J Eng Math (2006) 56:337–350
DOI 10.1007/s10665-006-9081-y

ORIGINAL PAPER

An adaptive least-squares spectral collocation method with triangular elements for the incompressible Navier–Stokes equations

Wilhelm Heinrichs

Received: 15 September 2005 / Accepted: 20 July 2006 /
Published online: 27 September 2006
© Springer Science+Business Media B.V. 2006

Abstract A least-squares spectral collocation scheme for the incompressible Navier–Stokes equations is proposed. Grid refinement is performed by means of adaptive triangular elements. On each triangle the Fekete nodes are employed for the collocation of the differential equation. On the element interfaces continuity of the functions is enforced in the least-squares sense. Equal-order Dubiner polynomials are used to obtain a stable spectral scheme. The collocation conditions and the interface conditions lead to an overdetermined system that can be solved efficiently by least-squares. The solution technique only involves symmetric positive-definite linear systems. The approach is first applied to the Poisson equation and then extended to singular perturbation problems where least-squares have a stabilizing effect. By adaptivity, a suitable decomposition of the domain is found where the boundary layer is well resolved. Finally, the method is successfully applied to the regularized driven-cavity flow problem. Numerical simulations confirm the high accuracy of the proposed spectral least-squares scheme.

Keywords Least-squares · Spectral collocation · Adaptivity · Triangular elements · Navier–Stokes

1 Introduction

Spectral methods [1–3] employ global polynomials for the numerical solution of differential equations. Hence they give very accurate approximations for smooth solutions with relatively few degrees of freedom (d.o.f). For analytical data exponential convergence can be achieved. A good review of these methods can be found in the book [4]. Reference [5] shows the good performance of least-squares spectral-element methods for incompressible flows with continuous analytical solutions. The spectral solution can exhibit (small) jumps over the boundary interfaces. In [6] the above approach was extended to one-dimensional singular perturbation problems where the least-squares collocation schemes promote stabilization. This was already observed by Eisen and Heinrichs in earlier work [7]. After a suitable decomposition of the domain, we were able to resolve the boundary layer (see [8, 9] for similar approaches).

W. Heinrichs (✉)
Universität Duisburg–Essen, Ingenieurmathematik, Universitätsstr. 3, D-45117 Essen, Germany
e-mail: Wilhelm.Heinrichs@uni-due.de

Here we extend this method to the two-dimensional case where a triangular decomposition of the domain is employed. The main advantage of triangles is that, after an adaptive refinement, hanging nodes and nonconforming elements (Mortar elements) are avoided (see [10] for a comprehensive review). Our technique of adaptive mesh refinement is adopted from the finite-element implementation (see, e.g., [11, 12]). The next question is the spectral discretization on the triangles. In earlier approaches (see [13–16]) mapping techniques from the square to the triangle were used. Disadvantages are a large condition number and irregular distributions of collocation nodes. Other approaches [17–20] directly deal with finding optimal interpolation nodes on the triangle. Reference [19] uses a second-order time-evolution method to minimize an electrostatic energy function. References [17, 18, 20] investigate Fekete points that maximize the determinant of the Vandermonde matrix. Here we also make use of the Fekete points. They were explicitly calculated in [20] and applied in [21] to spectral-element methods. For stability reasons, equal-order Dubiner polynomials [22] as basis functions are employed (see [15] for a comprehensive review). On each triangular element, we collocate the first-order system of differential equations. Continuity of the variables at interfaces is enforced in the least-squares sense and not by virtue of the element assembly process. The collocation conditions together with the interface conditions lead to an overdetermined system that can be approximately solved by the least-squares method. We obtain symmetric and positive-definite systems that can be efficiently solved by direct solvers with LU-factorization for sparse matrices. We recommend the UMFPACK Routine from the MATLAB library [23, 24].

Our spectral least-squares method is first applied to elliptic problems. We consider the Poisson equation and singular perturbation problems and observe a good performance of the adaptive grid refinement. For stiff problems, we obtain a stable scheme that is able to resolve boundary layers with spectral accuracy. These techniques are then extended to the Stokes and Navier–Stokes equations. It is already known from the finite-element case that the least-squares formulation of the Stokes [25–27] and Navier–Stokes equations [28, 29] leads to symmetric and positive-definite algebraic systems. For spectral methods, it is known that, if the velocity and the pressure are approximated by polynomials of the same degree, eight spurious modes are introduced that lead to an unstable system (see [30]). A well-known compatible approximating velocity–pressure pair is the so-called $\mathbb{P}_N \times \mathbb{P}_{N-2}$ formulation of [31] and [32]. Reference [33] employs this technique for the splitting of the Stokes equations. Here, the velocity components are approximated by polynomials in $\mathbb{P}_N$, and the pressure is approximated by two degrees lower-order polynomials in $\mathbb{P}_{N-2}$. The resulting discrete system constitutes a saddle-point problem that is difficult to solve numerically. Least-squares techniques offer theoretical and numerical advantages over the classical methods.

Spectral least-squares methods for the Stokes equations were first introduced in [34, 35]. Spectral least-squares for the Navier–Stokes equations were first presented in [36, 37], followed by [38]. In addition, there is recent work [39] on discontinuous least-squares spectral-element methods in two dimensions for incompressible flows with weak C^1-regularity [39]. Reference [40] introduced least-squares collocation schemes for the Navier–Stokes equations. Here, we extend these methods to a triangular decomposition of the domain combined with adaptive mesh refinement. On each subdomain, a spectral scheme with Fekete nodes is employed. In summary, our approach has the following advantages:

- Equal-order interpolation polynomials can be employed. It is still possible to vary the polynomial order from element to element.
- Stability for singular perturbation problems [6, 7] and the Stokes or Navier–Stokes equations [34, 35, 38, 40, 41].
- Adaptivity and a posteriori error estimators.
- Direct or iterative solvers for positive-definite systems (e.g., Cholesky or conjugate gradient methods).

In Sect. 2, the spectral least-squares scheme for elliptic problems is introduced. In Sect. 3, the adaptive mesh refinement is described and the implementation is discussed. Finally, a conclusion is presented.

2 The spectral least-squares scheme

We first introduce the spectral (partial) derivatives on T defined by

$T = \{(x, y):\ x \geq -1,\ y \geq -1,\ x + y \leq 0\}.$

For triangles, the usual choice for interpolation base is a triangular truncation of polynomials, spanned by the monomials

$\{x^m y^n,\ m + n \leq N\}.$

The polynomial subspace spanned by these monomials is denoted by $\mathbb{P}_N$ with $\dim \mathbb{P}_N = (N+1)(N+2)/2$. Since monomials yield very bad condition numbers for the Vandermonde matrix, it is practically useful to choose the Dubiner basis functions [22] given by

$$\phi_{mn}(x, y) = L_m\left(2\frac{1+x}{1-y} - 1\right)(1-y)^m P_n^{2m+1,0}(y), \quad m + n \leq N,$$

defined with respect to the Legendre polynomials $L_m = P_m^{0,0}$ and the Jacobi polynomials $P_n^{2m+1,0}$. The Dubiner basis is orthogonal in the triangle and hence leads to a well-conditioned Vandermonde matrix. [20] obtained for the Fekete points (see Fig. 1 for the Fekete points on a equilateral triangle) a MATLAB reported condition number of less than 50 (for $N \leq 19$). For given nodes,

$(x_{ij}, y_{ij}), \quad 0 \leq i + j \leq N,$

and basis functions ϕ_{mn}, the Vandermonde matrix $V = (v_{ij,mn})$ is given by

$v_{ij,mn} = \phi_{mn}(x_{ij}, y_{ij}), \quad i + j \leq N,\ m + n \leq N.$

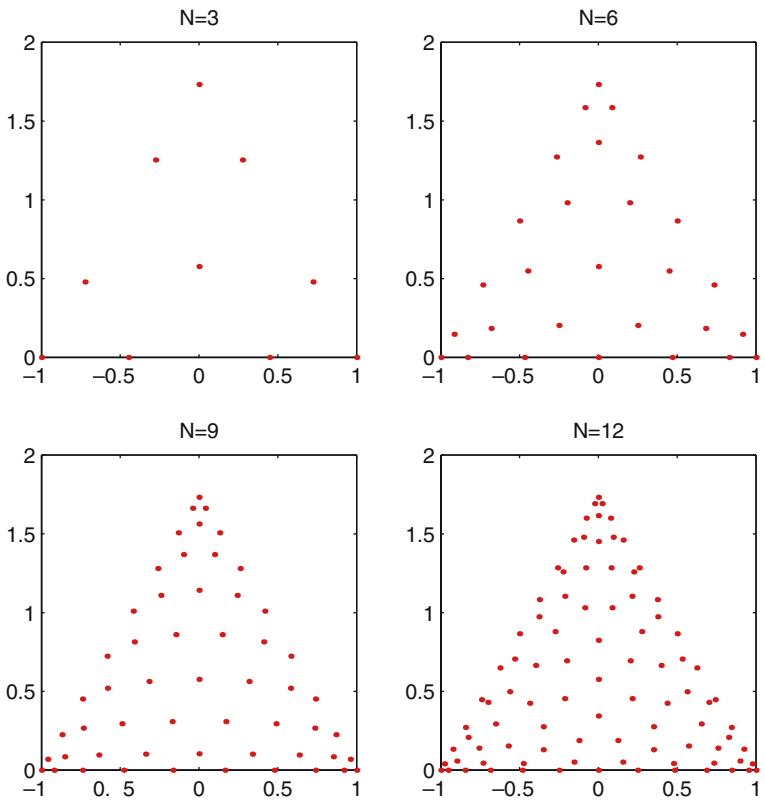

Fig. 1 Fekete points on the equilateral triangle for $N = 3, 6, 9, 12$

The Vandermonde matrix encapsulates the interpolation data at the above nodes. Now the Fekete points are determined by maximizing the determinant of the Vandermonde matrix. These points were approximately calculated by [20] using a modified steepest-ascent method. The spectral (partial) derivatives can be derived by first calculating the Dubiner expansion, and then taking partial derivatives, i.e.,

$$D_x = U_x V^{-1}, \quad D_y = U_y V^{-1},$$

where

$$U_x = \left(\frac{\partial}{\partial x}\phi_{mn}(x_{ij}, y_{ij})\right), \quad U_y = \left(\frac{\partial}{\partial y}\phi_{mn}(x_{ij}, y_{ij})\right).$$

For given N, these derivatives on T are calculated once at the beginning of the program. For spectral derivatives on general triangles $\hat{T}$ we use an affine transformation, i.e.,

$$\hat{x} = r_1 + r_2 x + r_3 y, \quad \hat{y} = s_1 + s_2 x + s_3 y,$$

where the coefficients are determined by mapping the three corners of T onto the vertices of $\hat{T}$. Now the partial derivatives transform as follows:

$$D_{\hat{x}} = \frac{1}{r_2 s_3 - s_2 r_3}\left(s_3 D_x - s_2 D_y\right), \quad D_{\hat{y}} = \frac{1}{r_2 s_3 - s_2 r_3}\left(-r_3 D_x + r_2 D_y\right).$$

These partial derivatives are also used for calculating normal derivatives along the edges of the triangles. Now we are able to write the spectral least-squares scheme for elliptic problems.

We consider the Poisson equation with Dirichlet boundary conditions

$$-\Delta u = f \text{ in } \quad \Omega = (0,1)^2 \tag{1}$$

$$u = g \text{ on } \quad \partial\Omega. \tag{2}$$

Equation 1 is split into an equivalent first-order system given by

$$u_x - v_1 = 0, \tag{3}$$

$$u_y - v_2 = 0, \tag{4}$$

$$-v_{1,x} - v_{2,y} = f \tag{5}$$

with the above Dirichlet boundary conditions (2) for u. For the finite-element method, the optimal splitting (see, e.g., [42, p. 102]) is given by

$$\nabla \times v = 0, \quad -\nabla \bullet v = f.$$

In this formulation, $v \in H^1(\Omega)$ makes for a stronger convergence. For the collocation method, it is not clear that the latter formulation is optimal, and hence we return to the conventional formulation. For a given triangulation of the domain, we approximate the partial derivatives on each triangle by means of the above spectral operators, D_x, D_y. The collocation equations can be written as

$$(u_x - v_1)(\hat{x}_{ij}, \hat{y}_{ij}) = 0, \quad \left(u_y - v_2\right)(\hat{x}_{ij}, \hat{y}_{ij}) = 0, \quad -\left(v_{1,x} + v_{2,y}\right)(\hat{x}_{ij}, \hat{y}_{ij}) = f(\hat{x}_{ij}, \hat{y}_{ij}),$$

where $(\hat{x}_{ij}, \hat{y}_{ij})$ are the collocation nodes on the triangle $\hat{T}$ for $0 \le i + j \le N$. Thus, we obtain for the above system $3(N+1)(N+2)/2$ collocation conditions which is identical to the number of unknowns in our system. Additionally, Dirichlet boundary conditions and continuity along element interfaces are enforced pointwise. From [17] it is known that the points along the boundary of the triangles are the standard one-dimensional Gauss–Lobatto nodes mapped onto the edge, say $(\xi_i, \eta_i), i = 0, \ldots, N$. Hence, on an interface we require, for instance,

$$\tilde{u}(\xi_i, \eta_i) = \hat{u}(\xi_i, \eta_i), \quad i = 0, \ldots, N,$$

where $\tilde{u}$ and $\hat{u}$ are approximations from different triangular elements sharing the same edge. These interface conditions are also explicitly written for the functions v_1, v_2. Hence at the Gauss–Lobatto points (on

each edge of the triangle) the spectral approximations should attain the same values (pointwise continuity!). The collocation conditions and the boundary/interface conditions are written into a matrix A and compiled into an overdetermined system $Az = r$ which is efficiently solved by least-squares. By minimizing

$$||Az - r||_{2,w} = \min!$$

in the weighted discrete L^2-norm we obtain the normal equations

$$\mathbf{A^T}\Lambda\mathbf{A}\,\mathbf{z} = \mathbf{A^T}\Lambda\,\mathbf{r},$$

where Λ is a diagonal matrix containing the quadrature weights. Here we choose $w \equiv 1$ and hence Λ is the identity matrix. The solution technique only involves symmetric positive-definite algebraic systems which allow the use of equal-order interpolation polynomials on all subdomains. Direct solvers for sparse matrices or robust iterative solvers such as (preconditioned) conjugate-gradient methods can be employed.

In the finite-element least-squares approach (e.g., by Gerritsma), there is a minimization of the least-squares functional written in L^2_w-norms. It is rewritten in a variational setting and evaluated by integration quadrature rules. In the collocation approach, we first write the collocation (pointwise) equations for the first-order system. The equations are implemented by interface and boundary conditions. The resulting overdetermined algebraic system is solved by means of the normal equations. For the finite-element least-squares the interface conditions are enforced in a variational setting. Hence, with the LSFEM the Sobolev errors are directly controlled, whereas in the collocation framework the only apparent control is in L^∞.

3 Adaptive mesh refinement

On each triangular element we evaluate an a-posteriori error indicator based on the H^1-norm. In particular, we refine when solution gradients are large, i.e., we require

$$||\nabla u^{(k)}|| \le \epsilon ||u^h||_1 \tag{6}$$

everywhere on the mesh, where $u^{(k)}$ denotes the approximate solution on the kth element. Here $||\cdot||$ denotes the discrete L^2 norm, $||\cdot||_1$ the discrete H^1 norm, and ϵ is a given discretization tolerance. In the later applications, we choose $\epsilon = 0.03$. This is a common refinement criterion in cases where no alternative measure of solution errors is available. In the least-squares variational setting, an a-posteriori error estimator is available through the least-squares functional itself. This yields a sharp and cheap measure for the local error. In the collocation approach (which is a pointwise method), this measure is not directly available without any further computations. One additionally has to compute the local error on each element. The costs are comparable to our error indicator.

For spectral methods, there exist other refinement strategies which are based on the decay of the coefficients in the polynomial expansion. A good survey on these methods can be taken from the review article [10]. From the discussion in [10] it is not clear that the other error indicators yield better results. Hence, for simplicity, we refer to the refinement criteria (6). In case it is not fulfilled, the corresponding triangle is refined as in the finite-element method. This means that all three edges are halved and the element is subdivided into four new sub-triangles. A more detailed review on these techniques can be found in the finite-element literature (see, e.g., [11] or [12]).

4 Numerical results

We have calculated the maximal errors over all elements, i.e.,

$$||u - u^h||, \quad ||u - u^h||_1$$

in the discrete L^2 and H^1 norm. First, we consider a smooth solution given by

Table 1 Numerical results for example (7)

N	$\|\|u-u^h\|\|$	$\|\|u-u^h\|\|_1$	Conv. rates	#el.	d.o.f.
3	1.63×10^{-1}	5.93×10^{0}	—	32	3×320
6	3.51×10^{-3}	2.12×10^{-1}	4.81	32	3×896
9	3.79×10^{-4}	3.85×10^{-2}	4.21	32	3×1760
12	1.65×10^{-7}	1.62×10^{-5}	27.02	32	3×2912

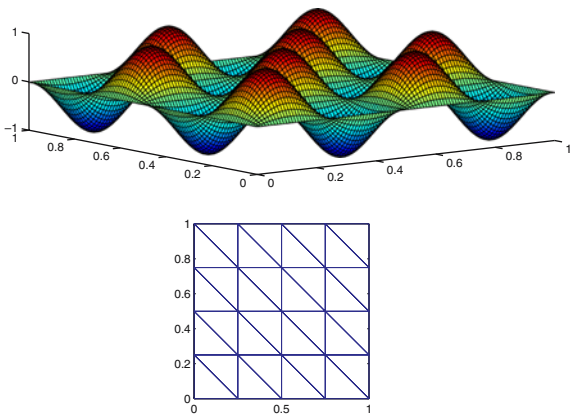

Fig. 2 3D-Plot and grid for example (7)

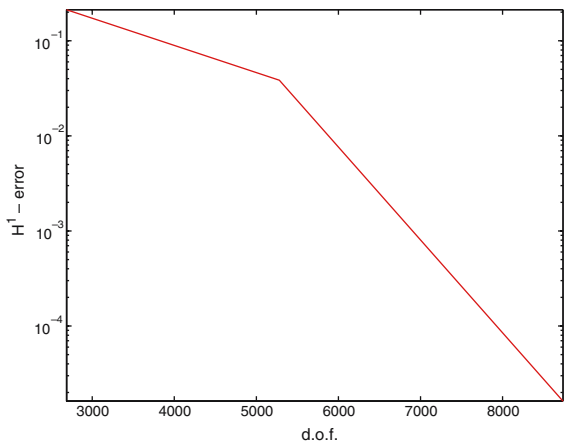

Fig. 3 H^1-error versus d.o.f. for example (7)

$$u(x,y) = \sin 4\pi x \, \sin 4\pi y. \tag{7}$$

In Table 1, we observe the high accuracy of our spectral least-squares scheme. The convergence rates in N (approximately given by the logarithm of the quotient between the errors in the H^1 norm divided by the logarithm of the quotient between step sizes in N), remain nearly constant for $N = 6, 9$. From $N = 9$ to $N = 12$, there is a jump in convergence speed since now the polynomial approximation is able to resolve the pronounced oscillation of the solution u. In Fig. 2 we plot the approximate solution and the adaptive grid after one step of refinement. We start with a regular grid of eight elements, where all elements are refined. Hence, we obtain a triangulation with 32 elements. The high spectral accuracy can also be seen from Fig. 3 where we plot the H^1-error versus the d.o.f. In the second example we consider the Poisson equation for a solution that has a steep gradient in the corner $(0, 0)$, given by

$$u(x,y) = \exp(-\alpha(x+y)), \tag{8}$$

where $\alpha = 100$. Numerical results for $N = 3, 6$ are given in Table 2. The convergence rates are defined with respect to N. Due to the corner singularity the convergence rate is only slightly increasing. Here we also present the number of grid refinements (#ref.) and number of collocation conditions (#coll.). The collocation conditions result from collocating the system (3–5) and then requiring boundary/interface conditions on each triangular element. The d.o.f are three times the number of collocation nodes due to the functions u, v_1, v_2. The number of collocation nodes is given by the number of nodes on each triangular element, i.e., $(N+1)(N+2)/2$, multiplied by the number of elements. Obviously, there are more collocation conditions than d.o.f and hence we obtain an overdetermined system. The well-adapted local grid for $N = 3$ after seven levels of refinement is displayed in Fig. 4. In Fig. 5 we give an error plot in the discrete H^1-norm for $N = 6$ and 3,4 resp. 5 steps of refinement. The error is plotted on the line $y = x$ showing a strong increase near the singularity. We consider two additional examples introduced by [43] for a numerical coordinate transformation based on adaptivity. First we once more solve the Poisson problem with

Table 2 Numerical results for examples (8)–(10)

Ex.	N	$\|\|u-u^h\|\|$	$\|\|u-u^h\|\|_1$	Conv. rates	#ref.	#el.	#coll.	d.o.f.
(8)	3	1.34×10^{-4}	1.06×10^{-1}	–	7	206	8652	6180
	6	2.56×10^{-5}	1.38×10^{-2}	2.94	5	48	5040	4032
(9)	3	5.23×10^{-3}	3.76×10^{-1}	–	5	1016	42672	30480
	6	4.61×10^{-5}	9.20×10^{-3}	5.35	3	340	35700	28560
(10)	3	2.17×10^{-1}	2.84×10^{0}	–	4	1465	61530	43950
	6	3.58×10^{-6}	7.75×10^{-4}	11.84	3	450	47250	37800

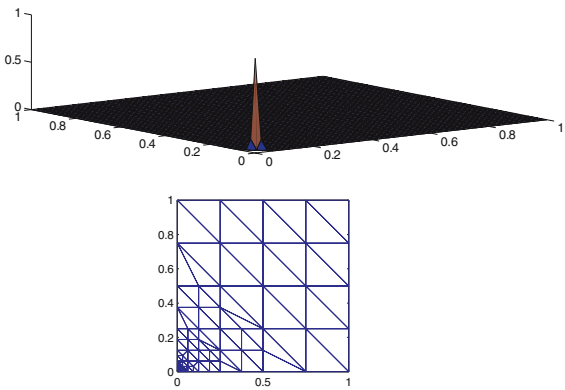

Fig. 4 3D-Plot and grid for example (8)

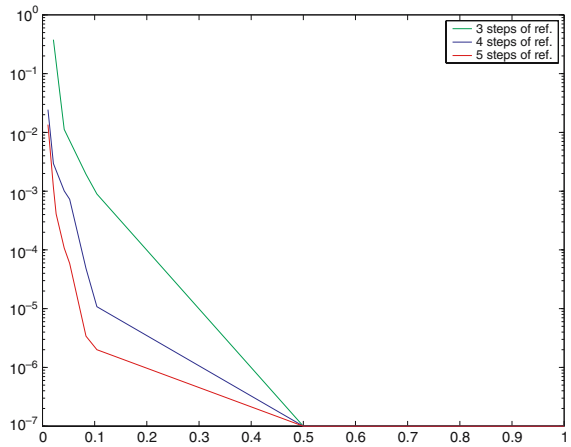

Fig. 5 H^1-error profile at $y = x$

$$u(x,y) = \tanh\left[\frac{1}{\epsilon}\left(\frac{1}{16} - \left(x-\frac{1}{2}\right)^2 - \left(y-\frac{1}{2}\right)^2\right)\right]. \tag{9}$$

For small ϵ, this function resembles a top hat with a front located on the circle $(x-1/2)^2+(y-1/2)^2 = 1/16$. Here we choose $\epsilon = 0.05$. Numerical results are given in Table 2, the 3D-Plot and the most refined grid are displayed in Fig. 6. We finally consider a convection-diffusion problem where (5) reads as:

$$-\epsilon\left(v_{1,x} + v_{2,y}\right) + v_1 = f,$$

where

$$f(x,y) = \epsilon\omega^2\left[1 - \mathrm{e}^{(x-1)/\epsilon)}\right]\sin\omega y$$

with boundary conditions corresponding to the exact solution

$$u(x,y) = \left[1 - \mathrm{e}^{(x-1)/\epsilon)}\right]\sin\omega y. \tag{10}$$

Here, ω and ϵ are prescribed positive constants with $\epsilon \ll 1$. In our computations, we choose $\omega = \pi$ and $\epsilon = 0.05$. The exact solution has a steep front along the line $x = 1$ whose gradient is independent of y. Table 2 contains the numerical results plotted in Fig. 7. Obviously, the least-squares scheme yields a stable method for the solution of this convection-diffusion problem. For one-dimensional singular perturbation problems this was already observed in [6]. Here this can be observed from Fig. 8 where we plot the exact versus the numerical solution ($N = 6$) at $y = 0.5$ for 0 and 1 steps of refinement. We start with a regular decomposition of eight triangular elements. Without refinement we observe (as expected) a jump at $x = 0.5$.

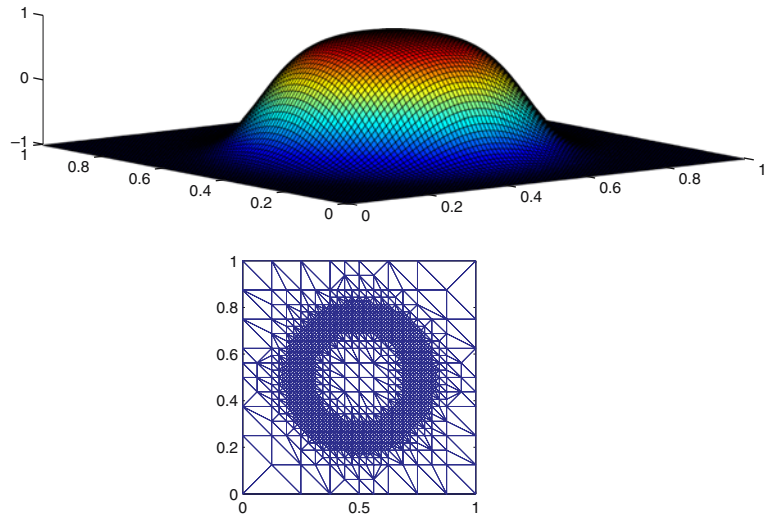

Fig. 6 3D-Plot and grid for example (9)

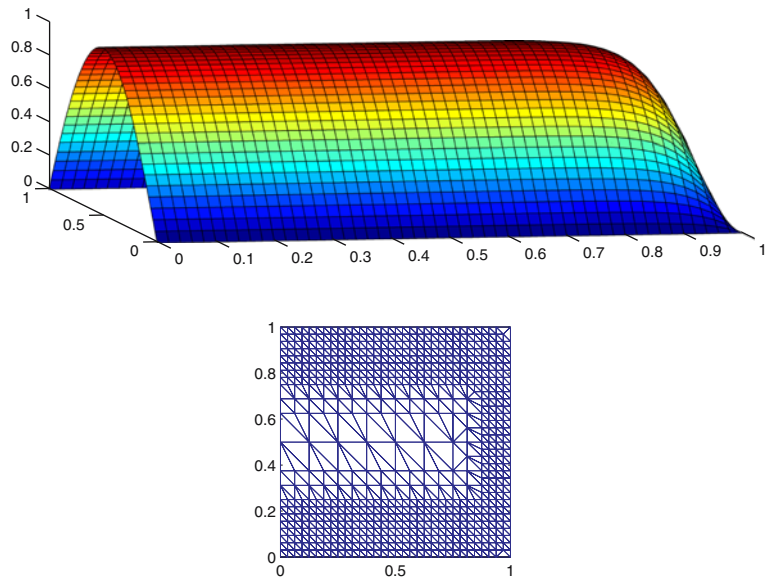

Fig. 7 3D-Plot and grid for example (10)

For all examples, we also present convergence rates, number of elements, number of collocation conditions and d.o.f. It is interesting to note that for $N = 6$ we have fewer degrees of freedom with much higher accuracy. Hence for the numerical computation we recommend $N = 6$. Because of the singular behavior of the solutions, numerical tests show that it does not pay off to use higher-degree polynomials. Furthermore, we observe that the number of total collocation conditions is about 25% higher than degrees of freedom. Since the systems are symmetric and positive definite, iterative solvers like (preconditioned) conjugate-gradient methods could be used, although they are not optimal. By numerical experiments we found that it is much faster to employ some direct solver with LU-factorization for sparse (banded) matrices. For instance, we recommend the UMFPACK Routine from the MATLAB 7.1 library developed by Tim Davis, Univ. of Florida [23, 24].

5 The Navier–Stokes equations

In order to apply the least-squares method, the Navier–Stokes problem is transformed into an equivalent first-order system of partial differential equations. This is accomplished by introducing the vorticity $\omega = \nabla \times \mathbf{u}$ as an auxiliary variable. By using the identity

$$\nabla \times \nabla \times \mathbf{u} = -\Delta \mathbf{u} + \nabla(\nabla \cdot \mathbf{u}),$$

and by using the incompressibility constraint $\nabla \cdot \mathbf{u} = 0$, we obtain

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p + \nu \nabla \times \omega = \mathbf{f} \quad \text{in } \Omega, \tag{11}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \tag{12}$$

$$\omega - \nabla \times \mathbf{u} = 0 \quad \text{in } \Omega, \tag{13}$$

where $\mathbf{u}^{\mathrm{T}} = [u, v]$ denotes the velocity vector, p the pressure, $\mathbf{f}^{\mathrm{T}} = [f_u, f_v]$ the forcing term, and ν the kinematic viscosity. Here it is assumed that the density equals unity. Spectral least-squares methods for the Stokes and Navier–Stokes equations were first introduced in [34, 35, 38, 41]. Reference [40] investigated least-squares collocation schemes for the Navier–Stokes equations. Here we extend these methods to a triangular decomposition of the domain combined with adaptive mesh refinement. Since our least-squares method is stable for singular perturbation problems, it can be expected to work well for both explicit and implicit time-integration schemes. As proposed in [38], we apply a θ-integration scheme in time combined with the Picard linearization to the momentum equation of the unsteady Navier–Stokes equations. The subscript "0" corresponds to the results obtained at a previous integration time step. Now the momentum equation reads as follows:

$$\frac{\mathbf{u} - \mathbf{u}_0}{\Delta t} + \theta \left(\mathbf{u}_0 \cdot \nabla \mathbf{u} + \nabla p + \nu \nabla \times \omega - \mathbf{f}\right) \tag{14}$$

$$= (\theta - 1)\left(\mathbf{u}_0 \cdot \nabla \mathbf{u}_0 + \nabla p_0 + \nu \nabla \times \omega_0 - \mathbf{f}_0\right). \tag{15}$$

By taking $\theta = 1$, the time integration is carried out by the backward Euler method, which is only first-order accurate in time. The second-order time integration of Crank–Nicolson can be obtained by setting $\theta = 1/2$. Since the Crank–Nicolson scheme has no damping, one often takes $\theta = 1/2 + O(\Delta t)$. The temporal accuracy remains second-order, and adding the small factor Δt effectively dampens the small waves in spectral-element simulations. Hence, in order to obtain time-accurate solutions, one should use $\theta = 1/2 + O(\Delta t)$. The θ-scheme is unconditionally stable for $1/2 \leq \theta \leq 1$. Here we only consider stationary problems where it is recommended to use backward Euler ($\theta = 1$) with a large time step to obtain steady-state solutions. Now the complete system for each time step can explicitly be written as

$$\begin{bmatrix} \frac{1}{\Delta t} + \theta u_0 \frac{\partial}{\partial x} + \theta v_0 \frac{\partial}{\partial y} & 0 & \theta \nu \frac{\partial}{\partial y} & \theta \frac{\partial}{\partial x} \\ 0 & \frac{1}{\Delta t} + \theta u_0 \frac{\partial}{\partial x} + \theta v_0 \frac{\partial}{\partial y} & -\theta \nu \frac{\partial}{\partial x} & \theta \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & -\frac{\partial}{\partial x} & 1 & 0 \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ \omega \\ p \end{bmatrix} = \mathbf{F},$$

where

$$\mathbf{F} = \begin{bmatrix} f_u + \frac{u_0}{\Delta t} + (\theta - 1)\left\{u_0 \frac{\partial u_0}{\partial x} + v_0 \frac{\partial u_0}{\partial y} + \nu \frac{\partial \omega_0}{\partial y} + \frac{\partial p_0}{\partial x}\right\} \\ f_v + \frac{v_0}{\Delta t} + (\theta - 1)\left\{u_0 \frac{\partial v_0}{\partial x} + v_0 \frac{\partial v_0}{\partial y} - \nu \frac{\partial \omega_0}{\partial x} + \frac{\partial p_0}{\partial y}\right\} \\ 0 \\ 0 \end{bmatrix}.$$

Table 3 Numerical results for examples (16–18)

N	$\|\|u-u^h\|\|_1$	$\|\|v-v^h\|\|_1$	Conv. rates v	$\|\|p-p^h\|\|$	Conv. rates p	#el.	d.o.f.
3	1.68×10^{-1}	1.46×10^{-1}	—	1.60×10^{-2}	—	8	4×80
6	6.39×10^{-3}	6.37×10^{-3}	4.52	3.41×10^{-4}	5.55	8	4×224
9	2.30×10^{-6}	4.02×10^{-6}	18.17	1.52×10^{-7}	19.03	8	4×440
12	2.21×10^{-8}	7.83×10^{-9}	21.69	4.06×10^{-9}	12.59	8	4×728

Table 4 The regularized cavity flow (19), Re = 100

N	M_1	M_2	#el.	#coll.	d.o.f.
3	8.45×10^{-2}	11.29	429	29900	17159
6	8.36×10^{-2}	13.32	101	16513	11311
[45]	8.34×10^{-2}	13.34	1	4802	4802

In the spectral scheme, all four functions u, v, ω, p are approximated by polynomials in $\mathbb{P}_N$. The partial derivatives on triangles are calculated as before. At the interface between triangular elements, we require continuity of both the functions and normal derivatives of u, v. For p we only require continuity and for ω we do not explicitly require interface conditions. The above system of differential equations, together with the boundary and interface conditions, leads to an overdetermined system that can be efficiently solved by least squares. After eliminating the constant mode for the pressure, we obtain a symmetric and positive-definite system.

We consider examples also treated in [44] for splitting schemes. First we introduce a smooth example where the velocity components u, v and the pressure p are given by

$$u(x,y) = \sin\left(\frac{\pi\hat{x}}{2}\right)\cos\left(\frac{\pi\hat{y}}{2}\right), \tag{16}$$

$$v(x,y) = -\cos\left(\frac{\pi\hat{x}}{2}\right)\sin\left(\frac{\pi\hat{y}}{2}\right), \tag{17}$$

$$p(x,y) = \frac{1}{4}\left(\cos\pi\hat{x} + \cos\pi\hat{y}\right) + 10(\hat{x}+\hat{y}), \tag{18}$$

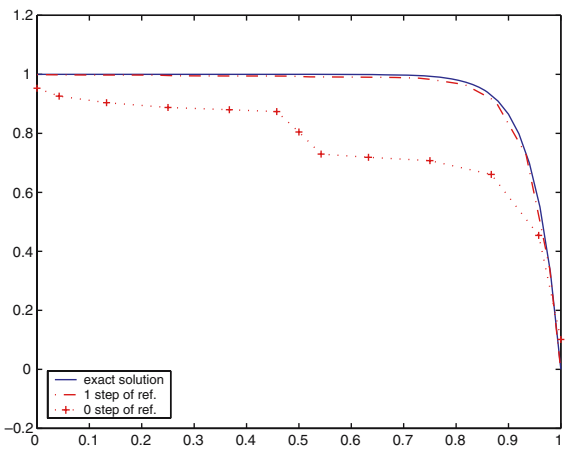

Fig. 8 2D cross-section plot at $y = 0.5$

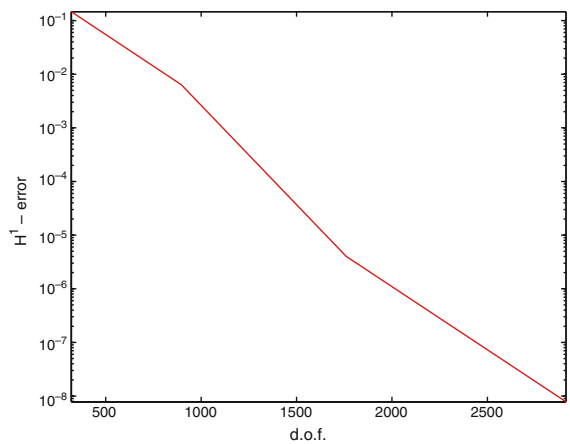

Fig. 9 H^1-error in v versus d.o.f. for example (16–18)

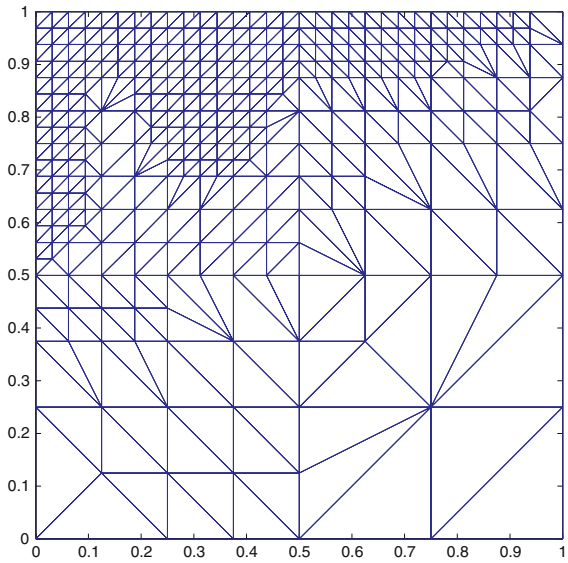

Fig. 10 Adapted grid for the regularized driven-cavity flow 19)

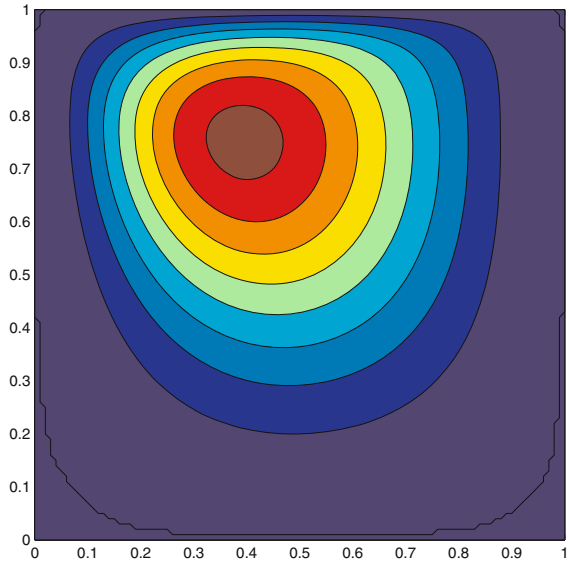

Fig. 11 Stream function for the regularized driven-cavity flow (19)

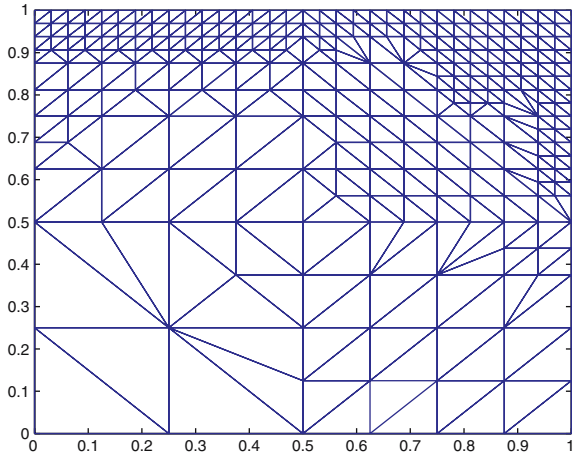

Fig. 12 Adapted grid for the regularized driven-cavity flow (20)

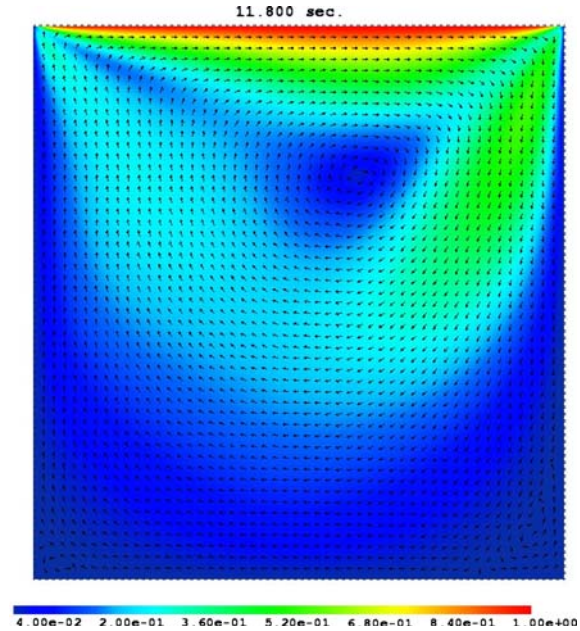

Fig. 13 Velocity profile for the regularized cavity flow (20)

where $\hat{x} = 2x - 1$, $\hat{y} = 2y - 1$ are the coordinates in $[-1, 1]$. For a Reynolds number of $\mathrm{Re} = 1/\nu = 100$, we calculated the discrete L^2 and H^1 errors on a triangular grid with 8 elements. For increasing N, we observe from Table 3 and Fig. 9 high spectral accuracy. The convergence rates given for v and p show exponential convergence. Furthermore, we consider the regularized cavity flow (see [44–46]) where the fluid velocity on the edge $y = 1$ is given by

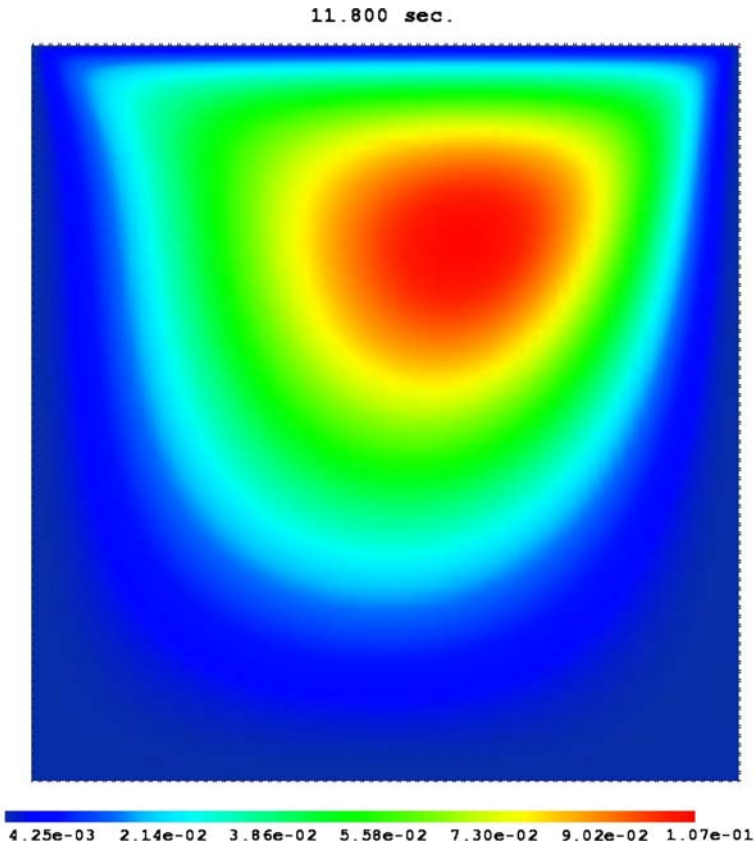

Fig. 14 Stream function for the regularized cavity flow (20)

$$u(x,1) = -16x^2(1-x)^2, \quad v(x,1) = 0, \tag{19}$$

where $u = v = 0$ at the other three edges. The source term $\mathbf{f}$ is identical to zero. After steady state is reached, we also calculate the stream function ψ by solving the equation

$$\Delta\psi = -\omega \quad \text{in } \Omega = (0,1)^2.$$

Here we also employ the error estimator (6) where we take the arithmetic mean of both velocity components. The refinement strategy is as before, and the error is balanced over the elements. On each level of refinement, we iterate with the above time-integration scheme. In order to compare our results with the results of [45], we calculated the maximal value of ψ on the collocation points of Ω, denoted by M_1. Furthermore, we computed the maximal value of ω on the edge $y = 1$. This value is denoted by M_2. In Table 4 we present results for $\mathrm{Re} = 100$. Here we employed four steps of refinement for $N = 3$ and two steps of refinement for $N = 6$. The adapted grid and the contour lines of the stream function ($\mathrm{Re} = 100$) are plotted in Figs. 10 and 11. The numerical results are in good agreement with results obtained in [45] where the vorticity–stream-function formulation was considered with a Chebychev expansion of degree $N = 48$ on the unit square (one element). In the approach of Botella, there are only two unknown functions and one element ($N = 48$), which results in $2 \times 2,401$ d.o.f. For a similar accuracy, the least-squares approach requires 101 elements ($N = 6$), which results in $4 \times 2,828$ degrees of freedom. Hence for the regularized cavity flow, the adaptive spectral least-squares approach is more expensive. But for more complex geometries and singularities, our adaptive approach is more favorable. Furthermore, we solved a tougher regularized problem where

$$u(x,1) = \begin{cases} \tanh(\beta x) & \text{for } 0 \le x \le 0.5, \\ -\tanh(\beta(x-1)) & \text{for } 0.5 \le x \le 1 \end{cases} \tag{20}$$

with $\beta = 50$. In Fig. 12 we present the adapted grid after four levels of refinement. In Figs. 13 and 14 we show the velocity profile and the stream function. The results are in good agreement with the benchmarks of Ghia and Ghia [47]. For more severe test cases (e.g., lid-driven-cavity flow with high Reynolds numbers) our code has to be optimized and more efficient solvers have to be implemented. This will be the subject of a future project.

6 Conclusion

A least-squares spectral collocation scheme for the incompressible Navier–Stokes equations has been presented. Grid refinement is performed by means of adapted triangular elements. On each triangle, Fekete nodes are employed. The solution technique only involves symmetric positive-definite linear systems. The method was successfully applied to some regularized driven-cavity-flow problems. Numerical simulations confirm the high accuracy of the proposed spectral least-squares scheme.

References

1. Canuto C, Hussaini MY, Quarteroni A, Zang TA (1989) Spectral methods in fluid dynamics, Springer Series in Computational physics. Springer-Verlag, Berlin-Heidelberg-New York
2. Gottlieb D, Orszag SA (1977) Numerical analysis of spectral methods: theory and applications. CBMS-NSF Regional Conference Series in Applied Mathematics No. 26, SIAM, Philadelphia
3. Orszag SA (1980) Spectral methods for problems in complex geometries. J Comput Phys 37:70–92
4. Deville MO, Fischer PF, Mund EH (2002) High-order methods for incompressible fluid flow. Cambridge monographs on applied and computational mathematics. Cambridge University Press
5. Gerritsma MI, Proot MJ (2002) Analysis of a discontinuous least-squares spectral element method. J Sci Comput 17:297–306
6. Heinrichs W (2003) Least-squares spectral collocation for discontinuous and singular perturbation problems. J Comput Appl Math 157:329–345
7. Eisen H, Heinrichs W (1992) A new method of stabilization for singular perturbation problems with spectral methods. SIAM J Numer Anal 29:107–122
8. Heinrichs W (1992) A stabilized multidomain approach for singular perturbation problems. J Sci Comput 7:95–125
9. Heinrichs W (1994) Spectral viscosity for convection dominated flow. J Sci Comput 9:137–148
10. Henderson RD (1999) Adaptive spectral element methods for turbulence and transition. In: Barth TJ, Deconinck H (eds) High order methods for computational physics. Springer, Berlin, pp 225–324
11. Verfürth R (1996) A review of a posteriori error estimation and adaptive mesh refinement. Wiley-Teubner, Chichester-New York-Stuttgart
12. Solin P (2005) Partial differential equations and the finite element method. Wiley, New York
13. Heinrichs W (1998) Spectral collocation on triangular elements. J Comput Phys 145:743–757
14. Heinrichs W, Loch B (2001) Spectral schemes on triangular elements. J Comput Phys 173:279–301
15. Karniadakis G, Sherwin SJ (1999) Spectral/hp element methods for CFD numerical mathematics and computation. Oxford Univ. Press, London
16. Mavriplis C, Van Rosendale A (1993) Triangular spectral elements for incompressible fluid flow. ICASE Rep pp 93–100
17. Bos L (1983) Bounding the Lebesgue function for Lagrange interpolation in a simplex. J Approx Theory 38:43–59
18. Chen Q, Babuška I (1995) Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. Comput Meth Appl Mech Engng 128:405–417
19. Hesthaven JS (1998) From electrostatics to almost optimal nodal sets for polynomial interpolation. SIAM J Numer Anal 35:655–676
20. Taylor MA, Wingate BA, Vincent RE (2000) An algorithm for computing Fekete points in the triangle. SIAM J Numer Anal 38:1707–1720
21. Pasquetti R, Rapetti F (2004) Spectral element methods on triangles and quadrilaterals: comparisons and applications. J Comput Phys 198:349–362
22. Dubiner M (1993) Spectral methods on triangles and other domains. J Sci Comput 6:345–390
23. Davis TA (2005) UMFPACK – Version 4.4 user guide. http://www.cise.ufl.edu/research/sparse/umfpack/: Dept. of Computer and Information Science and Engineering - Univ. of Florida

24. Davis TA (2004) UMFPACK – an unsymmetric-pattern multifrontal method with a column pre-ordering strategy. ACM Trans Math Software 30(2):196–199
25. Deang JM, Gunzburger MD (1998) Issues related to least-squares finite element methods for the Stokes equations. SIAM J Sci Comput 20:878–906
26. Jiang B-N (1998) On the least-squares method. Comput Meth Appl Mech Engng 152:239–257
27. Jiang B-N, Chang CL (1990) Least-squares finite elements for the Stokes problem. Comput Meth Appl Mech Engng 78:297–311
28. Jiang B-N (1992) A least-squares finite element method for incompressible Navier–Stokes problems. Int J Numer Meth Fluids 14:843–859
29. Jiang B-N, Povinelli L (1990) Least-squares finite element method for fluid dynamics. Comput Meth Appl Mech Engng 81:13–37
30. Bernardi C, Canuto C, Maday Y (1986) Generalized inf-sup condition for Chebyshev approximations to Navier–Stokes equations. C. R. Acad Sci Paris 303:971–974
31. Bernardi C, Maday Y (1992) Approximations spectrale de problémes aux limites elliptiques. Springer-Verlag, Berlin
32. Rønquist E (1988) Optimal spectral element methods for the unsteady three dimensional incompressible Navier–Stokes Equations. Ph. D. thesis, Massachusetts Institute of Technology, Cambridge
33. Heinrichs W (1993) Splitting techniques for the pseudospectral approximation of the unsteady Stokes equations. SIAM J Numer Anal 30(1):19–39
34. Proot MJ, Gerritsma MI (2002) A least-squares spectral element formulation for the Stokes problem. J Sci Comput 17:285–296
35. Proot MJ, Gerritsma MI (2002) Least-squares spectral elements applied to the Stokes problem. J Comput Phys 181: 454–477
36. Pontaza JP, Reddy JN (2003) Spectral/*hp* least-squares finite element formulation for the Navier–Stokes equations. J Comput Phys 190:523–549
37. Pontaza JP, Reddy JN (2004) Space-time coupled spectral/*hp* least-squares finite element formulation for the incompressible Navier–Stokes equations. J Comput Phys 197:418–459
38. Proot MJ, Gerritsma MI (2005) Application of the least-squares spectral element method using Chebyshev polynomials to solve the incompressible Navier–Stokes equations. Numer Algor 38:155–172
39. Pontaza JP, Reddy JN (2006) Least-squares finite element formulations for viscous incompressible and compressible fluid flows. Comput Meth Appl Mech Engng 195:2454–2494
40. Heinrichs W (2004) Least-squares spectral collocation for the Navier–Stokes equations. J Sci Comput 21:81–90
41. Gerritsma MI, Phillips TN (1998) Discontinuous spectral element approximation for the velocity-pressure-stress formulation of the Stokes problem. Internat J Numer Meth Engng 43:1401–1419
42. Jiang B-N (1998) The least-squares finite element method, Theory and applications in computational fluid dynamics and electromagnetics. Springer-Verlag, Berlin
43. Mulholland LS, Huang W-Z, Sloan DM (1998) Pseudospectral solution of near-singular problems using numerical coordinate transformations based on adaptivity. SIAM J Sci Comput 19:1261–1289
44. Haschke H, Heinrichs W (2001) Splitting techniques with staggered grids for the Navier–Stokes equations in the 2D case. J Comput Phys 168:131–154
45. Botella O (1997) On the solution of the Navier–Stokes equations using Chebyshev projection schemes with third-order accuracy in time. Comput Fluids 26:107–116
46. Botella O, Peyret R (1998) Benchmark spectral results on the lid-driven cavity flow. Comput Fluids 27:421–433
47. Ghia U, Ghia KN, Shin CT (1982) High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. J Comput Phys 48:387–411