

# A Pseudo-spectral Scheme for the Incompressible Navier–Stokes Equations Using Unstructured Nodal Elements

T. Warburton,\* L. F. Pavarino,† and J. S. Hesthaven\*

\**Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912; and*

†*Department of Mathematics, Università di Milano, Via Saldini 50, 20133 Milan, Italy*

E-mail: [timw@cfm.brown.edu](mailto:timw@cfm.brown.edu), [pavarino@ares.mat.unimi.it](mailto:pavarino@ares.mat.unimi.it), [Jan\\_Hesthaven@Brown.edu](mailto:Jan_Hesthaven@Brown.edu)

Received November 25, 1999; revised May 31, 2000

---

A pseudo-spectral scheme for solving the incompressible Navier–Stokes equations using unstructured nodal triangles is proposed. Efficient algorithms are developed with numerical evidence that indicates optimal rates of convergence can be achieved. Navier–Stokes simulations of Kovasznay, shear layer roll up, and flow past a cylinder are included to show comparisons between the different nodal sets considered and an alternative modal approach. © 2000 Academic Press

*Key Words:* spectral elements; nodal elements; unstructured meshes; domain decomposition.

---

## 1. INTRODUCTION

There have been a number of recent developments of high-order schemes on unstructured elements (triangles in two dimensions and tetrahedra, prisms, and pyramids in three dimensions). The modal approach employs a local approximation, on the element, that uses a warped tensor product basis based on Jacobi polynomials of variable weight. These have been derived independently in [9, 23, 26], and it is known that these polynomials are the solutions to a singular Sturm–Liouville problem [19, 23, 33, 34] analogous to the one-dimensional problem from which the classical orthogonal polynomials can be derived. Methods based on such modal elements have been shown to be computationally efficient and robust for simulating the Navier–Stokes equations in complex domains [27, 18, 34].

The approach we investigate in this paper likewise uses a compact, element-based representation of the solutions. In each element, however, we choose a set of nodes, construct Lagrange interpolating polynomials associated with these nodes, and use these polynomials to express the approximate solution. The optimal choice of the nodes on which to base the

interpolation is almost an open question for the triangle, as there is no obvious generalization analogous to the quadrilateral where a tensor product of one-dimensional Legendre polynomials is used. A sensible constraint is that the nodes coincide with the one-dimensional Legendre nodes along the boundary of the triangle. This allows the efficient interfacing of triangles and quadrilaterals together in the same mesh, making possible the use of thin quadrilaterals to capture short length scales in boundary layers.

We will consider two sets of nodes. First, we consider a set based on the Fekete principle; i.e., the nodes are chosen to maximize the determinant of the generalized Vandermonde matrix, which we shall define later. The Fekete nodes have been calculated to high order for the triangle [33], and they indicate good approximation properties measured by their Lebesgue constant. Additionally, we will examine the nodes calculated by Hesthaven [15] using an electrostatic principle motivated by a one-dimensional analogy generated between Gauss quadrature points and solutions to electrostatic problems [32].

There are alternative nodal sets for the triangle including a set based on a collapsed coordinate system and a filter [12], but these have a Chebyshev distribution on the boundary. Alternative nodal sets for the triangle and tetrahedron have been obtained by seeking an approximate minimum for the Lebesgue constant [6, 7]. These solutions, however, do not have a known simple distribution along the edges.

In Section 2 we will demonstrate how nodal sets can be integrated into a Galerkin framework and how they can be used to calculate high-order integrals and derivatives of functions. Section 3 is devoted to a brief discussion of how to discretize an elliptic problem and solve it using these nodes, while Section 4 investigates the accuracy of the projections based on these nodes. In Section 5 we discuss a possible preconditioner for the discrete elliptic operator which may be suitable for use in a moving domain code where the operator is time dependent. Section 6 demonstrates how this method can be integrated into a general scheme to solve the incompressible Navier–Stokes equations. Section 7 concludes with a few remarks.

In many instances we will use the terminology of  $h$ -refinement to imply that we are modifying the number and size of elements, while  $p$ -refinement refers to a change in the polynomial order used to represent the fields within each element.

## 2. ELEMENTAL OPERATIONS

### 2.1. *Coordinate Systems*

In the following sections, we will discuss the operators necessary to build a hybrid spectral element code. To form these operators we define a master element for the triangle and one for the quadrilateral. The triangle is defined as the set

$$T = \{(r, s) \mid -1 \leq r, s \leq 1; r + s \leq 0\},$$

and the quadrilateral as

$$Q = \{(r, s) \mid -1 \leq r, s \leq 1\}.$$

We map a point in the master triangle,  $T$ , to a point  $\mathbf{x}$  in the physical straight-sided triangle with the mapping

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

where  $\mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3$  are the spatial coordinates of the vertices of the physical triangle.

Similarly the master quadrilateral,  $Q$ , is mapped to a straight-sided physical quadrilateral with the following mapping:

$$\mathbf{x} = \frac{(1-r)}{2}\frac{(1-s)}{2}\mathbf{v}^1 + \frac{(1+r)}{2}\frac{(1-s)}{2}\mathbf{v}^2 + \frac{(1+r)}{2}\frac{(1+s)}{2}\mathbf{v}^3 + \frac{(1-r)}{2}\frac{(1+s)}{2}\mathbf{v}^4.$$

We return to the general case of curvilinear elements in Section 4.1.

## 2.2. Collocation Projection

To calculate the discrete expansion coefficients for an expansion using the orthogonal Koornwinder/Dubiner polynomials [9, 19] we use the interpolating property of the Lagrange polynomial; i.e., it is unity at a specific node and zero at all other nodes. We can expand any polynomial  $f \in \mathbb{P}^p = \{r^i s^j \mid 0 \leq i + j \leq p\}$  as a linear combination,

$$f(r, s) = \sum_{0 \leq j+k \leq p} \psi_{jk}(r, s) \hat{f}_{jk},$$

of the Koornwinder/Dubiner polynomials defined as

$$\psi_{ij}(r, s) = \sqrt{\frac{(2i+1)(i+j+1)}{2}} P_i^{0,0} \left( -\frac{2(r+s)}{1-s} \right) \left( \frac{1-s}{2} \right)^i P_j^{2i+1,0}(s),$$

where  $P_n^{\alpha,\beta}(x)$  represents the  $n$ th-order Jacobi polynomial. Subsequently we shall use the notation  $\phi_k = \psi_{ij}$  for any member of this basis, where  $k$  represents a unique pair  $(i, j)$ .

Let us now define the Vandermonde matrix as

$$V_{ij} = \phi_j(r_i, s_i),$$

where  $(r_i, s_i)$  is the local coordinate of the  $i$ th node in the master element. If this matrix is sufficiently well conditioned, we can find a unique polynomial representation for a function whose values are known at the nodal positions, as

$$\hat{f}_i = \sum_{0 \leq j < N_p} V_{ij}^{-1} f(r_j, s_j)$$

$$N_p = (p+1)(p+2)/2,$$

where the coefficients  $\hat{f}_i$  are represented in the same order as the basis functions. The conditioning of the Vandermonde matrix,  $\mathbf{V}$ , is determined by the choice of the basis it is formed with and the set of nodes at which this basis is evaluated. The orthonormal basis is a natural choice over the simpler monomial basis as it is designed to maintain linear independence. This slightly increases the complexity of constructing  $\mathbf{V}$  but dramatically improves the conditioning.

Recently there have been a number of efforts directed towards finding numerically stable sets of collocation points for the triangle [1, 2, 6, 15, 33]. We will focus on two sets of nodes. The first is the set of Fekete collocation points calculated by Taylor and Wingate [33]. These nodes are defined as the set of nodes which maximise the determinant of the Vandermonde matrix. The second set was derived by Hesthaven [15] using an analogy between the zeros of Jacobi polynomials and the solution of an electrostatics problem. In Figs. 5 and 6, we show a Delaunay triangulation [30] of these nodes on the master triangle for polynomial orders up to  $p = 12$ .

### 2.3. Differentiation

Given a set of values of a function at the nodal positions, we can evaluate the derivatives by exploiting the interpolating polynomial. Hence the  $x$  derivative of a function  $f$  is obtained by first calculating its polynomial coefficients and then differentiating the polynomial approximation and evaluating at the nodes as

$$\begin{aligned} \frac{\partial f}{\partial x}(r_i, s_i) &= \sum_{0 \leq j < N_p} (r_x(r_i, s_i)D_{ij}^r + s_x(r_i, s_i)D_{ij}^s) f(r_j, s_j) \\ D_{ij}^r &= \frac{\partial L_j}{\partial r}(r_i, s_i) \\ D_{ij}^s &= \frac{\partial L_j}{\partial s}(r_i, s_i), \end{aligned}$$

where  $L_j$  is the Lagrange interpolating function which is unity at the  $j$ th node and zero at all other nodes.  $D_{ij}^r$  is the evaluation of the derivative with respect to  $r$  of the  $j$ th Lagrange interpolating function at the  $i$ th node, and  $r_x, s_x$  are the derivatives of the master element coordinates with respect to the physical coordinate  $x$ . This is exact for all  $f \in \mathbb{P}^p$  by the uniqueness of the interpolation.

In practice we precalculate the matrices  $\mathbf{D}^r$  and  $\mathbf{D}^s$  using the identity

$$\begin{aligned} D_{ij}^r &= \sum_{0 \leq k < N_p} \frac{\partial \phi_k}{\partial r}(r_i, s_i) V_{kj}^{-1} \\ D_{ij}^s &= \sum_{0 \leq k < N_p} \frac{\partial \phi_k}{\partial s}(r_i, s_i) V_{kj}^{-1}. \end{aligned}$$

These matrices can be used for all straight-sided triangles of the same polynomial order. Using these, it is straightforward to calculate the  $x$  and  $y$  derivatives of the function. The derivatives of the  $\phi$  basis functions can be calculated using the identity

$$\frac{d}{dx} P_n^{\alpha,0}(x) = \frac{1}{2}(\alpha + n + 1)P_{n-1}^{\alpha+1,1}(x). \quad (1)$$

### 2.4. Inner Products

It will be necessary to evaluate inner products of the form

$$(u, v) = \int_{-1}^1 \int_{-1}^{-s} f(r, s)g(r, s)J(r, s) dr ds,$$

where  $f, g \in \mathbb{P}^p$  and  $J(r, s)$  is the transformation Jacobian associated with the mapping between  $(r, s)$  and  $(x, y)$ .

For the moment we will consider the case where the Jacobian ( $J$ ) is independent of  $r$  and  $s$ . This will be true for all straight-sided triangles. For quadrilateral elements, integrals of this type are usually calculated using a quadrature rule based on a tensor product of Gauss–Lobatto–Legendre quadrature points and weights. Using a  $p$ th-order quadrature exactly integrates the product  $f \cdot g \in P^{2p-1}$ . However, there is no obvious analogous set of weights for the Fekete or electrostatic nodes that will achieve the same level of accuracy. Taylor and Wingate [33] proposed a set of weights that will integrate functions up to total polynomial order  $p$ . This is useful in certain circumstances but is not sufficiently accurate to evaluate the above integral.

We propose to replace the usual set of weights with a semi-analytical approach to calculate the inner products in which we use the collocation projection described previously to project both the functions  $f$  and  $g$  to the orthogonal Koornwinder/Dubiner polynomials. The orthogonality of the basis functions allows us to simply take the dot product of the two coefficient vectors to calculate the inner product of the two polynomials

$$(f, g) = J \sum_{i=0}^{N_p-1} \sum_{j=0}^{N_p-1} \sum_{k=0}^{N_p-1} f_i (V_{ji}^{-1} V_{jk}^{-1}) g_k,$$

where  $\mathbf{V}$  is the Vandermonde matrix evaluated with respect to the orthonormal Koornwinder/Dubiner basis. The inner product is exact for all  $f, g \in P^p$  if the triangle is straight-sided. In practice, the matrix  $(\mathbf{V}^{-1})^T (\mathbf{V}^{-1})$  is calculated in preprocessing.

### 3. CONTINUOUS AND DISCRETE ELLIPTIC PROBLEMS

We consider the following model elliptic problem on a bounded Lipschitz region  $\Omega \subset \mathbb{R}^d$  with boundary  $\partial\Omega = \Gamma_D \cup \Gamma_N$ :

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

Dirichlet boundary conditions are imposed on  $\Gamma_D$ , a closed subset of  $\partial\Omega$  with positive measure, and Neumann conditions on  $\Gamma_N$ . More general linear, self-adjoint, second-order elliptic problems and boundary conditions could be considered as well. The standard variational formulation of this problem is:

Find  $u \in V = H_D^1(\Omega) = \{v \in H^1(\Omega): v = 0 \text{ on } \Gamma_D\}$  such that

$$a(u, v) = F(v), \quad \forall v \in V, \quad (2)$$

where

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

We assume the domain  $\Omega$  is a union of the spectral elements described previously

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

where each  $\Omega_k$  is the affine image of the reference triangle or square. Let  $\tau_K$  be the mesh defined by the spectral elements  $\Omega_k$ , and the spectral element space being defined as

$$V^{p,K} = \{v \in V : v|_{\Omega_k} \in \mathbb{P}^p, k = 1, \dots, K\}.$$

For quadrilaterals

$$\mathbb{P}^p = \{x^i y^j \mid 0 \leq i, j \leq p\},$$

and for triangles

$$\mathbb{P}^p = \{x^i y^j \mid 0 \leq i + j \leq p\}.$$

The standard Galerkin formulation of (2) is:

Find  $u \in V^{p,K}$  such that

$$a_{p,K}(u, v) = F_{p,K}(v) \quad \forall v \in V^{p,K}, \quad (3)$$

where  $a_{p,K}(\cdot, \cdot)$  and  $F_{p,K}(\cdot)$  are obtained from  $a(\cdot, \cdot)$  and  $F(\cdot)$  by using the integration previously described in Section 2.4.

The stiffness matrix  $A$  and load vector  $f$  of this discrete system are assembled from their elemental contributions on each  $\Omega_k$  by means of the  $Z$  operator, described in detail in [13], which assembles the local coefficients into the global coefficients and ensures  $C^0$  continuity.

There are numerous approaches to solving the resulting discrete system  $Ax = f$ . Besides direct methods, which can be very expensive and far from optimal for large-scale problems, common approaches are based on the iterative solution of this system by a preconditioned Krylov subspace method such as PCG. The preconditioners range from simple diagonal scaling or incomplete factorizations of  $A$  to more efficient domain decomposition methods, based on overlapping or iterative substructuring techniques. This latter class of algorithms is based on condensing out the nodes on the boundary of the element, solving a Schur complement system for the boundary nodes, and then solving local problems for the interior degrees of freedom [28]. In other words, by ordering the boundary nodes  $x_B$  and then the interior nodes  $x_I$ ,  $x = (x_B, x_I)$ , the system  $Ax = f$  is rewritten as

$$\begin{bmatrix} A_B - A_{BI}^T A_I^{-1} A_{BI} & 0 \\ A_{BI} & A_I \end{bmatrix} \begin{bmatrix} x_B \\ x_I \end{bmatrix} = \begin{bmatrix} f_B - A_{BI}^T A_I^{-1} f_I \\ f_I \end{bmatrix}.$$

The Schur complement system

$$Sx_B = (A_B - A_{BI}^T A_I^{-1} A_{BI})x_B = \tilde{f}_B = f_B - A_{BI}^T A_I^{-1} f_I \quad (4)$$

is solved iteratively by a preconditioned conjugate gradient method. Again, the preconditioners range from diagonal scaling to domain decomposition methods; see [25, 28].

#### 4. ACCURACY OF PROJECTION OPERATOR

We tested the accuracy of the nodal triangles for solving the elliptic problem by performing a  $p$ - and  $h$ -refinement study. The continuous problem is

$$\begin{aligned} (-\nabla^2 + 1)u &= (1 + 2\pi^2) \sin(\pi x) \sin(\pi y) & \text{in } \Omega = [-1, 1] \times [-1, 1], \\ u &= \sin(\pi x) \sin(\pi y) & \text{on } \Gamma_D = \partial\Omega. \end{aligned} \quad (5)$$

The domain is covered with  $K = 2M^2$  triangles, and the rate of convergence, for fixed  $p$ , is calculated as

$$\text{rate} = \frac{\log(\text{error}_{M+1}/\text{error}_M)}{\log(M/(M+1))}.$$

Table I shows the results using the Fekete points and Table II shows the results of the same study using the electrostatic points. Both sets of points achieve exponential convergence to the exact solution, with approximately  $(p+1)$ th-order accuracy. Figure 1 shows log-log plots, demonstrating the expected similarity between the errors from both methods. The slight differences for the errors near machine precision are caused by finite precision effects. The  $L_\infty$  norm was calculated using the values of the fields at the nodes; i.e. the locations of the sampling for this norm depend on the choice of basis.

##### 4.1. Accuracy of Projection on Curved Elements

To represent a curved boundary, it is convenient to use an isoparametric mapping of the coordinate system. In this work we use standard Hall blending techniques to curve the triangular elements [11].

In preprocessing we use a product-rule quadrature, on a collapsed coordinate system for the triangle [18], to calculate the mass matrices and stiffness matrices specific to each curved element. This allows specification of the order of accuracy of the spatial integration without being bound by the  $p$ th-order projection that causes aliasing for the product of the integrands, the geometric factors, and the Jacobian. For the solution of Eq. 5, on the domains shown in Fig. 2, with this approach we achieve, as seen in Fig. 3, exponential convergence at a suboptimal rate of just over  $p$ .

**TABLE I**  
Rate of  $p$  Convergence for the Elliptic Problem Using Fekete Nodes

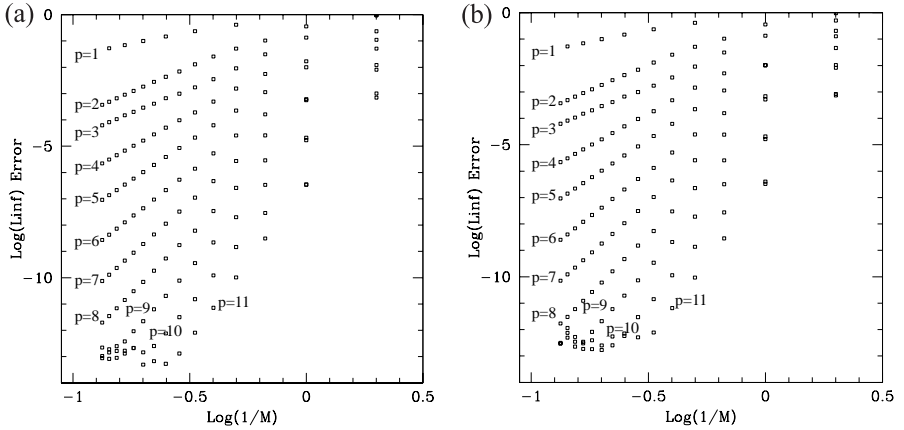
$p$	$M = 3$	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$M = 8$	$M = 9$	$M = 10$	$M = 11$
2	2.47	3.10	3.72	4.04	3.50	3.77	3.99	3.97	3.78
3	4.28	4.25	3.91	3.58	2.93	4.11	3.30	3.85	3.47
4	4.46	5.08	5.24	4.43	4.87	4.95	5.20	4.70	4.93
5	5.62	5.78	5.86	5.90	5.93	5.95	5.96	5.97	5.97
6	6.43	7.04	7.35	6.24	6.80	7.04	7.26	6.60	6.89
7	7.79	7.88	7.92	7.94	7.95	7.96	7.97	7.98	7.98
8	8.38	9.08	9.39	8.21	8.78	9.07	9.28	8.57	8.89
9	9.82	9.90	9.93	9.95	9.96	9.98	9.98	9.68	10.19
10	10.36	11.12	11.43	10.16	10.25	10.39	9.32	**	**
11	11.84	11.91	12.01	11.67	9.75	**	**	**	**
12	12.46	12.96	11.26	**	**	**	**	**	**

\*\* Indicates that the solution has already converged to the iterative tolerance at lower resolution.

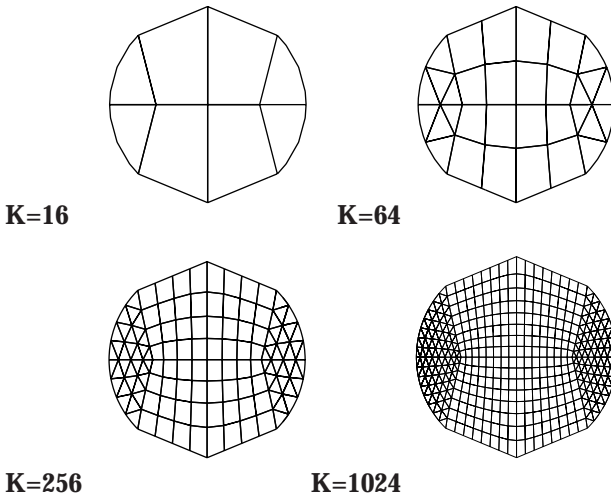
**TABLE II**  
**Rate of  $p$  Convergence for the Elliptic Problem Using the Electrostatic Nodes**

$p$	$M = 3$	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$M = 8$	$M = 9$	$M = 10$	$M = 11$
2	2.47	3.10	3.72	4.04	3.50	3.77	3.99	3.97	3.78
3	4.28	4.25	3.91	3.58	2.93	4.11	3.30	3.85	3.47
4	4.36	4.92	5.24	4.39	4.81	4.97	5.21	4.68	4.89
5	5.68	5.71	5.83	5.86	5.91	5.92	5.95	5.95	5.97
6	6.44	7.06	7.38	6.26	6.82	7.07	7.25	6.61	6.90
7	7.79	7.87	7.91	7.94	7.95	7.96	7.97	7.97	7.98
8	8.42	9.09	9.41	8.23	8.79	9.08	9.29	8.58	8.88
9	9.84	9.90	9.93	9.95	9.97	9.97	9.98	9.93	8.99
10	10.42	11.11	11.43	10.21	10.68	**	**	**	**
11	11.87	11.94	11.09	**	**	**	**	**	**
12	12.37	12.04	**	**	**	**	**	**	**

\*\* Indicates that the solution has already converged to the iterative tolerance at lower resolution.

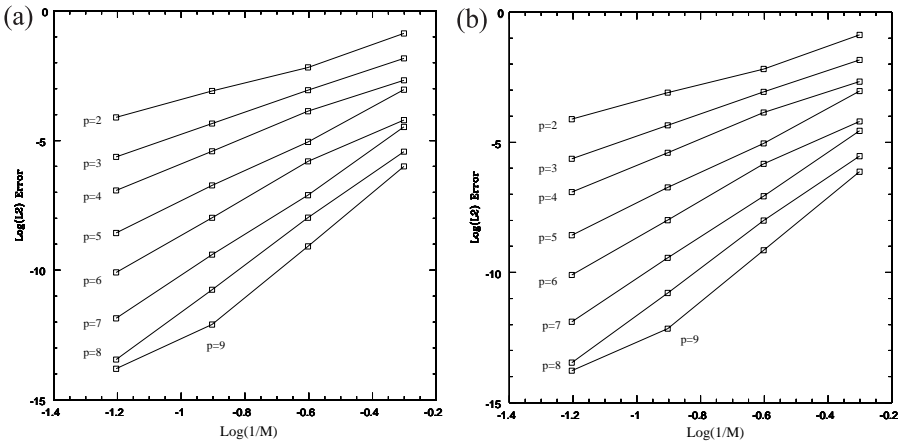


**FIG. 1.** An  $h$  and  $p$  convergence study for solution of the elliptic problem. (a) Fekete nodes. (b) Electrostatic nodes.



**FIG. 2.** Curved triangles used for  $h$ - and  $p$ -refinement tests.





**FIG. 3.** Convergence of  $L_2$  error for the standard elliptic problem on a curved domain using mass matrices calculated with quadrature. (a) Using Fekete nodes; (b) using electrostatic nodes.

## 5. $h$ -TYPE PRECONDITIONERS

The benefit of efficient, finite-element-based preconditioning for quadrilateral and hexahedral spectral elements [4, 8, 22, 24] is part of the motivation for using nodal triangles. The original method involves constructing a finite-element operator based on triangulating the GLL nodes of the spectral elements. The resulting operator is spectrally equivalent to the true spectral element operator and the preconditioner is hence quasi-optimal. The finite-element-based preconditioner is more readily approximately inverted using multigrid [31], ILU [21], or overlapping additive Schwarz [10] methods.

In this section we will experimentally determine whether similar results on the spectral equivalence hold for the nodal triangle, and we will show that the results do not hold for the Fekete or electrostatic choice of nodes. We have found that the condition number for the discrete elliptic operator  $a_{p,K}$ , preconditioned with the  $h$ -type preconditioner, scales linearly with the order of nodes used. This indicates that this preconditioner is no longer spectrally equivalent to the elliptic operator and thus will not be as efficient as for the quadrilateral case.

### 5.1. Fekete-Based $h$ -Type Preconditioner

To understand the spectral behaviour of a finite-element-based preconditioner we consider the solution of the Helmholtz equation in a square of length 2, covered in a regular array of  $M \times M$  squares each split into two triangular elements. A Delaunay triangulation [30] of the nodes is used to form a finite-element mesh as shown in Fig. 4. We see that for  $p = 4$  to  $p = 7$  this mesh is quite regular, while for a larger  $p$  we find that the underlying grid can be very irregular. The irregularity can be seen in the widely varying aspect ratios of the triangulation.

The results in Table III show the dependence of the condition number of the preconditioned operator on the polynomial order and grid resolution. It is clear that the preconditioner removes any dependence on the size of elements, but there remains a linear dependence on the polynomial order.

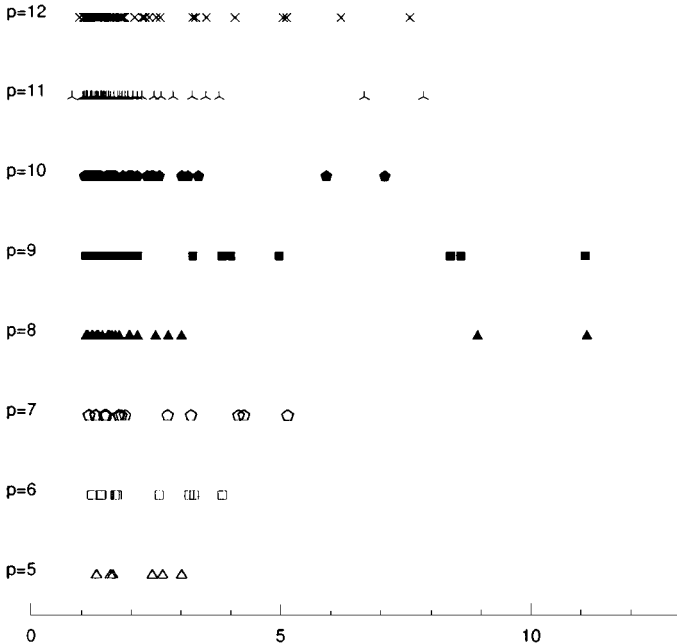


**TABLE IV**  
**Range of Angles and Edge Lengths of the Fekete- and Electrostatic-Based Finite-Element Mesh as a Function of  $p$  for  $M = 3$**

$p$	Min angle		Max angle		Min length		Max length	
	Fekete	Electro	Fekete	Electro	Fekete	Electro	Fekete	Electro
1	45	45	90	90	0.666	0.666	0.942	0.942
2	45	45	90	90	0.333	0.333	0.471	0.471
3	32.19	32.19	106.60	106.60	0.184	0.184	0.421	0.421
4	30.18	26.11	116.56	107.66	0.115	0.115	0.330	0.308
5	27.97	27.10	122.29	115.78	0.078	0.078	0.268	0.268
6	25.08	26.50	129.82	123.31	0.056	0.056	0.235	0.232
7	22.47	17.44	133.79	132.13	0.042	0.042	0.200	0.209
8	8.21	14.47	147.93	135.75	0.029	0.029	0.228	0.183
9	18.70	12.96	141.31	137.61	0.026	0.020	0.161	0.164
10	7.13	11.04	151.75	139.85	0.017	0.014	0.185	0.146
11	6.02	9.70	158.75	141.45	0.018	0.010	0.197	0.133
12	6.26	6.32	157.98	143.76	0.012	0.007	0.166	0.121

### 5.2. Electrostatic Based $h$ -Type Preconditioner

We have repeated the computation of condition number for the elliptic operator preconditioned by the  $h$ -type preconditioner based on the electrostatic nodes. The finite-element mesh derived from the nodes is shown in Fig. 6 and the  $h$ - and  $p$ -dependence of the



**FIG. 5.** Real eigenvalues of the elliptic operator on a one element triangular domain, preconditioned with the Fekete-based finite-element preconditioner.

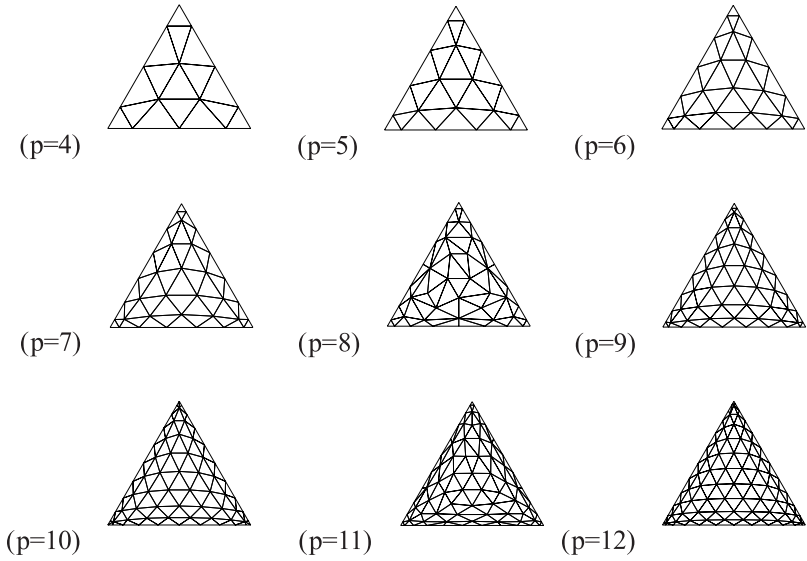


FIG. 6. Electrostatic-node-based finite-element mesh for a reference element.

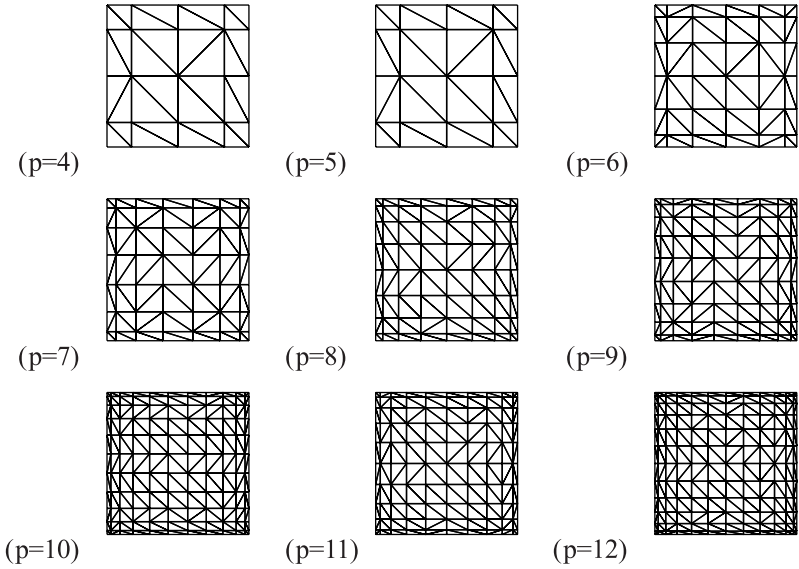
condition number of the preconditioned operator is shown in Table V. For this choice of nodes we find that the condition number grows linearly up to  $p = 6$ . For larger polynomial order the condition number is found to grow faster. In Table IV we see that the minimum edge length in the electrostatic meshes decreases rather more rapidly than for the Fekete mesh. We conjecture that this directly affects the scaling of the finite-element-based operator.

In Fig. 7 we show the eigenvalues for the preconditioned operator for a problem in which the reference triangle acts as the total domain. As we saw with the Fekete version, the eigenvalues are real, and most of the eigenvalues cluster around the interval 1 to 5. There

TABLE V  
The  $h$ - and  $p$ -Dependence of Condition Number of Electrostatic-Type  $h$ -Preconditioned Elliptic Operator ( $h = 2/M$ )

$p$	$M = 3$	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$M = 8$	$M = 9$	$M = 10$	$M = 11$	$M = 12$
1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	1.79	1.87	1.71	1.93	1.83	1.96	1.89	1.92	1.90	1.90
3	2.61	2.64	2.62	2.61	2.60	2.59	2.59	2.58	2.58	2.58
4	3.78	3.75	3.75	3.76	3.76	3.76	3.76	3.76	3.76	3.76
5	4.61	4.65	4.64	4.65	4.65	4.64	4.64	4.64	4.63	4.63
6	6.67	6.74	6.77	6.78	6.79	6.79	6.78	6.78	6.78	6.78
7	9.56	9.62	9.50	9.68	9.61	9.67	9.66	9.66	9.66	9.66
8	14.94	15.09	15.14	15.18	15.20	15.21	15.22	15.22	15.23	15.23
9	23.95	24.16	24.25	24.29	24.30	24.31	24.28	24.30	24.25	24.27
10	44.75	45.02	45.12	45.18	45.21	45.23	45.24	45.25	45.26	45.26
11	81.51	82.17	82.45	82.59	82.66	82.71	82.74	82.76	82.77	82.77
12	150.93	151.87	152.32	152.58	152.74	152.85	152.93	152.99	153.03	153.06





**FIG. 8.** Quadrilateral GLL-node-based finite-element mesh for a reference element.

that the condition number does not grow and experimentally validate the theory in [5], which states that the condition number is bounded independent of  $h$ - and  $p$ -refinement. In Fig. 8 we show the Delaunay triangulation created using the Gauss–Lobatto–Legendre nodes.

## 6. INCOMPRESSIBLE NAVIER–STOKES SIMULATIONS

The two-dimensional incompressible Navier–Stokes equations are

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} &= -\nabla P + \nu \nabla^2 \mathbf{v} + \mathbf{F}, \\ \nabla \cdot \mathbf{v} &= 0, \end{aligned}$$

where  $\mathbf{v}$  denotes the velocity of the fluid with components  $\mathbf{v} = [u(x, y, t), v(x, y, t)]^T$  in the  $x$  and  $y$  directions,  $P(x, y, t)$  is the pressure,  $\mathbf{F}(x, y, t)$  is a forcing function, and  $\nu$  is the kinematic viscosity. To discretize these equations in time we use a high-order splitting scheme [17] involving four substeps,

$$\tilde{\mathbf{v}} = \sum_{q=0}^{J_e-1} \alpha_q \mathbf{v}^{n-q} + \Delta t \left( \sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{v}^{n-q}) + F^{n+1} \right) \quad (6)$$

$$\frac{\partial \bar{P}^{n+1}}{\partial n} = -\mathbf{n} \cdot \left[ \sum_{q=0}^{J_e-1} \beta_q (\mathbf{N}(\mathbf{v}^{n-q}) + \nu \nabla \times (\nabla \times \mathbf{v}^{n-q})) \right] \quad (7)$$

$$\nabla^2 \bar{P}^{n+1} = \nabla \cdot \left( \frac{\tilde{\mathbf{v}}}{\Delta t} \right) \quad (8)$$

$$\nabla^2 \mathbf{v}^{n+1} - \frac{\gamma_0}{\nu \Delta t} \mathbf{v}^{n+1} = -\frac{1}{\nu \Delta t} (\tilde{\mathbf{v}} - \Delta t \nabla \bar{P}^{n+1}), \quad (9)$$

where  $n$  is the time step number,  $\Delta t$  is the size of the time step,  $J_i$  is the order of time integration for the diffusive term,  $J_e$  is the order of the advection term, and the coefficients  $(\alpha_q, \beta_q, \gamma_0)$  are listed in [18].

The first step is performed using the values of the fields  $(u, v, \bar{P})$  at the nodal points associated with each element. This will clearly involve some aliasing in the nonlinear term  $\mathbf{N}(\mathbf{v})$ . However, experience with the existing PRISM [13] and  $\mathcal{N}\varepsilon\kappa\mathcal{T}\alpha r$  [29] codes indicates that this is not typically a significant problem.

The second step is to evaluate the value of the pressure Neumann conditions associated with the Dirichlet velocity conditions. Again this is calculated using the nodal values of the  $(u, v)$  fields. The third step involves solving the linear system of equations described in Section 3. Here, the right-hand side is first evaluated using the derivative operators described in Section 2.3. The fourth step is similar to the third step and involves the solution of two independent equation systems, one for each velocity component.

### 6.1. Numerical Validations

For the following test cases we used Cholesky factorization on the pressure and velocity systems to factorize the Schur complement from Eq. (4) at the start of the computation. Direct solvers were also used for each element to solve the decoupled interior–interior node systems.

*6.1.1. Kovaszny flow.* The first Navier–Stokes solution we shall consider is the Kovaszny flow. This is a laminar flow behind a two-dimensional grid, the exact solution of which is due to Kovaszny [20]. This solution can be written as a function of Reynolds number  $\text{Re}$  in the form

$$\begin{aligned} u(x, y) &= \frac{\lambda}{2\pi} e^{\lambda y} \sin(2\pi x) \\ v(x, y) &= 1 - e^{\lambda y} \cos(2\pi x), \end{aligned}$$

where

$$\lambda = \frac{\text{Re}}{2} - \left( \frac{\text{Re}^2}{4} + 4\pi^2 \right)^{\frac{1}{2}}.$$

Using the exact solution as Dirichlet boundary conditions, a steady-state solution was obtained using the discretizations shown in Fig. 9. Using the exact solution allows us to calculate the  $L_2$  error for increasing expansion order as shown in Fig. 10. The rates of convergence for the two nodal sets are given in Tables VII and VIII. It is clear that the rate of convergence is close to  $p + 1$  up to the time stepping error which is of the order  $\Delta t^3$ . In this case  $\Delta t = 0.001$  and the convergence cut off is at  $O(10^{-9})$ .

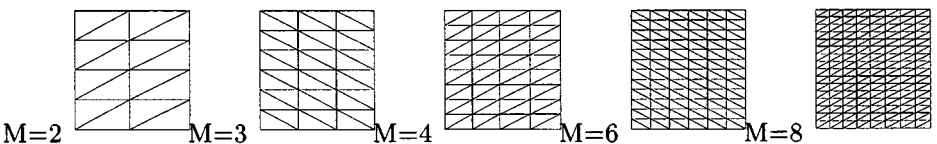


FIG. 9. Meshes used in  $h$ - and  $p$ -refinement studies of the Kovaszny problem.

**TABLE VII**  
**Rate of  $p$  Convergence, in the  $L_2$  Norm, for the Kovaszny**  
**Problem ( $Re = 40$ ) Using Fekete Nodes**

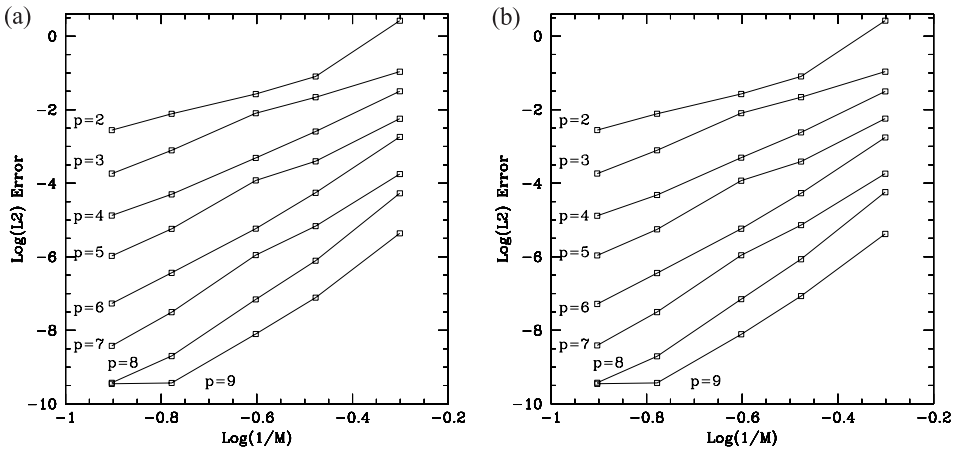
$p$	Mesh			
	$M = 2$	$M = 3$	$M = 4$	$M = 6$
2	8.61	3.80	3.04	3.55
3	3.93	3.49	5.73	5.05
4	6.21	5.74	5.62	4.60
5	6.57	4.16	7.48	5.87
6	8.62	7.81	6.81	6.67
7	8.03	6.30	8.81	7.33
8	10.42	8.41	8.74	5.81
9	9.93	7.91	7.56	**

\*\* Indicates that the solution has already converged to the time stepping error.

**TABLE VIII**  
**Rate of  $p$  Convergence, in the  $L_2$  Norm, for the Kovaszny**  
**Problem ( $Re = 40$ ) Using Electrostatic Nodes**

$p$	Mesh			
	$M = 2$	$M = 3$	$M = 4$	$M = 6$
2	8.61	3.80	3.04	3.55
3	3.93	3.49	5.73	5.05
4	6.32	5.51	5.76	4.52
5	6.62	4.16	7.50	5.68
6	8.59	7.74	6.84	6.72
7	7.94	6.54	8.78	7.25
8	10.34	8.71	8.81	5.74
9	9.58	8.31	7.51	**

\*\* Indicates that the solution has already converged to the time stepping error.



**FIG. 10.** Convergence in the  $L_2$  norm as a function of polynomial order for the steady-state Kovaszny flow at a Reynolds number  $Re = 40$ . (a) Using Fekete nodes; (b) using electrostatic nodes.



*6.1.2. Double shear layer flow.* The next test is similar to cases considered by Brown and Minion [3]. It consists of an initial value problem in a periodic box of length 2. Two thin shear layers are perturbed and subsequently roll up into two vortices with trailing arms. The initial condition is

$$\begin{aligned} u &= \tanh(\epsilon(y + 0.5)) & \text{for } y \leq 0 \\ &= \tanh(\epsilon(0.5 - y)) & \text{for } y > 0. \\ v &= \delta \cos(\pi x) \\ \epsilon &= 40.0 \\ \delta &= 0.05 \end{aligned}$$

We used this highly nonlinear flow to test the potential aliasing properties of the nodal triangle and compare the results with those using the modal triangle. In Fig. 11 we show comparisons, at time 1.87, of the vorticity field for the simulation run with  $p = 9$ ,  $p = 11$ ,  $p = 14$ , and  $p = 18$  on a mesh of  $12 \times 12 \times 2$  triangle elements. The plots on the left show the nodal results, and the plots on the right show the modal results.

This test is a good indication of the effects of underresolution. In Fig. 11a we see that four spurious vortices have been created due to lack of resolution. In contrast the modal version in Fig. 11b has created several weaker spurious vortices, and the roll up of the arms has been distorted. Both these cases confirm that lack of resolution can cause dubious physical phenomena. It is interesting to note that the different types of approximation have markedly different properties in this regime. It is difficult to mark either case as being more accurate.

Comparing Figs. 11c and 11d we see that for  $p = 11$  the latter, modal version, still exhibits a spurious vortex on the trailing arm of the lower vortex. The nodal version (c) shows no strong spurious vortices, but still has some residual oscillations on the trailing arms.

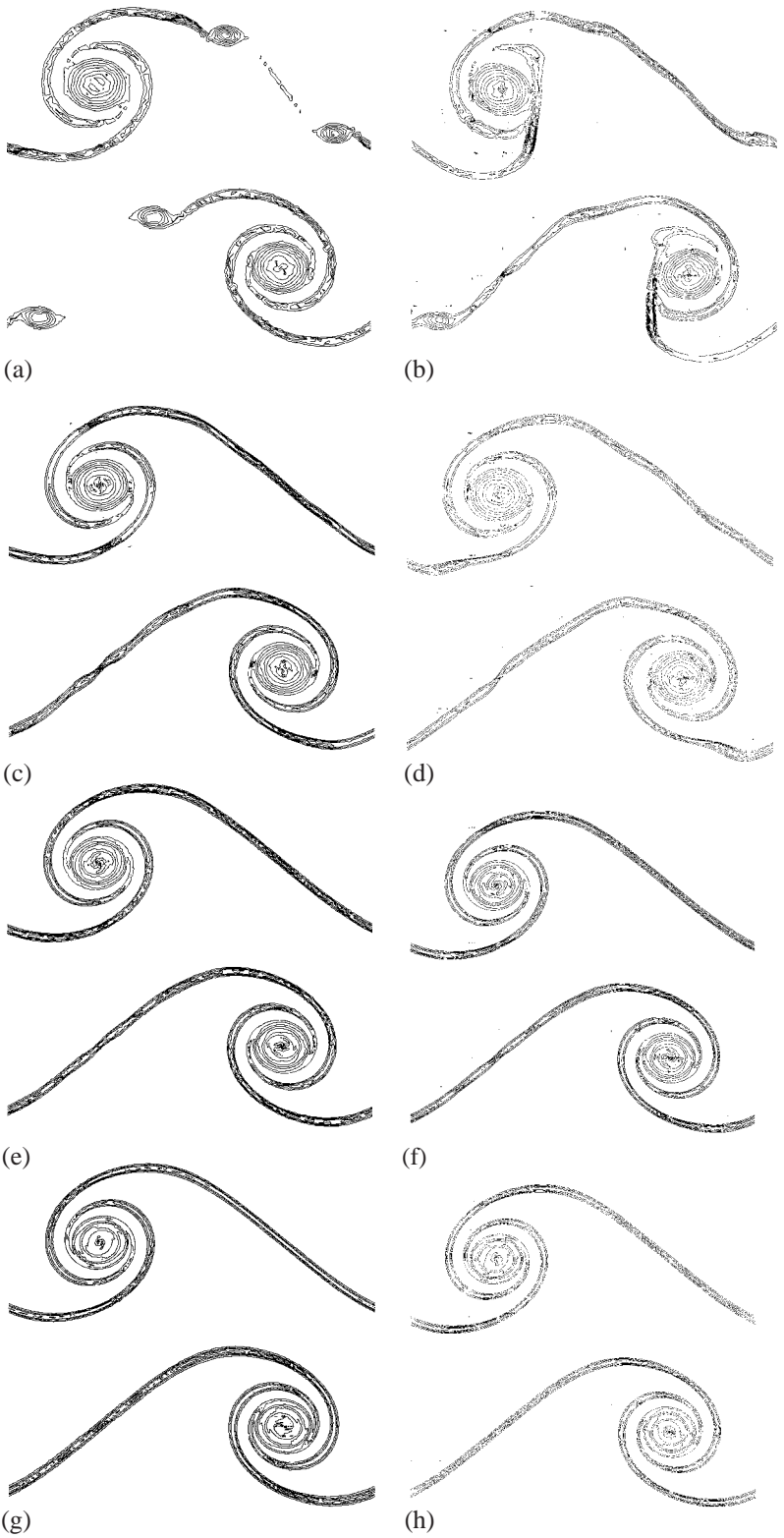
Increasing the resolution more, as shown in Figs. 11e–11h, clearly removes these anomalous vortices and restores the correct symmetries in the solutions. It appears from these results that aliasing causes no greater problem for the nodal triangle than for the modal triangle.

## 6.2. Flow Past a Cylinder

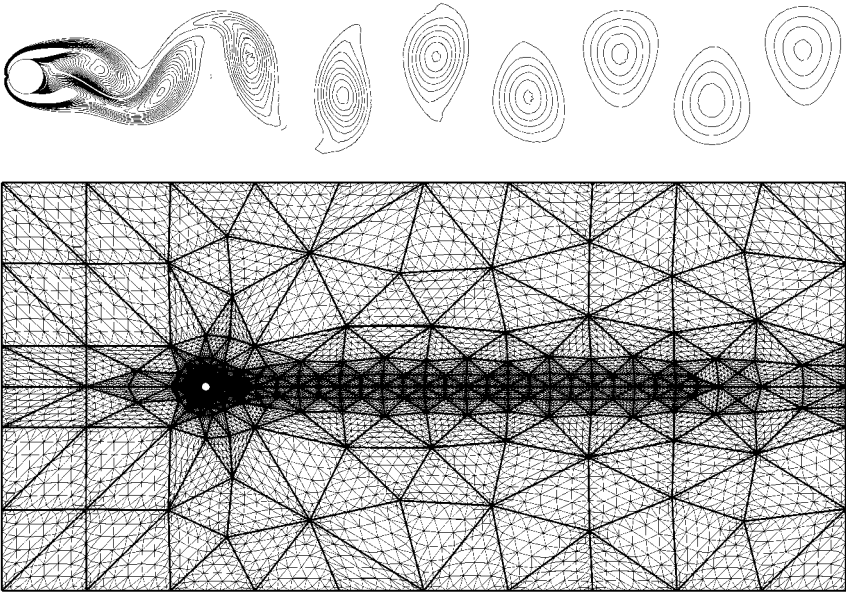
Flow past a cylinder provides a good way to verify an unsteady Navier–Stokes code. For  $\text{Re} \geq 40$  vortex shedding occurs at the cylinder and a von Karman street of vortices forms in the wake of the cylinder (Fig. 12). This shedding process causes the forces on the cylinder to oscillate with a distinct frequency, known as the Strouhal frequency ( $\text{St}$ ). The Strouhal frequency will be used as a measure to compare the results from this and other codes.

For Reynolds numbers up to approximately 190 the flow remains two-dimensional. Above this number three-dimensional instabilities occur, causing the two-dimensional approximation to be increasingly inaccurate for higher  $\text{Re}$ .

For this test of the nodal triangle we consider two-dimensional flow past a circular cylinder. The cylinder has unit diameter, and the domain surrounding the cylinder is a rectangle  $[-22, 69] \times [-22, 22]$ . Uniform velocity boundaries are used at the inflow, upper and lower boundaries. Zero Neumann boundary conditions are used for velocity and the pressure is set to zero at the outflow.



**FIG. 11.** Vorticity contours for the double shear layer flow ( $Re = 10,000$ ,  $t = 1.87$ ). Fekete nodes: (a)  $p = 9$ , (c)  $p = 11$ , (e)  $p = 14$ , (g)  $p = 18$ . Modified Dubiner basis: (b)  $p = 9$ , (d)  $p = 11$ , (f)  $p = 14$ , (h)  $p = 18$ .



**FIG. 12.** Top: Vorticity contours for flow past a cylinder ( $\text{Re} = 100$ ,  $p = 9$ , 490 elements). Bottom: Full mesh with thick lines showing element boundaries and thin lines showing triangulation of nodes.

In Table IX we compare the results for Strouhal frequency and drag coefficient  $C_d$  from the modal element code,  $\mathcal{N}_{\varepsilon\kappa}\mathcal{T}\alpha r$  [29, 34], and the nodal quadrilateral code Prism [13] for a  $\text{Re} = 100$  simulation. We see that very good agreement is reached between each of the different approaches for discretizing the domain.

## 7. SUMMARY

We have demonstrated a straight-forward algorithm for using unstructured nodal spectral elements. For straight-sided triangles we have shown that it is not necessary to use a set of Gauss-like weights for volume integrals. The nodal triangles achieve exponential convergence with an optimal rate of  $(p + 1)$ th-order accuracy. For curved triangles we have shown that it is sufficient to introduce a high-order quadrature or cubature to achieve a suboptimal rate of  $p$ th-order accuracy.

We have shown that there is not much difference between the accuracies using the Fekete or electrostatic nodes with this algorithm. However, it is more difficult to precondition

**TABLE IX**

**Resolution of Strouhal Frequency (St) and Drag Coefficient (Cd) by Nodal Triangles, Modal Triangles, Reference High-Resolution Modal Hybrid Grid, and Nodal Quadrilaterals (Prism)**

Code	Resolution	St	Cd
Nodal code	$K = 490$ , $p = 9$	0.1662	1.3447
Modal hybrid $\mathcal{N}_{\varepsilon\kappa}\mathcal{T}\alpha r$	$K = 490$ , $p = 9$	0.1661	1.3446
Modal hybrid $\mathcal{N}_{\varepsilon\kappa}\mathcal{T}\alpha r$	$K = 780$ , $p = 11$	0.1662	1.3447
Prism [14]	14000 dof	0.1664	1.3500

elliptic solves using a finite-element preconditioner based on the electrostatic nodes than when using the Fekete nodes.

The algorithm has been demonstrated on three incompressible Navier–Stokes simulations, showing that  $(p + 1)$ th-order accuracy is attained for Kovasznay flow and good agreement is achieved between modal and nodal triangles for well-resolved simulations of thin shear layers. Third, close agreement is reached between nodal triangles, modal triangles, modal hybrid elements, and nodal quadrilateral elements for flow past a cylinder.

In future work we intend to investigate the use of the rotational symmetries, as suggested in [16], in the nodal sets to improve the efficiency of the inner product and derivative operators at higher polynomial orders. We also intend to investigate the use of overlapping preconditioners to accelerate the elliptic solves.

### ACKNOWLEDGMENTS

Tim Warburton and Jan S. Hesthaven were supported under the following grants: DARPA F49620-96-1-0426 and DOE DE-FG02-98ER25346. Luca Pavarino was supported by Murst and by the National Science Foundation under Grant NSF CTS-9417520.

### REFERENCES

1. L. Bos, Bounding the Lebesgue function for Lagrange interpolation in a simplex, *J. Approx. Theor.* **38**, 43 (1983).
2. L. Bos, On certain configurations in  $\mathbf{R}^n$  which are unisolvent for polynomial interpolation, *J. Approx. Theor.* **64**, 271 (1991).
3. D. L. Brown and M. L. Minion, Performance of under-resolved two-dimensional incompressible flow simulations, *J. Comput. Phys.* **121** (November 1995).
4. C. Canuto, Stabilization of spectral methods by finite element bubble functions, *Comp. Meth. Appl. Mech. Eng.* **116**, 13 (1994).
5. M. Casarin, Quasi-optimal Schwarz methods for the conforming spectral element discretization, *SIAM J. Numer. Anal.* **34**, 2482 (1997).
6. Q. Chen and I. Babuška, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle, *Comp. Meth. Appl. Mech. Eng.* **128**, 405 (1995).
7. Q. Chen and I. Babuška, The optimal symmetrical points for polynomial interpolation of real functions in the tetrahedron, *Comp. Meth. Appl. Mech. Eng.* **137**, 89 (1996).
8. M. Deville and E. Mund, Finite element preconditioning for pseudospectral solutions of elliptic problems, *SIAM J. Sci. Stat. Comp.* **11**, 311 (1990).
9. M. Dubiner, Spectral methods on triangles and other domains, *J. Sci. Comp.* **6**, 345 (1991).
10. P. F. Fischer, An overlapping Schwarz method for spectral element solution of the incompressible Navier–Stokes equations, *J. Comput. Phys.* **133**, 84 (1997).
11. W. J. Gordon and C. A. Hall, Transfinite element methods: Blending function interpolation over arbitrary curved element domains, *Numer. Math.* **21**, (1973).
12. W. Heinrichs, Spectral collocation on triangular elements, *J. Comput. Phys.* **145**, 743 (1998).
13. R. D. Henderson, *Unstructured Spectral Element Methods: Parallel Algorithms and Simulations*, Ph.D. thesis (Princeton University, 1994).
14. R. D. Henderson, Details of the drag curve near the onset of vortex shedding, *Phys. Fluids* **7**(9), 1 (1995).
15. J. S. Hesthaven, From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, *SIAM J. Numer. Anal.* **35**, 655 (1998).
16. J. S. Hesthaven and C. H. Teng, Stable spectral methods on tetrahedral elements, *SIAM J. Sci. Comp.*, in press.

17. G. Karniadakis, M. Israeli, and S. A. Orszag, High-order splitting methods for the incompressible Navier–Stokes equations, *J. Comput. Phys.* **97**, 414 (1991).
18. G. Karniadakis and S. J. Sherwin, *Spectral hp Element Methods for CFD* (Oxford Univ. Press, London, 1999).
19. T. Koornwinder, Two-variable analogues of the classical orthogonal polynomials, in *Theory and Applications of Special Functions*, edited by A. R. A. (Academic Press, San Diego, 1975).
20. L. Kovasznay, Laminar flow behind a two-dimensional grid, *Proc. Cambridge Philos. Soc.*, 44 (1948).
21. J. A. Meijerink and H. A. Van der Vorst, Guidelines for the usage of incomplete decompositions in solving sets of linear equations as they appear in practical problems, *J. Comput. Phys.* **44**, 134 (1981).
22. S. A. Orszag, Spectral methods for problems in complex geometries, *J. Comput. Phys.* 70 (1980).
23. R. Owens, Spectral approximations on the triangle, *Proc. R. Soc. London A* **454**, 857 (1998).
24. S. V. Parter and E. E. Rothman, Preconditioning Legendre spectral collocation approximations to elliptic problems, *SIAM J. Numer. Anal.* 333 (1995).
25. L. F. Pavarino and T. Warburton, *Overlapping Schwarz Methods for Unstructured Spectral Elements*, Tech. Report 15/99 (University of Milan, 1999).
26. J. Proriol, Sur une famille de polynômes à deux variables orthogonaux dans un triangle, *C.R. Acad. Sci. Paris* **257**, 2459 (1967).
27. S. Sherwin and G. Karniadakis, Tetrahedral hp finite elements: Algorithms and flow simulations, *J. Comput. Phys.* **124**, 14 (1996).
28. S. Sherwin, T. Warburton, and G. Karniadakis, Spectral/hp methods for elliptic problems on hybrid grids, in *Domain Decomposition Methods 10*, edited by J. Mandel, C. Farhat, and X.-C. Cai, Vol. 218, AMS, Contemporary Mathematics (1998), pp. 191–216.
29. S. J. Sherwin, *Triangular and Tetrahedral Spectral/hp Finite Element Methods for Fluid Dynamics*, Ph.D. thesis (Princeton University, 1995).
30. J. R. Shewchuk, Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator, in *Applied Computational Geometry: Towards Geometric Engineering*, edited by M. C. Lin and D. Manocha, Vol. 1148, Lecture Notes in Computer Science (Springer-Verlag, New York, 1996), pp. 203–222. From the First ACM Workshop on Applied Computational Geometry.
31. K. Stuben and U. Trottenberg, Multigrid methods: Fundamental algorithms, model problem analysis and applications, in *Multigrid Methods*, edited by W. Hackbusch and U. Trottenberg (Springer Berlin, New York, 1982), pp. 1–176.
32. G. Szegő, Orthogonal polynomials, in *Colloquium Publications 23* (American Mathematical Society, Providence, R.I., 1939).
33. M. Taylor and B. Wingate, Fekete collocation points for triangular spectral elements, submitted for publication.
34. T. C. E. Warburton, *Spectral/hp Methods on Polymorphic Multi-Domains: Algorithms and Applications*, Ph.D. thesis (Brown University, Division of Applied Mathematics, 1999).