes problem (1) can be recast in an equivalent variational form:

Find $\mathbf{u} \in X$, $p \in Y$ *such that:*

$$\begin{aligned} \frac{1}{Re}(\nabla\mathbf{u}, \nabla\mathbf{v}) + \frac{1}{\Delta t}(\mathbf{u}, \mathbf{v}) - (p, \nabla\cdot\mathbf{v}) &= (\mathbf{f}, \mathbf{v}) \quad \forall\, \mathbf{v} \in X \\ - (q, \nabla\cdot\mathbf{u}) &= 0 \qquad \forall\, q \in Y \end{aligned} \tag{2}$$

where

$$\forall\, \phi, \psi \in \mathsf{L}^2(\Omega)\,, \quad (\phi, \psi) \equiv \int_\Omega \phi(\mathbf{x})\psi(\mathbf{x})\, d\mathbf{x}\,. \tag{3}$$

The proper subspaces for $\mathbf{u}$, $\mathbf{v}$ and p, q are [21]:

$$\begin{aligned} X &= \{\mathbf{v} : v_i \in \mathsf{H}^1(\Omega),\, i = 1, \ldots, d\,,\ \mathbf{v} = 0 \quad \text{on } \partial\Omega_v\} \\ Y &= \mathsf{L}^2(\Omega)\,. \end{aligned}$$

Here $\mathsf{L}^2(\Omega)$ is the space of all functions which are square integrable over Ω and $\mathsf{H}^1(\Omega)$ is the space of all functions belonging to $\mathsf{L}^2(\Omega)$ whose first derivatives are also in $\mathsf{L}^2(\Omega)$.

Spatial discretization proceeds by restricting $\mathbf{u}$, $\mathbf{v}$, p and q to compatible finite dimensional velocity and pressure subspaces, $X^N \subset X$ and $Y^N \subset Y$, respectively, and using appropriate quadrature to approximate the inner products in (2):

Find $\mathbf{u} \in X^N$, $p \in Y^N$ *such that:*

$$\begin{aligned} \frac{1}{Re}(\nabla\mathbf{u}, \nabla\mathbf{v})_{GL} + \frac{1}{\Delta t}(\mathbf{u}, \mathbf{v})_{GL} - (p, \nabla\cdot\mathbf{v})_G &= (\mathbf{f}, \mathbf{v})_{GL} \quad \forall\, \mathbf{v} \in X^N \\ - (q, \nabla\cdot\mathbf{u})_G &= 0 \qquad \forall\, q \in Y^N \end{aligned} \tag{4}$$

where the quadrature rules $(.,.)_{GL}$ and $(.,.)_G$ are related to the spaces X^N and Y^N.

In the spectral element method [29, 23] the bases for X^N and Y^N are defined by tessellating the domain into K non-overlapping subdomains, $\Omega = \cup_{k=1}^K \Omega^k$, and representing functions within each subdomain in terms of tensor-product polynomials on a reference subdomain $\hat{\Omega} = [-1, +1]^d$. (We will refer to the Ω^k's as *subdomains* to distinguish them from *elements* which will be defined in the context of finite element preconditioners in the next section.) Each Ω^k is the image of the reference subdomain under a mapping: $\mathbf{x}^k(\mathbf{r}) \in \Omega^k \Longrightarrow \mathbf{r} \in \hat{\Omega}$, with well defined inverse: $\mathbf{r}^k(\mathbf{x}) \in \hat{\Omega} \Longrightarrow \mathbf{x} \in \Omega^k$. Thus, each subdomain is a deformed quadrilateral in $\mathbb{R}^2$ or deformed parallelepiped in $\mathbb{R}^3$. The intersection of the closure of any two subdomains is either void, a vertex, an entire edge, or an entire face.

To avoid spurious pressure modes, Maday, Patera and Rønquist [25] and Bernardi and Maday [3] suggest the following approximation spaces for the velocity and pressure:

$$\begin{aligned} X^N &= X \cap \mathbb{P}^d_{N,K}(\Omega) \\ Y^N &= Y \cap \mathbb{P}_{N-2,K}(\Omega) \end{aligned} \tag{5}$$

where

$$\mathbb{P}_{\mathcal{N},K}(\Omega) = \left\{ v(\mathbf{x}^k(\mathbf{r}))\Big|_{\Omega^k} \in \mathbb{P}_{\mathcal{N}}(r_1) \otimes \ldots \otimes \mathbb{P}_{\mathcal{N}}(r_d),\, k = 1, \ldots, K \right\} \tag{6}$$

and $\mathbb{P}_{\mathcal{N}}(r)$ is the space of all polynomials of degree less than or equal to $\mathcal{N}$.

For the velocity space, we choose as a basis for $\mathbb{P}_N(r)$ the set of Lagrangian interpolants on the Gauss-Lobatto Legendre (GL) quadrature points in the reference domain: $\xi_i \in [-1,+1], i = 0,\ldots,N$. For the pressure space, the basis for $\mathbb{P}_{N-2}(r)$ is the set of Lagrangian interpolants on the Gauss Legendre (G) quadrature points $\eta_i \in]-1,+1[, i = 1,\ldots,N-1$. Fig 1 shows the nodal points for both the velocity (GL) and pressure (G) meshes for a regular subdomain configuration. Note that the basis for velocity is continuous across subdomain interfaces, while the basis for the pressure is not.

The Lagrangian bases permit convenient implementation of the quadrature rules which we now define. Let $f^k(\mathbf{r}) := f(\mathbf{x}^k(\mathbf{r}))$, $\mathbf{r} \in \hat{\Omega}$. In $\mathbb{R}^2$ we have:

$$(f,g)_{GL} := \sum_k \sum_{i=0}^{N} \sum_{j=0}^{N} f^k(\xi_i,\xi_j) \cdot g^k(\xi_i,\xi_j) \cdot |J^k(\xi_i,\xi_j)| \cdot \rho_i \rho_j \tag{7}$$

$$(f,g)_{G} := \sum_k \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} f^k(\eta_i,\eta_j) \cdot g^k(\eta_i,\eta_j) \cdot |J^k(\eta_i,\eta_j)| \cdot \sigma_i \sigma_j \tag{8}$$

where $J^k(\mathbf{r})$ is the Jacobian arising from the transformation $\mathbf{x}^k(\mathbf{r})$, ρ_i is the GL quadrature weight associated with ξ_i, and σ_i is the G quadrature weight associated with η_i. The extension to $\mathbb{R}^3$ follows readily from the tensor product forms.

2.3 Spectral element operators

The locally-structured/globally-unstructured bases of the spectral element method naturally define a two-level operator and data hierarchy which we now describe. Our notation will be two-dimensional, restricted to the case of affine mappings: $\mathbf{x}^k(r_1,r_2) = (x^k_{0,1} + \frac{L^k_1}{2} r_1, x^k_{0,2} + \frac{L^k_2}{2} r_2)$, where $x^k_{0,i}$ and L^k_i represent local translation and dilation constants.

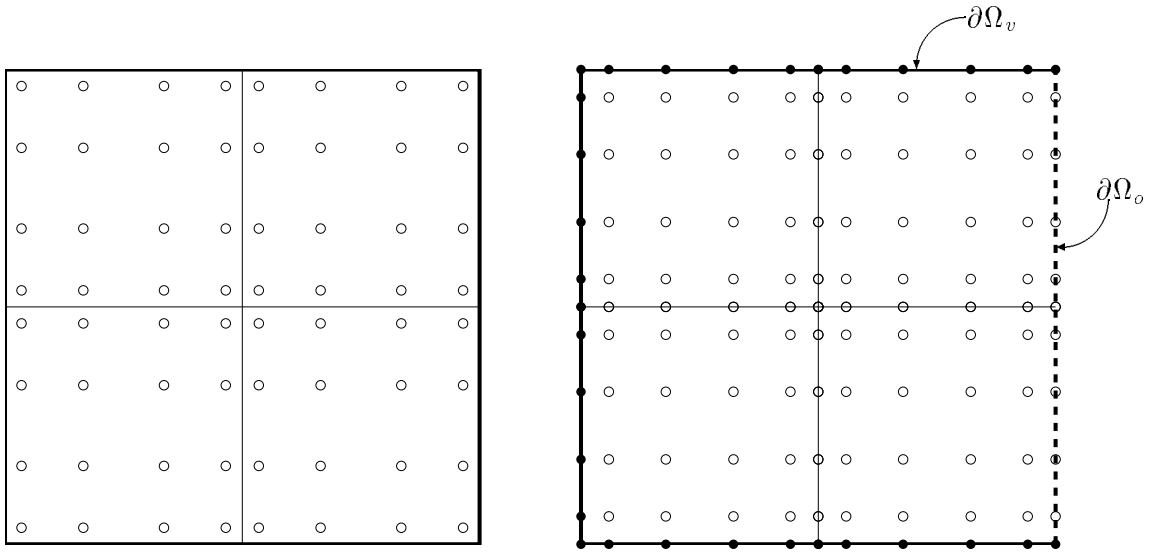

Figure 1: Spectral element configuration ($K = 4$, $N = 5$) showing Lagrange interpolation points for the pressure (Gauss) mesh on the left, and for the velocity (Gauss-Lobatto) mesh on the right. Open circles denote true degrees-of-freedom. Solid circles denote Dirichlet boundary nodes for velocity.

We first define the local bases and operators associated with the velocity space. Within a given subdomain, every scalar field in $\mathbb{P}_{N,K}(\Omega)$ is represented in the form:

$$f(\mathbf{x})|_{\Omega^k} \quad = \quad \sum_{i=0}^{N}\sum_{j=0}^{N} f_{ij}^k h_i(r_1)h_j(r_2) \qquad r_1,\, r_2 \;\in\; [-1,1]^2\,,$$

where $h_i(r) \in \mathbb{P}_N(r)$ is the Lagrange polynomial satisfying $h_i(\xi_j) = \delta_{ij}$, and δ_{ij} is the Kronecker delta function. For each subdomain, we associate a natural ordering of the nodal values f_{ij}^k, $i,j \in \{0,\ldots,N\}^2$ with the vector $\underline{f}^k$ and, in turn, associate a natural ordering of the vectors $\underline{f}^k$, $k \in \{1,\ldots,K\}$ with the $K(N+1)^2 \times 1$ vector $\underline{f}_L$. Note that if $f(\mathbf{x}) \in \mathsf{H}^1$, then $\underline{f}_L$ contains redundant information since basis coefficients on subdomain interfaces are represented in each adjoining subdomain.

We define the unassembled mass matrix to be the block-diagonal matrix $B_L := diag(B^k)$, where each local mass matrix is expressed as a tensor product of one-dimensional operators:

$$B^k \;=\; \left(\frac{L_1^k L_2^k}{4}\right)\, \hat{B}\otimes\hat{B} \qquad k \in \{1,\ldots,K\}\,.$$

Here, $\hat{B} = diag(\rho_i)$, $i = 0,\ldots,N$, is the one-dimensional mass matrix on the reference domain $[-1,1]$. In a similar fashion we define the unassembled stiffness matrix, or discrete Laplacian, as $A_L = diag(A^k)$, where A^k is the local stiffness matrix:

$$A^k \;=\; \left(\frac{L_2^k}{L_1^k}\right)\, \hat{B}\otimes\hat{A} \;+\; \left(\frac{L_1^k}{L_2^k}\right)\, \hat{A}\otimes\hat{B} \qquad k \in \{1,\ldots,K\}\,. \tag{9}$$

The one-dimensional stiffness matrix, $\hat{A}$, is defined in terms of the spectral differentiation matrix, $\hat{D}$:

$$\hat{A}_{ij} \;=\; \sum_{l=0}^{N} \hat{D}_{li}\rho_l\hat{D}_{lj} \quad i,j \;\in\; \{0,\ldots,N\}^2 \tag{10}$$

with

$$\hat{D}_{ij} \;:=\; \left.\frac{dh_j}{dr}\right|_{r=\xi_i} \qquad i,j \;\in\; \{0,\ldots,N\}^2\,.$$

Implementation details for fully deformed three-dimensional geometries are in [13].

The local subdomain operators A_L and B_L are formally incorporated into global $n_v \times n_v$ system matrices through the usual "direct stiffness" summation assembly procedure [35]. Let Q be the global-to-local mapping operator which transfers basis coefficients from their global ordering to their local ordering. The vector $\underline{f}_L = Q\underline{f}$ has basis coefficients duplicated in adjoining subdomains such that the corresponding continuous function $f(\mathbf{x})$ is in H^1. The action of Q^T upon a local vector, $\underline{f}_L$, is to sum any multiple contributions to global degrees-of-freedom from their constituent local nodal values. The assembled stiffness and mass matrices are given by Q^TA_LQ and Q^TB_LQ, respectively.

We refer to Q^TA_LQ as the Neumann Laplacian operator – it has a null-space of dimension unity corresponding to the constant mode. We define the associated Dirichlet operator by formally constructing a diagonal mask matrix, $\mathcal{M}$, which has ones on the diagonal, except at locations which correspond to Dirichlet boundary nodes where it is set to zero. We define the discrete Laplacian and mass matrices as: $A \;:=\; \mathcal{M}Q^TA_LQ\mathcal{M}$,

and $B := \mathcal{M}Q^T B_L Q \mathcal{M}$, respectively, and will treat them as being both invertible and symmetric positive definite (SPD), although this is not strictly true due to the null space associated with $\mathcal{M}$. Note that A is never formed explicitly. Only the *action* of A on a vector is required. This is computed via the tensor product form (9) with appropriate application of Q and $\mathcal{M}$.

2.4 Stokes operators

To complete the description of the Stokes operator we need to consider the bilinear form:

$$(q, \nabla \cdot \mathbf{u})_G \quad = \quad \sum_{l=1}^{d} \left(q, \frac{\partial u_l}{\partial x_l} \right)_G \ .$$

The definition (8) and the notations of the previous section give rise to the following matrix form in $\mathbb{R}^2$:

$$(q, \nabla \cdot \mathbf{u})_G \quad = \quad \sum_{k=1}^{K} (\underline{q}^k)^T \left(D_1^k \underline{u}_1^k \ + \ D_2^k \underline{u}_2^k \right) \ .$$

For the case of the affine mappings defined above, the local derivative matrices

$$D_1^k \ = \ \left(\frac{L_2^k}{2} \right) \tilde{I} \otimes \tilde{D} \ , \qquad D_2^k \ = \ \left(\frac{L_1^k}{2} \right) \tilde{D} \otimes \tilde{I} \ , \qquad k = 1, \ldots, K \ , \tag{11}$$

where

$$\tilde{D}_{ij} = \sigma_i \left. \frac{dh_j}{dr} \right|_{r=\eta_i} \ , \qquad \tilde{I}_{ij} = \sigma_i h_j(\eta_i) \qquad \begin{array}{l} i = 1, \ldots, N-1 \\ j = 0, \ldots, N \end{array}$$

are, respectively, the weighted one-dimensional differentiation and interpolation matrices mapping from the Gauss-Lobatto points to the Gauss points.

The extension from the local operator to the global operator proceeds exactly as in the previous section. The space of admissible functions in X^N is limited by the constraints that the velocity must be continuous at the subdomain interfaces and must satisfy the homogeneous boundary conditions, enforced by the action of the operators Q and $\mathcal{M}$, respectively. Let $D_i := D_{L,i} Q \mathcal{M}$, $i = 1, \ldots, d$, with $D_{L,i} := diag(D_i^k)$. In $\mathbb{R}^2$, the matrix form of the Stokes problem (4) is then:

$$\begin{bmatrix} H & & -D_1^T \\ & H & -D_2^T \\ -D_1 & -D_2 & 0 \end{bmatrix} \begin{pmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{p} \end{pmatrix} = \begin{pmatrix} \underline{f}_1 \\ \underline{f}_2 \\ \underline{f}_p \end{pmatrix} \ , \tag{12}$$

where $H = \frac{1}{Re} A + \frac{1}{\Delta t} B$ is the discrete Helmholtz operator.

2.5 Stokes solvers

A common approach to solution of the Stokes problem (12) is to decouple the velocity and pressure by formally carrying out block LU factorization (Uzawa decoupling) to yield a Schur complement system for the pressure, $S\underline{p} = \underline{g}$, which is solved iteratively. Here, $S = \sum_i D_i H^{-1} D_i^T$ and $\underline{g}$ is the corresponding inhomogeneity. Once the pressure is known, d Helmholtz solves serve to compute the velocity and complete the solution at time level n.

As it stands, the Uzawa approach requires a set of d Helmholtz solves for *each* iteration, since H^{-1} is embedded in S. An effective means to circumvent this difficulty is to decouple the viscous and pressure terms via an additional time splitting. Such an approach was suggested by Maday, Patera and Rønquist [24] and analyzed by Perot [4] and Couzy [7]; it follows classical splitting approaches (e.g., [18, 26]) which lead to a Poisson equation for the pressure except that, in the present case, the splitting is effected in the discrete form of the equations. The correct boundary conditions are preserved and no steady-state temporal errors are introduced.

Following [7], the unsteady Stokes system (12) is recast as:

$$\begin{bmatrix} \mathbf{H} & -\Delta t\,\mathbf{H}\mathbf{B}^{-1}\mathbf{D}^T \\ -\mathbf{D} & 0 \end{bmatrix} \begin{pmatrix} \underline{\mathbf{u}}^n \\ \underline{p}^n - \underline{p}^{n-1} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\underline{\mathbf{f}} + \mathbf{D}^T\underline{p}^{n-1} \\ \underline{f}_p \end{pmatrix} + \begin{pmatrix} \underline{\mathbf{r}} \\ \underline{0} \end{pmatrix} . \tag{13}$$

Here, boldface indicates the d-dimensional vector form of the previously defined operators. The residual

$$\underline{\mathbf{r}} \;:=\; (\Delta t\mathbf{H}\mathbf{B}^{-1} \;-\; \mathbf{I})\mathbf{D}^T(\underline{p}^{n-1} - \underline{p}^n)$$

is neglected, resulting in a method which is formally second-order accurate in time as noted in [4, 7]. Applying block Gaussian elimination to (13) (without the residual term) yields the reformulated Stokes problem to be solved at each time step:

$$\begin{bmatrix} \mathbf{H} & -\Delta t\mathbf{H}\mathbf{B}^{-1}\mathbf{D}^T \\ \mathbf{0} & E \end{bmatrix} \begin{pmatrix} \underline{\mathbf{u}}^n \\ \underline{p}^n - \underline{p}^{n-1} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\underline{\mathbf{f}} + \mathbf{D}^T\underline{p}^{n-1} \\ \underline{g} \end{pmatrix} \quad , \tag{14}$$

where

$$E \;:=\; \Delta t \sum_{i=1}^{d} D_i\, B^{-1}\, D_i^T \quad , \tag{15}$$

and $\underline{g}$ is the modified inhomogeneity arising from Gaussian elimination. The advantage of the splitting procedure is that matrix vector products involving E can be computed *without* system solves, since B is diagonal.

To summarize, time advancement of the Navier-Stokes equations involves: evaluating the contributions from the convective terms, solving for the viscous contribution in the construction of g (14), solving for the pressure (14-15), and finally computing the divergence-free solution, $\mathbf{u}^n$ (14). The systems involving H and E are solved iteratively. The Schur complement matrix, E, is the most ill-conditioned of the subproblems and we address strategies for preconditioning it next.

3 Preconditioning E

Since E is SPD, save a possible one-dimensional null space associated with the hydrostatic pressure mode in cases where $\partial\Omega_o = \emptyset$, preconditioned conjugate gradient iteration can be employed if a suitable SPD matrix, M^{-1}, can be found which is spectrally close to E^{-1}. Furthermore, we require that $M^{-1}E$ be easily computable. There are several features of E which make this task difficult. First, due to the embedded interpolation between the pressure space and the velocity space, the computational stencil of E is locally full with $O(N^d)$ nonzeros per row. (However, matrix-vector products involving E can be *evaluated* in only $O(KN^{d+1})$ operations due to the tensor-product forms (11).) Second, because it

is in L^2, no continuity or boundary conditions are applied directly to the pressure – these conditions are enforced in the velocity space. Fortunately, as discussed in [4], E is in many respects similar to a discrete Laplacian with suitable boundary conditions. Hence, classical preconditioning strategies developed for finite element discretizations of Poisson's equation can be used as a basis for the development of a preconditioner for E.

3.1 Laplacian based preconditioning for E

To illustrate the equivalence of E and the Laplacian we consider preconditioned conjugate gradient iteration for the pressure on the first step of impulsively started flow past a cylinder. The $K = 93$ spectral element mesh is shown in Fig. 2 and is typical of many (conforming) production meshes, so we have used it as a baseline in a number of studies [16, 17]. The cylinder of diameter $D = 1$ is centered at the origin in the half-domain $\Omega = [-10, 28]\times[0, 15]$. The Reynolds number is $Re = DU/\nu = 5000$, where $(U, 0)$ is the free-stream velocity taken both as the initial condition and the inflow boundary condition at $x = -10$. Symmetry boundary conditions are imposed at $y = 0$ and $y = 15$ with Neumann-velocity (outflow) boundary conditions at $x = 28$. The free stream velocity is $U = 1$ and the time step is $\Delta t = 0.025$.

The preconditioners considered are all global and based upon a Laplacian with homogeneous Dirichlet boundary conditions specified at outflow ($\partial\Omega_o$) and homogeneous Neumann conditions on the remainder of the boundary. The first preconditioner, $M_s^{-1} := R_{vp}^T A_s^{-1} R_{vp}$, is based upon the spectral element Laplacian, A_s, and a prolongation operator, R_{vp}^T, which interpolates from the pressure to the velocity mesh. The second, M_t, is identical to M_s, save that A_s is replaced by a finite element based Laplacian using a triangulation of the Gauss-Lobatto points. The third, $M_g^{-1} := R_{aug}^T A_g^{-1} R_{aug}$, uses a discrete Laplacian based upon a triangulation of the Gauss (pressure) points, augmented with additional points to enforce the Dirichlet boundary condition.

Table 1 shows the number of iterations required to reduce the residual of the E system by 10^{-5} for varying degree and number of elements. The $K = 372$ and $K = 1488$ meshes are obtained through successive quarterings of the elements in the base configuration. The dimension of the E system is $K(N-1)^2$. Note that the spectral preconditioner exhibits no K dependence in the iteration count for this problem, while the finite element preconditioners exhibit mild degradation in the rate of convergence. Of the two finite element precondition-

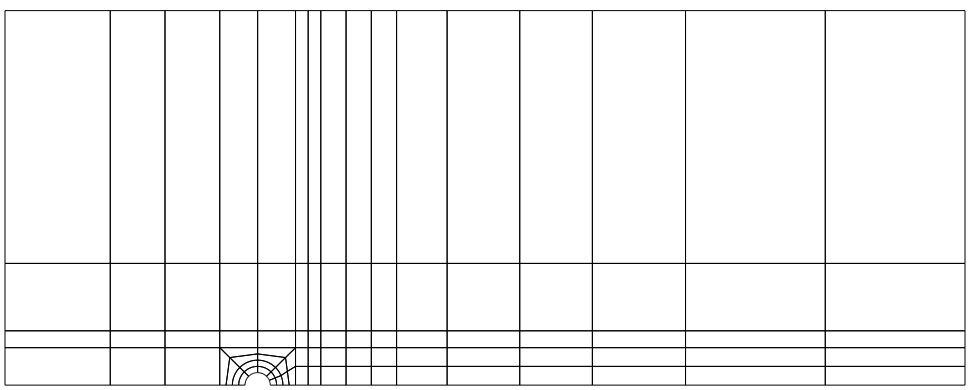

Figure 2: Spectral element mesh ($K = 93$) for iterative convergence study.

Table 1: Laplacian preconditioners for cylinder flow.

	$N = 7$			$N = 9$		
K	M_s	M_t	M_g	M_s	M_t	M_g
93	15	24	16	12	27	16
372	12	26	18	12	28	17
1488	12	26	18	12	29	18

ers, it is clear that M_g is to be preferred over M_t. Not only does it yield a lower iteration count, it also has a much less expensive restriction operator (R_{aug} is essentially the identity matrix) and a lower dimensional system to be solved with each iteration.

4 Overlapping Schwarz Method

The Laplacian-based preconditioners described above require the solution of a global linear system on each iteration. In practice, this is too expensive and a suitable approximation, or preconditioning strategy, is required to replace the global solve. This problem is decidedly easier than preconditioning E directly as there exist many good preconditioners for the finite element based Laplacian.

Domain decomposition based preconditioners are a natural choice for the spectral element method since the data is structured within a subdomain but is otherwise unstructured. Here we use the overlapping additive Schwarz procedure developed by Dryja and Widlund [11] and Widlund [37]. We have chosen the additive Schwarz procedure because it is intrinsically parallel and symmetric. Although multiplicative procedures have better convergence rates and often can also be parallelized and symmetrized [34], this is not easy to do if the originating operator (E) has less locality than the preconditioner, as in the present case. We have chosen the overlapping method as opposed to a Schur complement, or substructuring, approach because the additive method does not require interface data structures, an important consideration in this case since the pressure nodes do not lie on the natural subdomain interfaces.

Formally, the additive Schwarz preconditioner is expressed as the sum of outputs from several subproblems:

$$M_o^{-1} \;=\; R_0^T A_0^{-1} R_0 \;+\; \sum_{k=1}^{K} R_k^T A_k^{-1} R_k \quad .$$

The subproblems for $k \geq 1$ correspond to the solution of local Poisson problems on overlapping subdomains, $\tilde{\Omega}^k$. The restriction and prolongation operators, R_k and R_k^T, $k \geq 1$, are Boolean matrices which transfer data to and from the subdomain problems. The product $\underline{p}_k = R_k \underline{p}$ extracts the components of a vector $\underline{p}$ which belong to $\tilde{\Omega}^k$, while $\underline{p} = R_k^T \underline{p}_k$ copies the components of a subdomain solution, $\underline{p}_k$, to a global vector, $\underline{p}$, and sets components outside of $\tilde{\Omega}^k$ to zero. In addition to the local problems, the Schwarz preconditioner has a coarse grid component, denoted here by subscript 0, which serves to efficiently eliminate low-wave number components of the residual. The coarse grid problem corresponds to a Poisson problem discretized on a mesh defined by a triangulation of the subdomain vertices. The prolongation operator, R_0^T, is simply an interpolant from the coarse grid to the Gauss points.

4.1 FDM application to the subdomain problems

In this section we consider the development of solvers for the local problems which are particularly well suited to the spectral element method in $\mathbb{R}^3$. Rather than working with principal submatrices of A_g as was done in [17], we derive the local stiffness matrices, A_k, $k \geq 1$, from a tensor-product of one-dimensional finite element bases. This difference in strategy is reflected in Fig. 3 which contrasts the previous unstructured finite element (FEM) basis on the left with the structured tensor-product basis on the right. This allows the use of FDM based solvers which require only $O(N^d)$ storage and $O(N^{d+1})$ work per solve [7, 8, 31]. An added benefit is the avoidance of having to tetrahedralizing the Gauss points in $\mathbb{R}^3$.

We begin with the definition of the overlapping subdomains by considering the one-dimensional example shown in Fig. 4. Degrees-of-freedom are associated with the nodes (open circles) in $\tilde{\Omega}^k$. The points ζ_i, $i \in \{1, \ldots, N-1\}$ are the images of the Gauss points in $]-1, 1[$ mapped onto Ω^k. Similarly, ζ_i, $i \leq 0$ and $i \geq N$ are the images of the corresponding Gauss points mapped onto the left and right subdomains, respectively. The overlapping region, $\tilde{\Omega}^k \in [\zeta_{-1}, \zeta_{N+1}]$, is obtained by extending Ω^k by two nodal points in each direction. Homogeneous Dirichlet boundary conditions are applied at ζ_{-1} and ζ_{N+1} when Ω^k is in the interior of Ω so the extension only adds two degrees-of-freedom to the local problem. We refer to this as the minimal overlap case. If the left (right) side of $\partial\Omega^k$ is coincident with the boundary, $\partial\Omega$, then the domain is not extended beyond γ_0 (γ_N) and homogeneous Dirichlet or Neumann boundary conditions are imposed at that point in accordance with the boundary conditions on $\partial\Omega$.

To construct the finite element operators for the standard (interior) one-dimensional case, we consider the space of piecewise linear functions, $\phi_i(\zeta)$, $\zeta \in [\zeta_{-1}, \zeta_{N+1}]$, $i = 0, \ldots, N$:

$$\phi_i(\zeta) \quad = \quad \begin{cases} \frac{\zeta - \zeta_{i-1}}{\zeta_i - \zeta_{i-1}} & \zeta_{i-1} \leq \zeta < \zeta_i \\ \frac{\zeta - \zeta_{i+1}}{\zeta_i - \zeta_{i+1}} & \zeta_i \leq \zeta < \zeta_{i+1} \\ 0 & \text{otherwise.} \end{cases} \qquad i \in \{0, \ldots, N\} \tag{16}$$

The variational form for the homogeneous Dirichlet problem, $-u''(x) = f(x)$ in $\tilde{\Omega}^k$, $u = 0$

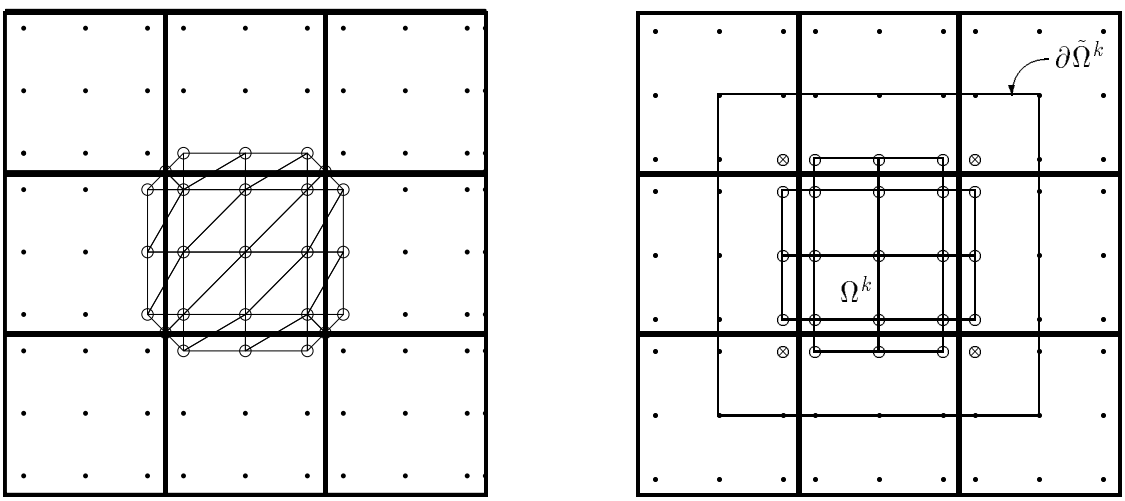

Figure 3: Degrees-of-freedom (open circles) for FEM based (left) and tensor-product based (right) discretizations of local problems. Values at nodes marked "$\otimes$" are set to zero by R_k. Zero Dirichlet boundary conditions are applied on $\partial\tilde{\Omega}^k$.

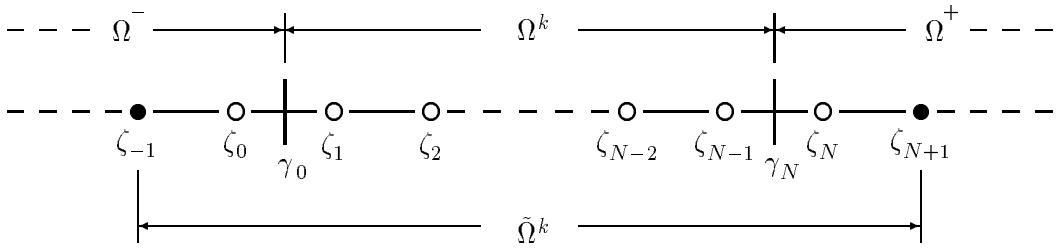

Figure 4: Depiction of overlapping subdomain $\tilde{\Omega}^k$ in one dimension, minimal overlap case.

on $\partial\tilde{\Omega}^k$, gives rise to the tridiagonal stiffness matrix:

$$\tilde{A}_{ij} \;=\; \int_{\zeta_{-1}}^{\zeta_{N+1}} \frac{d\phi_i}{d\zeta}\frac{d\phi_j}{d\zeta} d\zeta \quad i,j \;\in\; \{0,\ldots,N\}^2 \;,$$

and associated diagonal (lumped) mass matrix:

$$\tilde{B}_{ij} \;=\; \delta_{ij} \int_{\zeta_{-1}}^{\zeta_{N+1}} \phi_j(\zeta) d\zeta \quad i,j \;\in\; \{0,\ldots,N\}^2 \;.$$

The matrices are modified in the usual way if either end of Ω^k coincides with $\partial\Omega$.

The construction of the one-dimensional problem is extended to $\mathbb{R}^d$ by taking the tensor product of the bases and operators just described. A typical overlapping domain in $\mathbb{R}^2$ is shown in Fig. 3 (right). The degrees-of-freedom correspond to Lagrangian basis coefficients associated with the nodes (open circles) in the interior of $\tilde{\Omega}^k$. If the nodes are numbered lexicographically, then the stiffness matrix for the two-dimensional Laplacian on $\tilde{\Omega}^k$ can be written as the Kronecker product:

$$A_k \;=\; \tilde{B}_2 \otimes \tilde{A}_1 \;+\; \tilde{A}_2 \otimes \tilde{B}_1 \;. \tag{17}$$

Here, the subscript on the one-dimensional matrices, $\tilde{A}$ and $\tilde{B}$, indicates the associated coordinate direction in the reference element.

Matrices which satisfy (17) have a particularly simple inverse based upon the FDM. If $\tilde{A}$ is symmetric and $\tilde{B}$ is symmetric positive definite, then the following similarity transformation holds:

$$S^T \tilde{A} S = \Lambda, \qquad S^T \tilde{B} S = I,$$

where $S = (\underline{s}_1, \ldots, \underline{s}_n)$ is the matrix of eigenvectors, and $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$ the matrix of eigenvalues associated with the generalized eigenvalue problem $\tilde{A}\underline{s} = \lambda \tilde{B}\underline{s}$. As a result, A_k is readily diagonalized, and its inverse is given by:

$$A_k^{-1} \;=\; (S_2 \otimes S_1)\,(I \otimes \Lambda_1 \;+\; \Lambda_2 \otimes I)^{-1}\,(S_2^T \otimes S_1^T)\,.$$

The three-dimensional form is similar:

$$A_k^{-1} \;=\; (S_3 \otimes S_2 \otimes S_1)\, D^{-1} (S_3^T \otimes S_2^T \otimes S_1^T)\,,$$

with

$$D \;=\; (I \otimes I \otimes \Lambda_1 + I \otimes \Lambda_2 \otimes I + \Lambda_3 \otimes I \otimes I).$$

This solution method was introduced by Lynch *et al.* [31] and successfully used in a number of spectral element preconditioning applications by Couzy and Deville [8] and by Couzy [7].

It is important to note that the use of tensor-product forms allows matrix-vector products to be recast as *matrix-matrix* products which are particularly efficient on modern vector and cache-based processors. For example, if $\underline{\tilde{u}}^k = u^k_{ij}$, $i,j \in \{0,\ldots,N\}^2$ is the set of nodal basis coefficients on $\tilde{\Omega}^k$, then

$$(S_2^T \otimes S_1^T)\underline{u}^k = S_1^T \tilde{U}^k S_2\,,$$

where $\tilde{U}^k$ is simply $\underline{\tilde{u}}^k$ viewed as the matrix having entries u^k_{ij}. In two dimensions, the computation of $A_k^{-1}\underline{\tilde{u}}^k$ requires four matrix-matrix products. The leading order complexity for the work is $8(N+1)^3$ operations and for storage is $3(N+1)^2$. This compares quite favorably with a banded solver which requires $4(N+1)^3$ operations and $(N+1)^3$ storage. In three dimensions, the leading order complexities for the tensor product based solver are $12(N+1)^4$ operations and $(N+1)^3$ words of storage. In this case the banded solver is not competitive as it requires $4(N+1)^5$ operations and $(N+1)^5$ words of storage.

In general, the FDM cannot be used for arbitrarily deformed subdomains since it is not always possible to express the Laplacian in the tensor product form (17). However, for the purposes of a preconditioner, all one requires is an approximation to the Laplacian on $\tilde{\Omega}^k$. Two essential simplifications make it possible to apply the FDM to arbitrary (valid) spectral element meshes in $\mathbb{R}^d$. The first is to construct restriction operators R_k which only gather data from the $2d$ subdomains which share a common face (edge) with Ω^k and which map zeros to nodes in the overlap region corresponding to diagonally adjacent elements (denoted by $\otimes$ in Fig. 3). This circumvents difficulties arising from cases where, e.g., three or five elements share a vertex in $\mathbb{R}^2$.

The second simplification, as suggested in the thesis of Couzy [7], is to define the Poisson problem on a rectangle or regular parallelepiped having the correct average dimensions in each coordinate direction in the reference domain. These dimensions are readily computed using Gauss-Lobatto quadrature. For example, in $\mathbb{R}^2$, the average separation of the faces in the r_1-direction is computed as:

$$L_1^k \;=\; \left[\frac{\sum_j \rho_j[(x^k_{Nj} - x^k_{0j})^2 \;+\; (y^k_{Nj} - y^k_{0j})^2]}{\sum_j \rho_j}\right]^{\frac{1}{2}}$$

A similar procedure is used to find the average thickness (e.g., the distance $|\zeta_1 - \zeta_{-1}|$ in Fig. 4) of each of the overlap regions. With these average dimensions, construction of the Laplacian follows directly from the tensor product formulation outlined above.

5 The coarse grid problem

The coarse grid operator, A_0, is derived from a linear finite element discretization of the Laplacian subject to homogeneous Dirichlet boundary conditions at outflow ($\partial\Omega_o$) and homogeneous Neumann conditions on the remainder of the boundary. The coarse grid space is based upon linear elements having nodes at the spectral element vertices. The prolongation operator, R_0^T is simply a bi- or tri-linear interpolant from the 2^d vertices to the $(N-1)^d$ Gauss points in the reference domain, and can be cast as a sequence of efficient matrix-matrix products.

In two-dimensions, the quadrilateral spectral element mesh is readily triangulated by connecting one pair of diagonally opposing vertices in each of the elements. In three dimensions, an equivalent local procedure is complicated by the fact that the tetrahedral

decomposition of a cube introduces a diagonal on each face which must match the direction of the diagonal introduced on the face of the adjoining cube for the resulting discretization to be conforming. The tessellation problem in $\mathbb{R}^3$ can be localized by computing the (local) stiffness matrices for the two complementary tetrahedralizations of the reference cube shown on the left in Fig. 5. If A' and A'' denote the global matrices obtained by assembling compatible sets of local stiffness matrices, then $A_0 := \frac{1}{2}(A' + A'')$ is the global stiffness matrix one would obtain by taking the average of two conforming stiffness matrices. However, A_0 can be constructed without solving the non-local problem of determining a conforming tetrahedralization by simply assembling the local stiffness matrices of all ten tetrahedra defined by the two complementary decompositions.

5.1 Parallel coarse grid solver

Solution of the coarse grid problem has long been recognized as a bottleneck in parallel applications where communication costs are non-negligible, such as on networks of workstations or when the number of processors is large (see, e.g., [2, 12, 22, 34]). Since A_0^{-1} is full, each coarse grid solve requires an all-to-all communication as every entry of the distributed input has a nontrivial impact on every output value. Assuming that each processor is capable of sending or receiving only one message at a time and that contention-free communication time for an m-word message obeys a linear cost model of the form $t_c[m] = \alpha + \beta m$, then the minimum time for solution of the distributed coarse grid problem is $\alpha \log_2 P$. It is typically best to use a contention-free routing schedule which implies a minimum time of $2\alpha \log_2 P$ for standard schedules on low-dimensional networks.

As noted by Gropp in [19], most parallel solvers for an $n \times n$ coarse grid problem require $\log_2 P$ messages of length n for each solve. Since $n > P$, this can become prohibitive if either β or P is large. We have recently developed a parallel coarse grid solver with a communication complexity of only $O(n^{\frac{1}{2}}\beta log_2 P)$ in $\mathbb{R}^2$ and $O(n^{\frac{2}{3}}\beta log_2 P)$ in $\mathbb{R}^3$ [16, 36]. The solver derives from the observation that projection of a distributed vector onto a distributed basis is naturally parallel.

Let $A_0\underline{x} = \underline{b}$ denote the $n \times n$ coarse grid system to be solved, with $\underline{b}$ and $\underline{x}$ identically distributed across processors at the beginning and end of the solution phase. If $X = (\underline{x}_1, \ldots, \underline{x}_l) \in \mathbb{R}^{n \times l}$ is a matrix of A_0–orthonormal vectors satisfying $\underline{x}_i^T A_0 \underline{x}_j = \delta_{ij}$, then the projection of $\underline{x}$ onto $span\{\underline{x}_1, \ldots, \underline{x}_l\}$ is given by $\bar{\underline{x}} = XX^T\underline{b}$. If the $\underline{x}_i$'s are mapped in the same manner as $\underline{x}$ and $\underline{b}$, then parallel evaluation of $\bar{\underline{x}}$ on P processors involves three steps:

$$\begin{array}{lllll} i. & \alpha_i^{(p)} & = & \left(\underline{x}_i^{(p)}\right)^T \underline{b}^{(p)} & i = 1, \ldots, l \qquad p = 0, \ldots, P-1 \\ ii. & \alpha_i & = & \sum_p \alpha_i^{(p)} & i = 1, \ldots, l \\ iii. & \bar{\underline{x}}^{(p)} & = & \sum_i \alpha_i \underline{x}_i^{(p)} & p = 0, \ldots, P-1\,. \end{array} \tag{18}$$

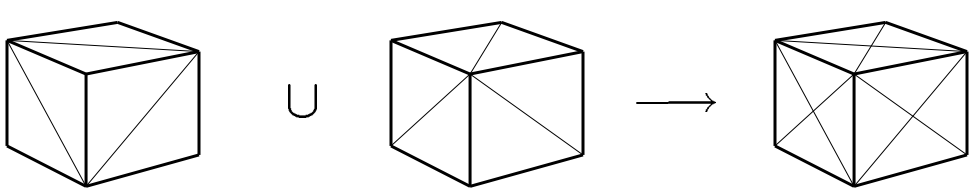

Figure 5: The symmetric union of two complementary decompositions of the reference cube localizes the problem of finding a conforming coarse grid space.

Here, the superscript (p) indicates the processor index for distributed data. Step (ii) is an inter-processor vector-reduction and can be computed via a fan-in/fan-out on a binary tree in $2\log_2 P$ communication phases with messages of length l.

Note that if $l = n$, then $\underline{\bar{x}} \equiv \underline{x}$, and the projection produces the exact solution. If A_0 is sparse, it is possible to choose a (quasi-) sparse basis for X such that many of the $\underline{x}_i$'s are void on any given processor. This implies a reduction in communication as well as work since the corresponding values of α_i are not needed by all processors. For low-order discretizations in $\mathbb{R}^d$ it is possible to choose the columns of X such that it has only $O(n^{\frac{2d-1}{d}})$ nonzeros and such that only $2\log_2 P$ messages of length $O(n^{\frac{d-1}{d}})$ are required to compute $XX^T\underline{b}$. Further details may be found in [36].

We note that the vertex-based coarse grid problems such as considered here nominally require communication in the restriction and prolongation steps because each vertex may be shared by many processors. These extra communications can be avoided by embedding them directly into the XX^T computation. Let $\underline{z} := R_0^T A_0^{-1} R_0 \underline{r}$, denote the full coarse grid problem. Consider the factorization:

$$R_0 \;=\; Q_0^T Q_P^T \mathcal{I}_P^T \tag{19}$$

where $\mathcal{I}_P$ represents the local interpolation from the subdomain vertices to the Gauss points, Q_P^T represents the direct-stiffness summation (or assembly of the load vector) of vertex values within each processor, and Q_0^T represents the interprocessor direct-stiffness summation step. Only the application of Q_0 (Q_0^T) requires communication. Writing $X_0 := XQ_0$, we have

$$\underline{z} = (\mathcal{I}_P Q_P)\, X_0 X_0^T \left(Q_P^T \mathcal{I}_P^T \underline{r} \right) .$$

This corresponds to computing a projection of the form $\underline{x}_0 = X_0 X_0^T \underline{b}_0$ and is identical in complexity to (18) on an enlarged vector space. No pre- or post-communication is required since $(\mathcal{I}_P Q_P)$ is local.

6 Numerical Results

We compare the results of the FDM-based additive Schwarz method to the results of the FEM-based additive Schwarz preconditioner [17] and the block-Jacobi/deflation based scheme developed in [15, 32].

6.1 Two-dimensional cylinder problem

We first consider the cylinder problem of Fig. 2. The conditions are the same as those used in the Laplacian preconditioning tests of Section 3.1 save that we restrict the polynomial order to $N = 7$. Table 2 shows the iteration count and CPU times for the FDM-based additive Schwarz procedure with minimal overlap. Also shown are the iteration counts and times for the additive Schwarz procedure based upon the unstructured FEM discretization where the local stiffness matrices, A_k, $k \geq 1$, are principal submatrices of A_g. The $N_o = 0$ column corresponds the the FEM scheme with no overlap. Introducing a minimal amount of overlap ($N_o = 1$) reduces the iteration count almost two-fold and the CPU time slightly less than two-fold. Increasing the overlap to $N_o = 3$ does not yield much further improvement. The importance of the coarse grid solve is illustrated by the $A_0 = 0$ column, which shows a five- to eight-fold increase in iteration count for the $K = 1488$ case when the coarse grid solver is excluded. The final column shows the performance of our deflation based production code

[15]. It requires roughly twice the number of iterations as the FDM scheme and almost three times the CPU time. (The deflation approach requires two applications of E per iteration.)

Table 2: Performance of the additive Schwarz algorithm

	FDM		$N_o = 0$		$N_o = 1$		$N_o = 3$		$A_0 = 0$		Deflation	
K	iter	CPU	iter	CPU	iter	CPU	iter	CPU	iter	CPU	iter	CPU
93	67	4.4	121	10.	64	5.9	49	5.6	169	19.	126	17.
372	114	37.	203	74.	106	43.	73	39.	364	193.	216	125.
1488	166	225.	303	470.	158	274.	107	242.	802	1798.	327	845.

We note that, due to the use of the approximate Laplacians, the FDM-based scheme has a slightly higher iteration count than the FEM scheme in the minimal overlap case ($N_o = 1$). Despite this and despite its higher complexity estimate ($8K(N+1)^3$ vs $4K(N+1)^3$) the FDM-based scheme requires less time. This clearly illustrates the importance of the matrix-matrix product based solution algorithm.

Somewhat disappointingly, the iteration counts for the overlapping Schwarz method are not bounded with K. Our experience indicates that the iteration count does eventually approach a bound, but only after many levels of refinement. We have investigated two possible solutions. The first, suggested by Widlund [38], is to use more overlap on the (few) subdomains which have high aspect ratio. This reduces the iteration count while maintaining low CPU time [17]. The second is use of nonconforming spectral element methods which remove these high aspect ratio subdomains all together. As demonstrated by G. Kruse [20] this results in significantly lower iteration counts.

6.2 Three-dimensional hemisphere problem

We now consider parallel simulation of the three-dimensional flow arising from the interaction of a flat plate boundary layer with a hemispherical protuberance. This flow was studied experimentally by Acalar and Smith [1] and, at sufficiently high Reynolds numbers, exhibits periodic shedding of hairpin vortices as evidenced by the isotherms in the centerplane of the channel shown in Fig. 6. The unit radius hemisphere is centered at $\mathbf{x} = (0, 0, 0)$ and

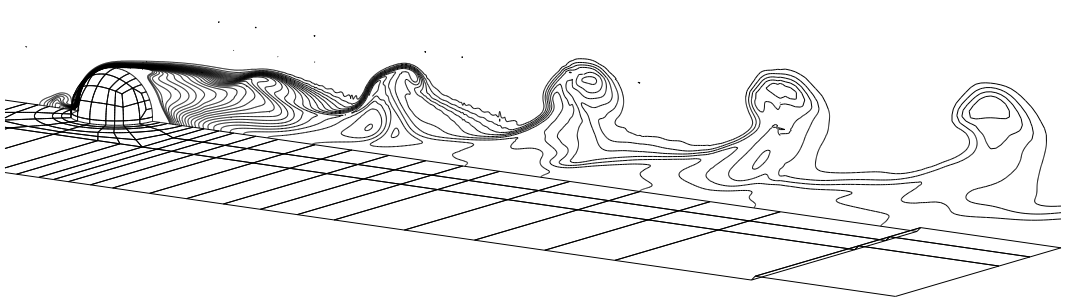

Figure 6: Isotherms reveal the presence of hairpin vortices generated by the interaction of a flat-plate boundary later with a (heated) hemisphere in this ($K = 512, N = 11$) spectral element simulation.

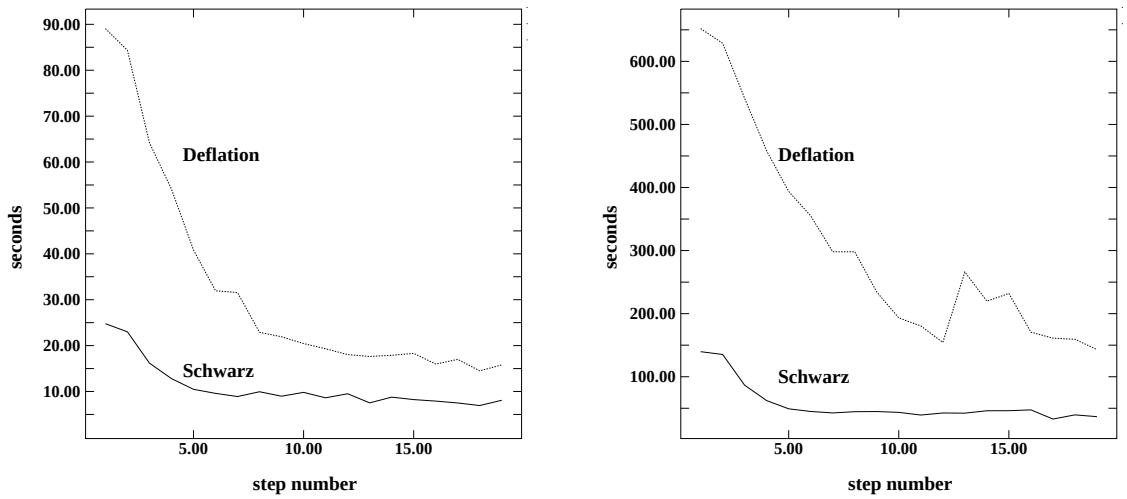

Figure 7: 512-node CPU time for the first 19 steps of the hemisphere problem for $(K, N) = (512, 11)$ (left) and $(4096, 9)$ (right).

the Reynolds number is $Re = \frac{RU_\infty}{\nu} = 500$. A Blasius profile with $\delta_{.99} = 1.15$ and $U_\infty = 1$ is specified for the x-component of velocity both as an initial condition and inlet profile at $x = -8.4$. Symmetry boundary conditions are specified at $y = 0$, $y = -6.4$, and $z = 6.5$, and Neumann outflow boundary conditions are imposed at $x = 25.6$. Discretizations consisting of $K = 512$ and 4096 spectral elements of order $N = 7$, 9, and 11 are considered for a fixed time step of $\Delta t = 0.00636$. Timings are performed on the $P = 512$ node Intel Delta at Caltech, which is a mesh connected multicomputer based on 512 Intel i860 40MHz microprocessors, each with 16 Mbytes of memory. Sustained performance on this machine for these runs is typically about 5 GFLOPS in 32-bit arithmetic.

In Fig. 7 we show the CPU time per step for first 19 steps of the deflation- and FDM-based computations with $(K, N) = (512, 11)$ and (4096,9). A good initial guess, computed from an orthogonal projection of the data onto previous solutions, significantly reduces the iteration count after the first few time steps, so the performance at later times is most representative of the asymptotic behavior of the solvers during the course of the simulation [14]. Table 3 shows the number of pressure iterations and CPU time required for the 19th step. We observe that the overlapping Schwarz procedure yields a three- to five-fold reduction in iteration count over the deflation scheme and a a four-fold improvement in CPU time for the largest problem. The fact that the CPU-time reduction is less than that of the iteration count shows that the overlapping Schwarz procedure has effectively eliminated the pressure solve as the computational bottleneck for this class of problems. In

Table 3: Timing for hemisphere/plate problem on 512 node Delta

				Deflation		Schwarz	
K	N	# vel. pts.	# pres. pts.	# iters.	sec./step	# iters.	sec./step
512	7	179000	111000	19	5.3	4	3.2
512	9	380000	262000	27	8.8	5	4.7
512	11	693000	512000	36	15.8	13	8.1
4096	7	1423000	884000	78	58.4	20	18.2
4096	9	3016000	2097000	137	143.0	26	36.7

fact, it is now on par with the cost of the Helmholtz solves.

We evaluate the performance of the XX^T based coarse grid solver in our overlapping Schwarz code via direct comparison to the same production code but modified to use a distributed A^{-1} based solver which requires a $\log_2 P$ gather of a full n-vector for each coarse grid solve. For $K = 512$ the dimension of the coarse grid system is $n = 781$, while for $K = 4096$ it is $n = 5114$. Table 4 indicates the percentage of overall solution time spent in the coarse grid solver as well as the time per coarse grid solve for both cases. For the $K = 4096$ data, there is a twelve and nine percent reduction in overall solution time due to the use of XX^T based solver. In addition, there is four-fold improvement in the time per coarse grid solve for $n = 5114$, and two-fold for $n = 781$. We note that this is consistent with predictions based on the theoretical models for both solvers discussed in [36].

Table 4: Coarse grid costs for hemisphere/plate problem on 512 node Delta

			A^{-1} Method		XX^T Method	
K	N	# d.o.f.	% time	sec./slv.	% time	sec./slv.
512	7	781	4.82	0.021	2.66	0.011
512	9	781	5.32	0.021	3.03	0.011
512	11	781	5.38	0.021	3.08	0.012
4096	7	5114	16.4	0.091	4.26	0.021
4096	9	5114	12.6	0.094	3.45	0.023

7 Conclusions

We have developed an overlapping Schwarz preconditioner for the pressure-subproblem in time-split spectral element formulations of the incompressible Navier-Stokes equations which is particularly efficient for problems in three dimensions. The method employs tensor-product discretizations for the local subdomain problems which admit solution via fast diagonalization techniques having the same computational complexity as the originating spectral element operators, and which are readily implemented within the locally structured context of the spectral element method. The parallel performance of the method is enhanced by a fast coarse grid solve algorithm which has significantly better communication complexity than competing approaches. In comparison to our earlier block-Jacobi/deflation based production code, we observe a five-fold reduction in iteration count and, for the largest problems, a four-fold reduction in CPU time. The overlapping Schwarz preconditioner has effectively eliminated the pressure solve as a computational bottleneck. It now becomes important to consider whether the other phases of the solution process can be further improved.

Acknowledgments

This work was supported by the NSF under Grant ASC-9405403 and by the AFOSR under Grant F49620-95-1-0074. Computer time was provided on the Intel Delta at Caltech by the Center for Research on Parallel Computation under NSF Cooperative agreement CCR-8809615.

References

[1] M.S. Acalar and C.R. Smith, "A study of hairpin vortices in a laminar boundary layer. Part 1. Hairpin vortices generated by a hemisphere protuberance.", *J. Fluid Mech.*, **175**, pp. 1-41 (1987).

[2] F. Alvarado, A. Pothen, and R. Schreiber, "Highly parallel sparse triangular solution", Univ. Waterloo Research Rep. CS-92-51, Waterloo, Ontario (1992).

[3] C. Bernardi and Y. Maday, "A collocation method over staggered grids for the Stokes problem", *Int. J. Numer. Meth. Fluids*, **8** , pp. 537-557 (1988).

[4] J. Blair Perot, "An analysis of the fractional step method", *J. Comput. Phys.*, **108** , pp. 51-58 (1993).

[5] M. Casarin "Quasi-optimal Schwarz methods for the conforming spectral element discretization", *Tech.Rep. 705*, Dept. Comp. Sci., Courant Inst., NYU (1995).

[6] M. Casarin, "Schwarz preconditioners for spectral and mortar finite element methods with applications to incompressible fluids", Ph.D. Thesis, Courant Institute of Math. Sci., NYU (1996).

[7] W. Couzy, "Spectral element discretization of the unsteady Navier-Stokes equations and its iterative solution on parallel computers", Thèse No. 1380, École Polytechnique Fédérale de Lausanne (1995).

[8] W. Couzy and M.O. Deville, "A Fast Schur Complement Method for the Spectral Element Discretization of the Incompressible Navier-Stokes Equations", *J. Comput. Phys.*, vol 116, pp. 135-142 (1995).

[9] P. Demaret and M.O. Deville, "Chebyshev pseudo-spectral solution of the Stokes equations using finite element preconditioning", *J. Comput. Phys.*, **83**, pp. 463-484 (1989).

[10] M.O. Deville and E.H. Mund, "Finite element preconditioning for pseudospectral solutions of elliptic problems", *SIAM J. Statis. Comput.*, **11**(2), pp. 311-42 (1990).

[11] M. Dryja and O.B. Widlund, "An additive variant of the Schwarz alternating method for the case of many subregions", *Tech. Rep. 339*, Dept. Comp. Sci., Courant Inst., NYU (1987).

[12] C. Farhat and P.S. Chen, "Tailoring Domain Decomposition Methods for Efficient Parallel Coarse Grid Solution and for Systems with Many Right Hand Sides", *Contemporary Math.*, **180**, pp. 401-406 (1994).

[13] P.F. Fischer, "Spectral Element Solution of the Navier-Stokes Equations on High Performance Distributed-Memory Parallel Processors", Ph.D. Thesis, Massachusetts Institute of Technology (1989).

[14] P.F. Fischer, "Projection techniques for iterative solution of $A\underline{x} = \underline{b}$ with successive right-hand sides", *ICASE Report No. 93-90*, NASA CR-191571 (1993).

[15] P.F. Fischer and E.M. Rønquist, "Spectral Element Methods for Large Scale Parallel Navier-Stokes Calculations", *Comp. Meth. Appl. Mech. Engr.*, pp. (1994).

[16] P.F. Fischer, "Parallel multi-level solvers for spectral element methods", in *Proceedings of Intl. Conf. on Spectral and High-Order Methods '95, Houston, TX*, A.V. Ilin and L.R. Scott eds., pp. (1996).

[17] P.F. Fischer, "An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations", *J. of Comp. Phys.*, **133**, pp. 84-101 (1997).

[18] N.K. Ghaddar, K. Korczak, B.B. Mikic, and A.T. Patera, "Numerical investigation of incompressible flow in grooved channels. Part 1: Stability and self-sustained oscillations.", *J. Fluid Mech.*, **163**, 99-127 (1986).

[19] D.E. Keyes, Y. Saad, and D.G. Truhlar, "Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering", SIAM (1995).

[20] G.W. Kruse, "Parallel Nonconforming Spectral Element Solution of the Incompressible Navier-Stokes Equations in Three Dimensions," Ph.D. Thesis, Brown University (1997).

[21] V. Girault and P.A. Raviart, "Finite Element Approximation of the Navier-Stokes Equations", Springer (1986).

[22] W.D. Gropp, "Parallel Computing and Domain Decomposition", in *Fifth Conference on Domain Decomposition Methods for Partial Differential Equations*, T.F. Chan, D.E. Keyes, G.A. Meurant, J.S. Scroggs, and R.G. Voigt eds., SIAM, Philadelphia, PA, pp. (1992).

[23] Y. Maday and A.T. Patera, "Spectral element methods for the Navier-Stokes equations", in *State of the Art Surveys in Computational Mechanics*, A.K. Noor ed., ASME, New York, pp. (1989).

[24] Y. Maday, A.T. Patera, and E.M. Rønquist, "An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow", *J. Sci. Comput.*, **5**(4), pp. 310-37 (1990).

[25] Y. Maday, A.T. Patera, and E.M. Rønquist, "The $P_N - P_{N-2}$ method for the approximation of the Stokes problem," *Numer. Math.*, pp. (1987).

[26] S.A. Orszag and L.C. Kells, "Transition to turbulence in plane Poiseuille flow and plane Couette flow", *J. Fluid Mech.*, **96**, pp. 159-205 (1980).

[27] S.A. Orszag, "Spectral methods for problems in complex geometries", *J. Comput. Phys.*, **37**, pp 70-92 (1980).

[28] S.S. Pahl, "Schwarz type domain decomposition methods for spectral element discretizations", Masters Thesis, Dept. of Comput. and Appl. Mathematics, Univ. of the Witwatersrand, Johannesburg, South Africa (1993).

[29] A.T. Patera, "A spectral element method for fluid dynamics; Laminar flow in a channel expansion", *J. Comput. Phys.*, **54**, pp. 468-488 (1984).

[30] 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**(4), pp. (1996).

[31] R.E. Lynch, J.R. Rice, and D.H. Thomas, "Direct Solution of Partial Difference Equations by Tensor Product Methods", *Numerische Mathematik*, **6**, pp. 185-199 (1964).

[32] E.M. Rønquist, "A Domain Decomposition Method for Elliptic Boundary Value Problems: Application to Unsteady Incompressible Fluid Flow", in *Fifth Conference on Domain Decomposition Methods for Partial Differential Equations*, T.F. Chan, D.E. Keyes, G.A. Meurant, J.S. Scroggs, and R.G. Voigt, eds., SIAM, Philadelphia, PA, pp. (1992).

[33] E.M. Rønquist, "A domain decomposition solver for the steady Navier-Stokes equations", in *Proceedings of Intl. Conf. on Spectral and High-Order Methods '95, Houston, TX*, A.V. Ilin and L.R. Scott eds., pp. (1996).

[34] B. Smith, P. Bjørstad, and W. Gropp, "Domain Decomposition", Cambridge University Press, 1996.

[35] G. Strang and G. Fix, *An analysis of the finite element method*, Prentice-Hall, Englewood Cliffs, NJ (1973).

[36] H. M. Tufo and P. F. Fischer, "Fast Parallel Direct Solvers For Coarse Grid Problems", submitted to *J. Parallel Comp* (1997).

[37] O.B. Widlund, "Some Schwarz Methods for Symmetric and Nonsymmetric Elliptic Problems", *in Fifth Conference on Domain Decomposition Methods for Partial Differential Equations*, T.F. Chan, D.E. Keyes, G.A. Meurant, J.S. Scroggs, and R.G. Voigt eds., SIAM, Philadelphia, PA, pp. (1992).

[38] O.B. Widlund. Personal communication (1997).