Proceedings of FEDSM2000
ASME 2000 Fluids Engineering Division Summer Meeting
June 11-15, 2000, Boston, Massachusetts, USA

FEDSM2000-11338

ELEMENT TOPOLOGY CONSIDERATIONS WHEN EMPLOYING A HIERARCHICAL BASIS FOR STABILIZED FEM IN FLUID DYNAMICS

Kenneth E. Jansen *
Anil Kumar Karanam
Christian H. Whiting †

Scientific Computation Research Center
Mechanical and Aeronautical Engineering & Mechanics
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: jansen@rpi.edu

ABSTRACT

A stabilized finite element formulation for fluid dynamics using mesh-entity based hierarchical basis functions is presented. The effect of element topology on solution quality and algorithm efficiency are carefully studied. The numerical formulation is presented in the context of the *p*-version finite element method for the 3 dimensional, incompressible Navier Stokes equations. A simple flow where analytical solution exists is studied and results are presented that verify the theoretical convergence results. The higher order simulations are compared to traditional linear basis finite element methods. Each methods cost in attaining a fixed level of accuracy is assessed and cubic hexahedral elements are seen to be the most efficient.

INTRODUCTION

Over the last two decades, stabilized finite element methods have grown in popularity, especially in application to fluid dynamics. Starting with the SUPG method (Brooks & Hughes, 1982), through the Galerkin/least squares (GLS) method (Hughes *et al*, 1989) and up to recent work on multiscale methods of (Hughes *et al*, 1998) and related work on residual free bubbles by (Rúsco, 1996) and (Brezzi *et al*, 1997), a number of stabilized formulations have been proposed. A key feature of stabilized methods is that they have been proven (for relevant model problems) stable and to attain optimal convergence rates with respect to the interpolation error (Franca & Frey, 1992; Hughes *et al*, 1989). This implies that as the polynomial order of the underlying finite element basis is increased, the error in the numerical solution will decrease at the same rate as the interpolation error.

To achieve cost effective simulations, however, some combination of *h*- and *p*-refinement is necessary for problems with discontinuities (Oden, 1994). Smooth regions of the flow benefit from *p*-refinement, while regions with discontinuities or sharp layers are better resolved using *h*-refinement. The present work is concerned with the development of an efficient uniform *p*-version stabilized finite element code for the compressible and incompressible Navier-Stokes equations, with the later goal of non-uniform $p-$ as well as $hp-$adaptive simulations. To complete simulations in a reasonable amount of time, the implementation relies heavily on the use of parallel computers using the MPI library for interprocessor communication.

The hierarchical basis that we have chosen is based on an abstract mesh data structure (Beall & Shephard, 1997), where basis functions are considered to be attached to the individual topological entities of the mesh. Mesh entity based hierarchical basis functions support non-uniform *p* refinement of meshes of

*Work supported by National Science Foundation, Address all correspondence to this author.
†Work supported by NASA Langley.

arbitrary element type, e.g. tetrahedral, hexahedral, and pyramid. To gain this generality, we have dispensed with the traditional finite element mesh data structures consisting of only element nodal connectivity (Hughes, 1987) in favor of a more general and complete topological adjacency mesh representation. More detail concerning the hierarchical basis used herein can be found in (Whiting & Jansen, 1999).

There has for many years been an ongoing debate related to the accuracy of hexahedra versus tetrahedra (similar debates continue about wedges as well) when using traditional Lagrange basis (usually with $p \leq 2$). Herein we re-examine this debate with regard to the hierarchic basis where much higher polynomial order can be easily considered.

The incompressible Navier-Stokes equations

Stabilized finite element methods have been proven to be stable and higher-order accurate for a linear advective-diffusive system (the closest model problem to the Navier-Stokes equations) in (Hughes *et al*, 1989) and for the linearized incompressible Navier-Stokes equations in (Franca & Frey, 1992). These types of formulations have also proven effective for computing complex compressible and incompressible turbulent flows (Jansen, 1999), wherein lies the target application of the present work. The higher-order accuracy properties as well as the robustness on complex flows have motivated our choice of finite element formulation. We first provide the strong form of the incompressible Navier-Stokes equations, followed by a description of the finite element method used to discretize the associated weak form.

Strong form

Consider the application of the mesh entity based hierarchical basis functions (described above) to the time-dependent, incompressible Navier-Stokes equations. First, consider the strong form of the continuity and momentum equations written in the so-called advective form (Gresho, 1991)

$$\begin{aligned} u_{i,i} &= 0 \\ \dot{u}_i + u_j u_{i,j} &= -p_{,i} + \tau_{ij,j} + f_i \end{aligned} \tag{1}$$

where u_i is the i^{th} component of velocity, p the pressure divided by the density ρ (assumed constant), f_i the prescribed body force, and τ_{ij} the viscous stress tensor given by:

$$\tau_{ij} = \nu(u_{i,j} + u_{j,i}) \tag{2}$$

where $\nu = \frac{\mu}{\rho}$ is the kinematic viscosity, and the summation convention is used throughout (sum on repeated indices).

Weak form – Finite element discretization

To proceed with the finite element discretization of the weak form of the Navier-Stokes equations (1), we first introduce the discrete weight and solution function spaces that are used. Recall that $\bar{\Omega} \subset R^N$ represents the closure of the physical spatial domain, $\Omega \cup \Gamma$, in N dimensions; only $N = 3$ is considered. The boundary is decomposed into portions with natural boundary conditions, Γ_h, and essential boundary conditions, Γ_g, i.e., $\Gamma = \Gamma_g \cup \Gamma_h$. In addition, $H^1(\Omega)$ represents the usual Sobolev space of functions with square-integrable values and derivatives on Ω.

Subsequently Ω is discretized into n_{el} finite elements, $\bar{\Omega}_e$. With this, we can define the discrete trial solution and weight spaces for the semi-discrete formulation as

$$\begin{aligned} S_h^p = \{v | v(\cdot,t) \in H^1(\Omega)^N, t \in [0,T], v|_{x \in \bar{\Omega}_e} \in P_p(\bar{\Omega}_e)^N, \\ v(\cdot,t) = g \text{ on } \Gamma_g\}, \end{aligned} \tag{3}$$

$$\begin{aligned} W_h^p = \{w | w(\cdot,t) \in H^1(\Omega)^N, t \in [0,T], w|_{x \in \bar{\Omega}_e} \in P_p(\bar{\Omega}_e)^N, \\ w(\cdot,t) = 0 \text{ on } \Gamma_g\}, \end{aligned} \tag{4}$$

$$P_h^p = \{p | p(\cdot,t) \in H^1(\Omega), t \in [0,T], p|_{x \in \bar{\Omega}_e} \in P_p(\bar{\Omega}_e)\} \tag{5}$$

where $P_p(\bar{\Omega}_e)$ denotes a polynomial basis complete to order p. Let us emphasize that the local approximation space, $P_p(\bar{\Omega}_e)$, is the same for both the velocity and pressure variables. This is possible due to the stabilized nature of the formulation to be introduced below. These spaces represent discrete subspaces of the spaces in which the weak form is defined.

The stabilized formulation used in the present work is based on that described by (Taylor *et al*, 1998), but is modified to include the higher-order basis functions. Given the spaces defined above, we present the semi-discrete, stabilized, Galerkin finite element formulation applied to the weak form of (1) as:

Find $u \in S_h^p$ and $p \in P_h^p$ such that

$$B(w_i, q; u_i, p) = 0$$

$$\begin{aligned} B(w_i, q; u_i, p) = & \int_\Omega \{ w_i \left(\dot{u}_i + u_j u_{i,j} - f_i \right) \\ & + w_{i,j} \left(-p\delta_{ij} + \tau_{ij} \right) - q_{,i} u_i \} dx \\ & + \sum_{e=1}^{n_{el}} \int_{\bar{\Omega}_e} \{ \tau_M (u_j w_{i,j} + q_{,i}) L_i + \tau_C w_{i,i} u_{j,j} \} \, dx \\ & + \sum_{e=1}^{n_{el}} \int_{\bar{\Omega}_e} \{ w_i \overset{\Delta}{u}_j u_{i,j} + \overset{-\Delta}{\tau} u_j w_{i,j} \overset{\Delta}{u}_k u_{i,k} \} \, dx \\ & + \int_{\Gamma_h} \{ w_i \left(p\delta_{in} - \tau_{in} \right) + q u_n \} \, ds \end{aligned} \tag{6}$$

for all $w \in W_h^p$ and $q \in P_h^p$. The boundary integral term arises from the integration by parts and is only carried out over the portion of the domain without essential boundary conditions. We have used L_i to represent the residual of the i^{th} momentum equation,

$$L_i + p_{,i} - \tau_{ij,j} - f_i \tag{7}$$

The fourth line in the stabilized formulation, (6), represents the typical stabilization added to the Galerkin formulation for the incompressible set of equations (Franca & Frey, 1992). The first term in the third line of (6) was introduced by (Taylor *et al*, 1998) to overcome the lack of momentum conservation introduced as a consequence of the momentum stabilization in the continuity equation. The second term on this line was introduced to stabilize this new advective term. The stabilization parameters for continuity and momentum are defined in (Whiting & Jansen, 1999).

Numerical examples

This section presents numerical simulations using the methods described above. The accuracy of the method is first verified on a problem with a known analytical solution. A second example is then provided to demonstrate the ability of higher-order basis methods to attain more accurate simulations for substantially less cost.

The simulations described below were all performed with the full three-dimensional code. To simulate the two-dimensional flows described below, we have used two vertices in the x_3 direction and imposed no x_3 velocity and zero viscous flux through the x_3 planes; i.e., $u_3 = 0, \tau_{i3} = 0$. The depth in the x_3 direction was set equal to a characteristic length in the problem.

Kovasznay flow

The first example we wish to consider is an analytic solution known as Kovasznay flow (Kovasznay, 1948) which was devised to resemble the flow behind a grid. We will use this flow to demonstrate the convergence of the method since we have a closed form analytical expression for the exact solution, given by:

$$u_1 = 1 - e^{\lambda x_1}\cos(2\pi x_2) \tag{8}$$

$$u_2 = \frac{\lambda}{2\pi} e^{\lambda x_1}\sin(2\pi x_2) \tag{9}$$

with

$$\lambda = \frac{\mathrm{Re}}{2} - \sqrt{\frac{\mathrm{Re}^2}{4} + 4\pi^2} \tag{10}$$

and we have taken $\mathrm{Re} = 40$ for the present study. The flow is considered on a rectangular domain of $0 \le x_1 \le 1$ and $-\frac{1}{2} \le x_2 \le \frac{1}{2}$ with the exact solution imposed as an essential boundary condition in the inflow and upper and lower walls, while the pressure was set at the outflow. The qualitative behavior of the solution is depicted in Figure 1 which shows contours of fluid speed for the cubic tetrahedral simulation on the 21×21 mesh.

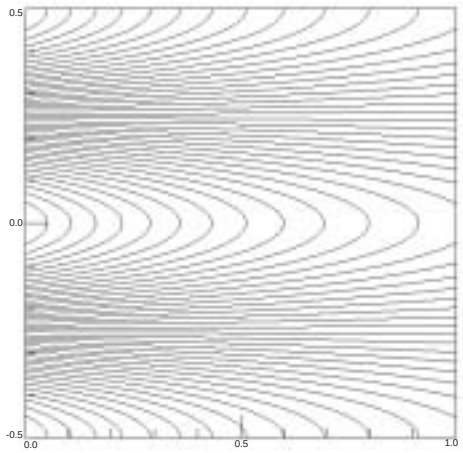
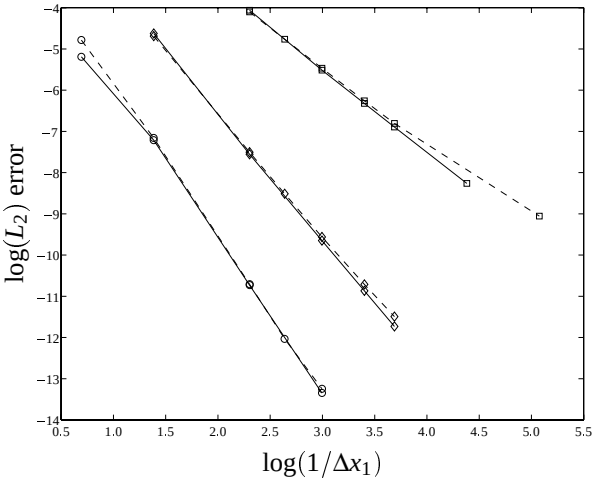


Figure 1. KOVASZNAY FLOW. CONTOURS OF FLUID SPEED FOR CUBIC SIMULATION ON 21×21 MESH

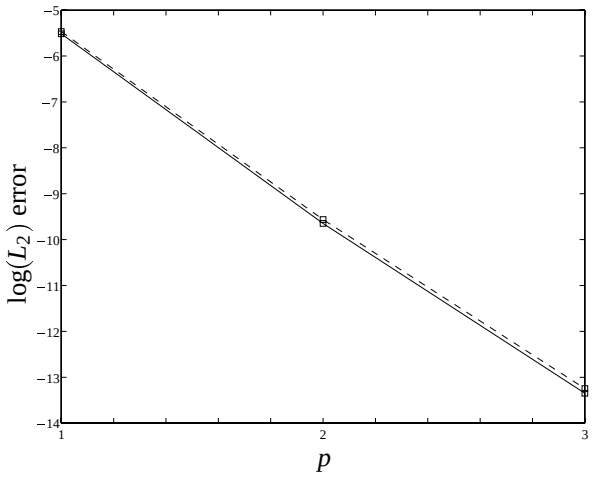
A convergence study was performed for this flow to determine the accuracy of the method while varying the element topology, the mesh size, Δx_1, and the polynomial order of the basis. Figures 2(a) and 2(b) show the log of the normalized L_2 error in the velocity field versus $\log(1/\Delta x_1)$ and polynomial order, respectively. From Figure 2(a) we have determined the rates of convergence of the tetrahedral elements to be 1.97, 3.00, and 3.93 for the linear, quadratic, and cubic simulations, respectively. In (Whiting & Jansen, 1999) results for the error in H_1 also showed optimal convergence rates (i.e. $O(h^p)$). The hexahedral elements achieve the theoretical predictions for the interpolation error $O(\Delta x_1^{p+1})$ (which yields 1.99, 3.00, and 3.81 for linear, quadratic and cubic respectively). H_1 norms for hexahedra also show optimal convergence (not shown here). Furthermore, it is clear from Figure 2(a) that the constant in the error estimate also greatly improves for the higher-order simulations, making the higher polynomial order basis most attractive even on the coarsest meshes. Finally, we point out that hexahedral elements have a better constant than the tetrahedra (there error is lower than the tetrahedra for any given mesh), though the difference is not easily seen in this figure. Figure 2(b) demonstrates the exponential convergence of the method when Δx_1 is fixed and the polynomial order is increased.

The under-performance of the linear tetrahedra, relative to

tri-linear hexahedra has been observed in the past. This under performance is offset to some degree by issues such as geometric flexibility and reduced cost to integrate and solve. We now turn our attention to this issue.



(a) log of L_2 error vs. $\log(1/\Delta x_1)$. $\circ$; $p = 3$, $\diamond$; $p = 2$, $\square$; $p = 1$, solid line; hex, dashed line; tet.



(b) log of L_2 error vs. p. Solid line; hex, dashed line; tet.

Figure 2. KOVASZNAY FLOW CONVERGENCE STUDY.

It is somewhat difficult to get a fair cost comparison on 2D problems when using a 3D code. This is due to the fact that the cost of element formation and solving the resulting system does not scale in the same way for tetrahedra and hexahedra. Furthermore, the cost of the higher polynomial order methods will be adversely affected by adding modes to the third dimension (where no resolution is needed). Still, we would like to make some estimates of the relative cost and efficiencies of these methods and so we consider three cost indices. C_1 will be associated with the cost of forming a left hand side (LHS) (residual tangent matrix) ($C_1 = n_f * n_{shp}^2 * n_{int}$) that is, the number of face elements (n_f) that a 2D solver would use times the number of shape functions (n_{shp}) on one of the 2D elements times the number of integration points (n_{int}) required to accurately integrate each 2-D element. C_2 will be associated with the cost of forming a right hand side (RHS) residual vector ($C_2 = n_f * n_{shp} * n_{int}$). C_3 will be associated with the cost of performing the linear solve with an iterative method ($C_3 = n_k * (n_v * nnz_v + n_e * nnz_e + n_f * nnz_f)$ where the subscripts n_v, n_e, and n_f correspond to the total number of vertices, edges and faces respectively (in an equivalent 2D domain) and nnz_v, nnz_e, and nnz_f correspond to the predictable (only on this structured mesh) non-zero fill pattern associated with each entity and n_k is the total number of Krylov vectors used by the iterative solver. This last cost is appropriate because we have employed a sparse-storage solver which, for each Krylov vector, performs an Ap product by only multiplying the non-zero entries of A times p.

Of particular relevance is the value of each of these cost measures when trying to obtain a solution of a specified accuracy. Consider the case where the quadratic and the linear bases attempt to achieve the same accuracy as the cubic basis on the coarsest hexahedral mesh (5×5 points). The results of these cost studies are illustrated in Table 1. From this Table it is clear that, for this level of accuracy, the cubic simulation is the most cost effective for almost all cost measures considered. For matrix formation costs, C_1, the higher order approaches enjoy only a slight advantage. This is to be expected since element matrices grow nonlinearly (i.e. n_{shp}^2) so that despite the reduction in the number of elements required to attain a given accuracy, this cost remains relatively constant. This issue can be mitigated by noting that it is not necessary to form the element stiffness matrix at every time step. We have observed that, once past the first few steps, where the tangent to the residual changes rapidly, the LHS can be stored and reused for 10 or more steps/iterations. This effectively takes this cost out of the problem. This is important since otherwise, as p rises, LHS matrix formation can become the dominant cost. Even unsteady problems can typically follow this strategy though greater care must be taken to ensure that convergence within a given step is not compromised. The second cost index, C_2, shows a significant cost reduction for increasing p (factor of 12-15 more for linear (hexes or tets) than cubic hexes).

Though we have a rather complicated RHS to form (viz. (6)) the total cost of our calculations are often dominated by the iterative solver. Therefore, C_3 is often an important cost to consider. Here, we see an even more dramatic increase in computational effort as p decreases. While one expects the equations to get more stiff as the polynomial order increases we observe that the accuracy attained more than offsets the cost. It is important to note that decreasing element size also results in element

stiffness and lower order methods must undergo many more subdivisions to obtain the same accuracy of the higher polynomial orders, increasing the number of iterations required and increasing the number of degrees of freedom in the system, both direct contributors to the cost of solution.

Also note from this table that tetrahedra are also less cost efficient than hexahedra according to all three cost indices (except for the quadratic tets for C_2). Again we have interpolated the real tetrahedral data to fictional meshes that would achieve the same level of accuracy as the coarsest, cubic hexahedral mesh according to Figure 2(a).

The same calculations were repeated to determine the cost of obtaining solutions with the same total error as the next finer (11×11 point) cubic hexahedral mesh. The results (not shown here) showed an even more dramatic advantage for the cubic elements relative to quadratic and linear elements. This is not surprising since cubics not only enjoy the fastest rate of convergence but they also have the best constant. These attributes allow them to consistently use enough fewer elements to offset the increase in stiffness associated with increasing polynomial order. Hexahedra continued to enjoy some computational advantage over tetrahedra due to the fact that the a hex's element degree of freedom count rises slower than that of a tet.

Table 1. KOVASZNAY RELATIVE COST COMPARISON, EXTRAPOLATING DATA TO FICTIONAL MESHES WHICH WOULD ACHIEVE THE SAME ACCURACY AS THE CUBIC HEXAHEDRA ($5 \times 5 POINTS$).

Topology	p	C_1	C_2	C_3
Hex(5x5)	3	1.00	1.00	1.00
Hex(10x10)	2	1.27	1.90	1.84
Hex(49x49)	1	4.00	12.0	30.1
Tet(5x5)	3	1.04	1.25	2.00
Tet(10x10)	2	0.95	1.90	2.08
Tet(51x51)	1	3.66	14.6	83.4

Lid-driven cavity flow

The final problem considered is the steady, two-dimensional flow inside a closed container driven by it's lid. The lid slides to the right across the top of the cavity, shearing the fluid and setting up a recirculation region. There is a primary vortex in the center of the cavity and secondary eddies in the lower corner (the number of these secondary eddies depends on the Reynolds number). While there is no exact solution available for this flow, the singularities in the corners provide a very different challenge to the method than the Kovasznay flow. For the present study, we have chosen to consider $Re = 400$ (based on the lid velocity), for which there exist well-established benchmark results with which to compare (Ghia *et al*, 1982).

In addition to the velocity constraints (no slip on walls and unit velocity on the top), the pressure field is constrained by setting its value at the single vertex in the lower left corner of the cavity. Uniform meshes were used with equal spacing in the x_1- and x_2- directions. To isolate the singularities in the upper corners, nested local mesh refinement was used by subdividing the original corner elements. The number of new corner elements was chosen such that the first point is 3.90625×10^{-4} units from the corner for each mesh. This distance dictates the extent to which the discontinuity in the velocity field is resolved (i.e. how much fluid is "leaked" from the cavity).

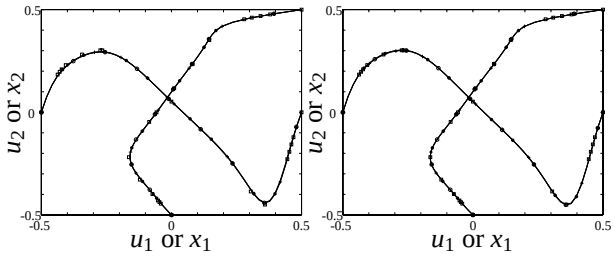
A cost comparison study similar to the Kovasznay flow study was carried out for the lid-driven cavity flow (see the velocity profiles that are of the same quality in Figure 3(a) and Figure 3(b)). The simulations, along with the second and third cost indices are shown in Table 2. Again, the cubic simulations show a dramatic advantage relative to all other choices on all cost indices. Hexahedral elements enjoy an efficiency advantage even though we are using three meshes with identical point placement. This is due to the fact that they have a slower rate of growth in terms of degrees of freedom per element as polynomial order increases. Here we have also provided information comparing the memory requirements and disk storage required for the simulations. The "Matrix" column of Table 2 indicates the number of nonzero blocks for the sparse storage of the tangent matrix (the dominant memory requirement), indicating that the memory requirements for the cubic hexahedral simulation are about 28 times less than that of the linear hexahedral simulation (tetrahedra are slightly better at only 22 times the memory), while the quadratic requires about a factor of 9 times the memory for both topologies. The "Mesh" column compares the size of the file that stores the coordinates and connectivity of the mesh. Here again, we see a dramatic advantage for the coarse, very accurate cubic hexahedral element over all other topologies. We also see that tetrahedral elements require a larger connectivity than their hexahedral counterparts for all polynomial orders. This well known result for linears extends to the hierarchical basis.

Conclusions

A stabilized finite element method using hierarchical basis functions applied to the incompressible Navier-Stokes equations has been presented. Though the implementation is for unstructured three dimensional meshes (hexahedral or tetrahedral), two dimensional problems were studied with the goal of assessing the cost versus performance of various combinations of element topology and polynomial order ($p < 4$). From the problems studied in this paper it was clear that the cubic hexahedral elements were the most cost effective regardless of the cost measure (LHS formation, RHS formation, iterative solve, runtime memory, or disk storage) and regardless of the level of accuracy desired (e.g.,

Table 2. LID-DRIVEN CAVITY NORMALIZED COST COMPARISON

Mesh	p	C_2	C_3	Mesh	Matrix
11×11 hex	3	1.00	1.00	1.00	1.00
41×41 hex	2	6.00	29.7	6.96	9.29
161×161 hex	1	21.3	480	52.3	27.8
11×11 tet	3	1.25	1.47	1.81	1.38
41×41 tet	2	6.00	31.9	10.6	9.04
161×161 tet	1	24.0	457	75.1	21.6



(a) Qualitatively similar solutions for hexes.

(b) Qualitatively similar solutions for tets.

Figure 3. LID-DRIVEN CAVITY FLOW. PLOTS OF $u_1(x_1 = 0, x_2)$ AND $u_2(x_1, x_2 = 0)$, (∗) LINEAR ON 161×161 MESH, (∘) QUADRATIC ON 41×41 MESH AND (+) CUBIC ON 11×11 MESH, (□) GHIA *et al*.

even on very coarse meshes). It should be pointed out, however, that as p increased, tetrahedral elements closed the gap and are capable of attaining exponential convergence. This is important since the most robust automatic mesh generators and the most straightforward adaptivity methods still rely on tetrahedral elements when dealing with complex geometry. However, it was also observed that tetrahedral elements add degrees of freedom more rapidly than hexahedral elements as p increases.

Acknowledgments

The authors would like to thank Farzin Shakib of Acusim software for providing access to his linear algebra software.

REFERENCES

M. W. Beall and M. S. Shephard, "A general topology-based mesh data structure", *Int. J. Numer. Meth. Engng.*, **40** (1997) 1573–1596.

F. Brezzi, L. P. Franca, T. J. R. Hughes, and A. Russo, "$b = \int g$", *Comp. Meth. Appl. Mech. Engng.*, **145** (1997) 329–339.

A. N. Brooks and T. J. R. Hughes, "Streamline upwind / Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations", *Comp. Meth. Appl. Mech. Engng.*, **32** (1982) 199–259.

L. P. Franca and S. Frey, "Stabilized finite element methods: II. The incompressible Navier-Stokes equations", *Comp. Meth. Appl. Mech. Engng.*, **99** (1992) 209–233.

U. Ghia, K. N. Ghia, and C. T. Shin, "High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method", *Journal of Computational Physics*, **48** (1982) 387–441.

P. M. Gresho, "Some current CFD issues relevant to the incompressible Navier-Stokes equations", *Comp. Meth. Appl. Mech. Engng.*, **87** (1991) 201–252.

T. J. R. Hughes, L. P. Franca, and G. M. Hulbert, "A new finite element formulation for fluid dynamics: VIII. The Galerkin / least–squares method for advective–diffusive equations", *Comp. Meth. Appl. Mech. Engng.*, **73** (1989) 173–189.

T. J. R. Hughes, G. Feijóo., L. Mazzei, and J. B. Quincy, "The variational multiscale method – A paradigm for computational mechanics", *Comp. Meth. Appl. Mech. Engng.*, **166** (1998) 3–24.

Thomas J. R. Hughes, *The finite element method: Linear static and dynamic finite element analysis*. Prentice Hall, Englewood Cliffs, NJ, 1987.

K. E. Jansen, "A stabilized finite element method for computing turbulence", *Comp. Meth. Appl. Mech. Engng.*, **174** (1999) 299–317.

L. I. G. Kovasznay, "Laminar flow behind a two-dimensional grid", *Proc. Cambridge Philos. Soc.*, **44** (1948).

J. T. Oden, "Parallel adaptive hp-finite element methods for problems in fluid and solid mechanics", in T. J. R. Hughes, E. Onate, and O. C. Zienkiewicz, editors, *Recent developments in finite element analysis*, 29–35, International Center for Numerical Methods in Engineering, Barcelona, Spain, 1994.

A. Rússo, "Bubble stabilization of the finite element methods for the linearized incompressible Navier-Stokes equations", *Comp. Meth. Appl. Mech. Engng.*, **132** (1996) 335–343.

C. A. Taylor, T. J. R. Hughes, and C. K. Zarins, "Finite element modeling of blood flow in arteries", *Comp. Meth. Appl. Mech. Engng.*, **158** (1998) 155–196.

C. H. Whiting and K. E. Jansen, "A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis", *International Journal of Numerical Methods in Fluids*, **accepted** (1999), SCOREC Report 9-1999, Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY.