Journal of Computational Physics **173,** 279–301 (2001)
doi:10.1006/jcph.2001.6876, available online at http://www.idealibrary.com on IDEAL®

Spectral Schemes on Triangular Elements

Wilhelm Heinrichs and Birgit I. Loch[1]

Ingenieurmathematik, Universität Essen, 45117 Essen, Germany
E-mail: heinrich@ing-math.uni-essen.de

Received June 22, 1999; revised June 29, 2001

The Poisson problem with homogeneous Dirichlet boundary conditions is considered on a triangle. The transformation from square to triangle is realized by mapping an edge of the square onto a corner of the triangle. Then standard Chebyshev collocation techniques can be implemented. Numerical experiments demonstrate the expected high spectral accuracy for smooth solutions. Furthermore, it is shown that finite difference preconditioning can be successfully employed to construct an efficient iterative solver. Then the convection–diffusion equation is considered. Here finite difference preconditioning with central differences leads to instability. However, using the first-order upstream scheme, we obtain a stable method. Finally, a domain decomposition technique is applied to the patching of rectangular and triangular elements.

Key Words: spectral; collocation; triangle; preconditioning; Poisson; convection–diffusion; domain decomposition.

INTRODUCTION

Pseudospectral collocation methods deliver good approximations for smooth solutions of elliptic partial differential equations. However, they possess a great disadvantage as these methods are confined to rectangles. Additionally, the spectral operator is ill-conditioned compared with finite difference or finite element operators and requires preconditioning to construct an effective iterative solver.

Here, we apply the standard Chebyshev collocation method for solving partial differential equations on certain right triangles. We introduce a transformation between the triangle and the standard square where spectral collocation can be applied. This transformation maps one edge of the square onto one corner of the triangle so that the nonequally spaced collocation points cluster in that corner. Mavriplis and Van Rosendale [13] consider a different approach.

[1] Current address: CPAI, Department of Mathematics, University of Queensland, Brisbane 4072, Australia (bil@maths.uq.edu.au).

0021-9991/01 $35.00

They take half the nodes and then use our transformation only for quadrature. This leads to a reduced computational effort for evaluation of residuals but to a somewhat slower approximation rate since less collocation points and polynomial degrees are used. We do not use quadrature here. The tensor product development and the method of nested quadrature lead to a cost of $O(N^3)$. Karniadakis and Sherwin [12] use our horizontal mapping in their (3.4), (3.5). The only difference is that the triangle lies in $[-1, 1]$ instead of $[0, 1]$. Therefore in the 2D case our results are comparable to their work. In [11] another approach has been examined. Our results are compared to those in [11] and we present a comparison of collocation points.

Our method is then applied to the Poisson equation with homogeneous Dirichlet boundary conditions on a right triangle. It is numerically shown that for smooth solutions high spectral accuracy can be achieved, and as a second example we use a function that possesses poles outside the triangle. Then we introduce a singularity caused by the behavior of the right-hand side leading to a somewhat slower convergence of the approximation. This could be overcome by mapping techniques as in Boyd [1, 2]. Preconditioning by finite differences yields a condition number increasing as $O(N)$.

The convection–diffusion equation is then considered. To overcome the instability for small ϵ we choose N odd (see [6]). Preconditioning by central finite differences yields an unbounded condition number such that an upwind method has to be applied. Dubiner [5]and Sherwin and Karniadakis [16] introduce a different basis which is useful if time-dependent problems have to be treated.

Finally, domain decomposition problems are investigated. The Poisson problem is numerically solved on patchings of rectangular and triangular elements. A Dirichlet–Neumann interface relaxation is iterated until continuity of normal derivatives is achieved. By numerical results the efficiency of this treatment is demonstrated. As in [13] we recommend the use of quadrilateral elements as often as possible (e.g., for the interior of a domain) and triangles for approximation along edges. Delves *et al*. examine and apply domain decomposition effectively in [4].

TRANSFORMATION OF THE RIGHT TRIANGLE

The standard Chebyshev collocation scheme (see [3, 14]) is defined for the nonequally spaced Chebyshev–Gauss–Lobatto nodes $(s_i, t_j) = (\cos \frac{i\pi}{N}, \cos \frac{j\pi}{N})$ on the square $[-1, 1]^2$. Using linear transforms, arbitrary rectangles can be considered. However, if we are interested in triangular domains the mapping is more complicated. In [11] a mapping applying polar coordinate transformation and bending of an edge of the triangle was introduced and analyzed. Numerical results showed the effectiveness of this method. Here we consider a new transformation between the standard square $R = \{(x, y) \mid -1 < x, y < 1\}$ and the right triangle $T = \{(x, y) \mid 0 < x, y < 1, \text{ and } x + y < 1\}$. The original mapping is given in [16] and has been changed for our purposes. The transformation reads as

$$x = \frac{1}{4}(x_R + 1)(1 - y_R), \quad y = \frac{1}{2}(y_R + 1),$$

$$x_R = \frac{2x}{1 - y} - 1, \qquad y_R = 2y - 1$$

and is displayed in Fig. 1.

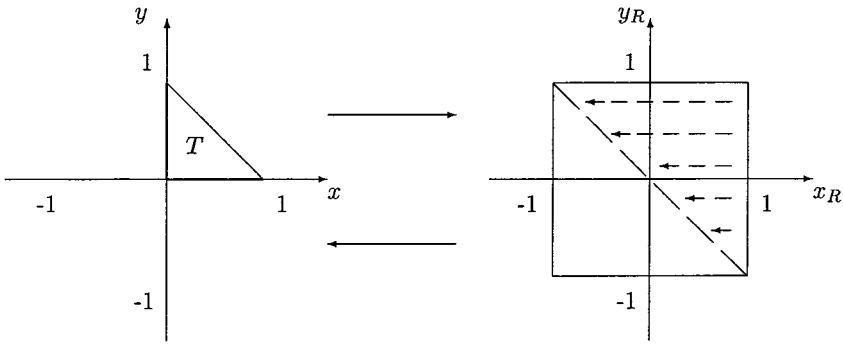

FIG. 1. Horizontal transformation.

We will call this the horizontal transform as every node is actually moved horizontally. The vertical transform is

$$x = \frac{1}{2}(x_R + 1), \quad y = \frac{1}{4}(y_R + 1)(1 - x_R),$$
$$x_R = 2x - 1, \qquad y_R = \frac{2y}{1 - x} - 1$$

and will find an application later on.

This transformation is no longer injective. We will see that this does not affect the accuracy of our approximation. The upper edge of R is mapped onto $P(0, 1)$ on T. As this edge belongs to the border of our domain, boundary conditions are applicable, which are then treated separately. Complex boundary conditions can also be dealt with this way as they become part of the spectral operator.

In addition, partial derivatives must be transformed. Using the horizontal transform we derive

$$u_x = 2u_{x_1} = \frac{4}{1 - y_R} u_{x_R},$$
$$u_{xx} = 4u_{x_1x_1} = \frac{16}{(1 - y_R)^2} u_{x_Rx_R},$$
$$u_y = 2u_{y_1} = 2\frac{x_R + 1}{1 - y_R} u_{x_R} + 2u_{y_R},$$
$$u_{yy} = 4u_{y_1y_1} = 4\frac{(x_R + 1)^2}{(1 - y_R)^2} u_{x_Rx_R} + 8\frac{x_R + 1}{1 - y_R} u_{x_Ry_R} + 8\frac{x_R + 1}{(1 - y_R)^2} u_{x_R} + 4u_{y_Ry_R}.$$

The Laplacian then reads as follows:

$$\Delta u = u_{xx} + u_{yy}$$
$$= 4\frac{4 + (x_R + 1)^2}{(1 - y_R)^2} u_{x_Rx_R} + 8\frac{x_R + 1}{1 - y_R} u_{x_Ry_R} + 8\frac{x_R + 1}{(1 - y_R)^2} u_{x_R} + 4u_{y_Ry_R}.$$

THE POISSON PROBLEM

Numerous spectral algorithms for the numerical simulation of physical phenomena demand the approximate solution of one or more Poisson problems in a bounded domain.

We now study the problem

$$\begin{aligned} \Delta u &= f \quad \text{in } T, \\ u &= 0 \quad \text{on } \partial T, \end{aligned}$$

where ∂T denotes the boundary of T. We apply the standard Chebyshev collocation scheme to the exact solutions

$$u_1(x, y) = xy(e^{x+y} - e) \tag{1}$$

and

$$u_2(x, y) = xy\frac{1 - x - y}{(x + \delta)(y + \delta)}. \tag{2}$$

These functions obviously fulfill the boundary condition. The first function is smooth everywhere whereas the second function possesses poles at $(x, y) = (-\delta, -\delta)$. We choose $\delta = 0.1$, leaving the poles outside the triangle.

Table I shows the discrete L_2 error $E_2 := \|u - u_N\|_2/N$. One observes the exponential decay of the error for Example 1 and slower convergence for Example 2. As we can see, high spectral accuracy can also be reached on the triangle T. We find that the best approximation of the solution is at P(0, 1), as the collocation points cluster there. Figure 2 shows the position of the collocation points for $N = 16$ on the triangle and on the square.

For the smooth solution the horizontal mapping yields much better approximation results than the radial mapping (see Table 1). This results from the clustering of points at (0, 1) for the horizontal mapping, whereas for the radial mapping they group at (0, 0). At (0, 0) the solution u_1 is highly smooth so that a clustering of collocation points in this region is not necessary. It is desirable to have a higher density of points at (0, 1), where u_1 is somewhat steeper.

By the linear transformation

$$T(x, y) = (1 - x - y, x)$$

the corner points of the triangle are rotated in a mathematically positive sense. The transformed collocation points of the horizontal mapping also cluster at (0, 0). Hence the

TABLE I
Error Using Horizontal Transformation and [11]

N	E_2 for u_1	E_2 for u_1 in [11]	E_2 for u_2
4	$1.94 \cdot 10^{-5}$	$1.89 \cdot 10^{-4}$	$1.55 \cdot 10^{-2}$
8	$2.04 \cdot 10^{-11}$	$8.85 \cdot 10^{-7}$	$7.75 \cdot 10^{-4}$
16	$2.12 \cdot 10^{-16}$	$1.84 \cdot 10^{-11}$	$3.34 \cdot 10^{-6}$
32	$4.29 \cdot 10^{-16}$	$1.78 \cdot 10^{-16}$	$6.40 \cdot 10^{-11}$

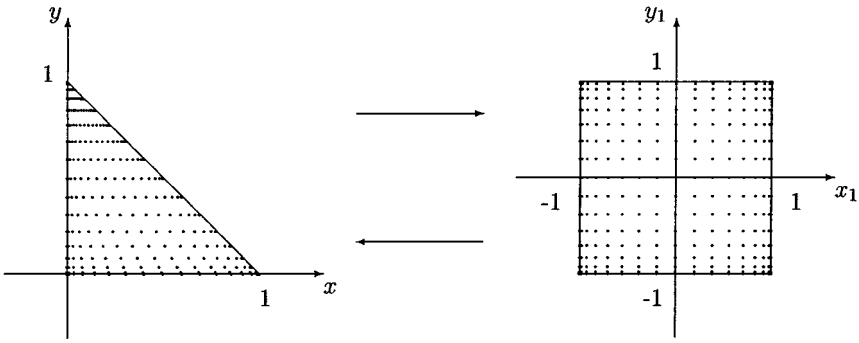

FIG. 2. Positions of the Chebyshev collocation nodes for $N = 16$.

collocation sets of the horizontal and radial mapping are now comparable. The modified horizontal mapping reads

$$\begin{aligned}(x, y) &= \left(1 - \frac{1}{4}(x_R + 1)(1 - y_R) - \frac{1}{2}(y_R + 1), \frac{1}{4}(x_R + 1)(1 - y_R)\right) \\ &= \frac{1}{4}(x_R y_R - x_R - y_R + 1, (x_R + 1)(1 - y_R)) \\ &= \frac{1}{4}(1 - y_R)(1 - x_R, 1 + x_R),\end{aligned}$$

where $x_R, y_R \in [-1, 1]$ denote the rectangular coordinates. The radial mapping for $\theta_1 = \frac{\pi}{2}$, $t_1 = 1$ (see [11]) is given by

$$(x, y) = \frac{r}{\cos\theta + \sin\theta}(\cos\theta, \sin\theta).$$

Using rectangular coordinates x_R, y_R we now write

$$r = \frac{1}{2}(1 - y_R), \quad \theta = \frac{\pi}{4}(1 + x_R).$$

Here $r \in [0, 1]$ where $r = 0$ for $y_R = 1$ and $r = 1$ for $y_R = -1$. As a result of symmetry this yields the same collocation points as the original mapping $r = \frac{1}{2}(1 + y_R)$ but with different numbering.

By this substitution we obtain

$$(x, y) = \frac{1}{4}(1 - y_R)(1 - z_R, 1 + z_R),$$

where

$$z_R = z_R(x_R) = \frac{\sin\frac{\pi}{4}(1 + x_R) - \cos\frac{\pi}{4}(1 + x_R)}{\sin\frac{\pi}{4}(1 + x_R) + \cos\frac{\pi}{4}(1 + x_R)}.$$

Here z_R denotes a one-to-one mapping from $[-1, 1]$ onto $[-1, 1]$. If z_R is equal to the identity, i.e., $z_R(x_R) = x_R$, we obtain the horizontal mapping. Hence the main difference between the horizontal and the radial mapping can be expressed by the one-to-one function

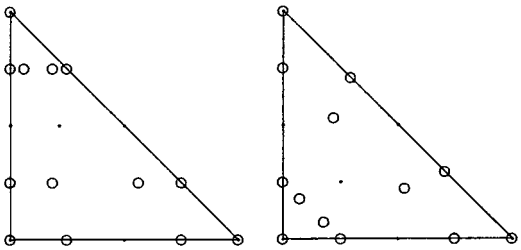

FIG. 3. Collocation points for $N = 2$ (dots) and $N = 3$ (circles) compared to [11].

z_R. Other mappings could also be generated by other choices of z_R. Using the Chebyshev collocation nodes we obtain for $N = 2$

$$x_R \in \{-1, 0, 1\}, \quad z_R \in \{-1, 0, 1\}.$$

Hence for $N = 2$ both mappings yield the same collocation points. This is no longer true for $N \geq 3$. For $N = 3$ we derive

$$x_R \in \{-1, -1/2, 1/2, 1\}, \quad z_R \in \{-1, 1 - \sqrt{2}, \sqrt{2} - 1, 1\},$$

which are obviously different. The collocation points on the original triangles for both mappings ($N = 3$) are plotted in Fig. 3.

Next we consider the singular problem where $f \equiv -1$. We compare the results for $N = 4$, 8, 16, and 32 to those obtained for $N = 36$ at the fixed points displayed in Fig. 4. These points are the collocation nodes for $N = 4$ which are also used for larger N divisible by 4. We expect the error to be smallest close to $y = 1$ because there the collocation nodes cluster. We deal with the following nodes:

$$P_1(0, 0), \quad P_2\left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right), \quad P_3\left(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right), \quad P_4\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right),$$
$$\text{and} \quad P_5\left(-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right).$$

The approximation converges more slowly than in the last examples due to the incompatibility of the differential equation and its boundary condition.

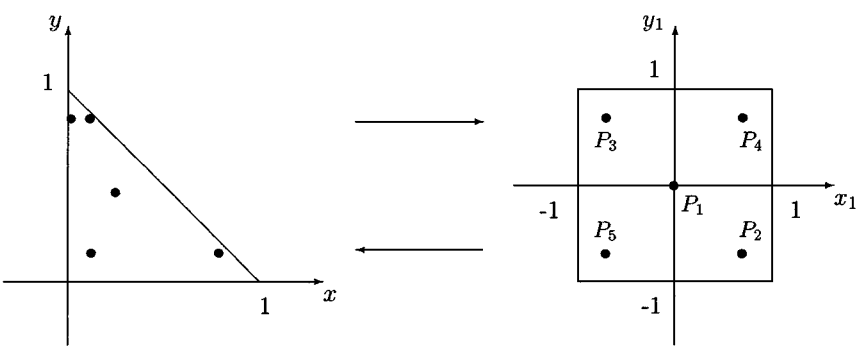

FIG. 4. Positions of the five nodes.

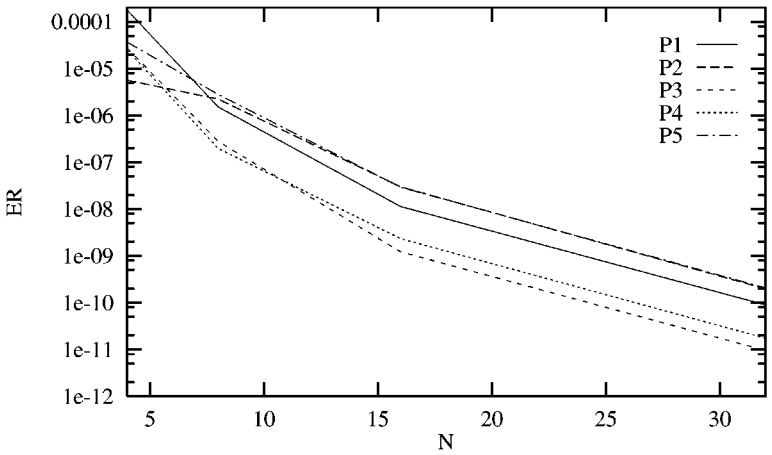

FIG. 5. Poisson problem with constant f.

To obtain an overview we present $ER = |u_N - u_{36}|$, which is the absolute value of the difference for every node, in a diagram (Fig. 5). The results at points between the nodal points are comparable. It can be seen that away from the singularity the error is uniformly distributed.

Next we choose f discontinuous:

$$f(x, y) = \begin{cases} -1 & \text{for } y - x > 0, \\ 0 & \text{for } y - x \leq 0. \end{cases}$$

As Fig. 6 shows the triangle is now bisected. The transformation of the line $y = x$ on the triangle gives the hyperbola $y = 2\frac{x+1}{x+3} - 1$ on the square. The results can be found in Fig. 7. The approximation is relatively inaccurate close to the separating line. Since f is discontinuous the solution of the partial differential equation is no longer smooth and there is no high spectral accuracy. Hence we only have a first-order method. For $f = -1$ the solution is singular in the corners and this slows down the convergence speed of the spectral Chebyshev method. By mapping techniques (see [1, 2]) the convergence speed can be dramatically improved.

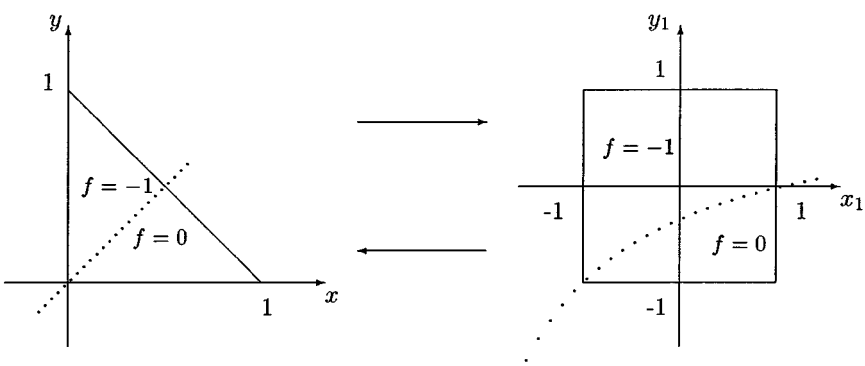

FIG. 6. Transformation of the line.

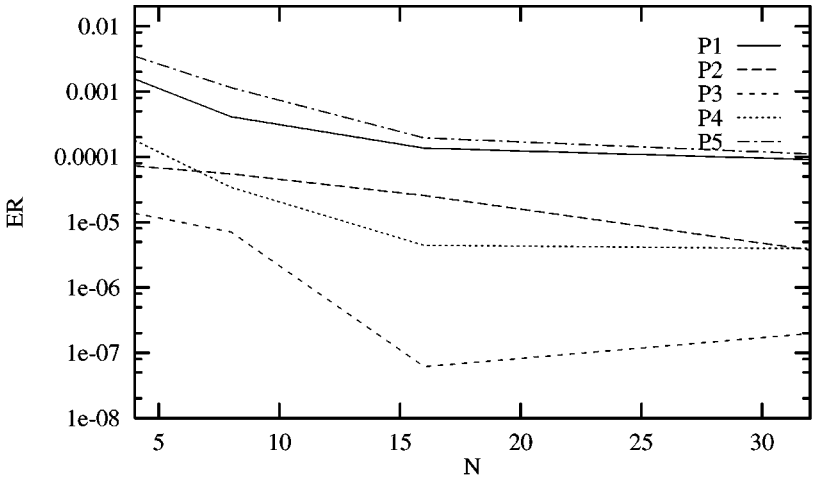

FIG. 7. Poisson problem with discontinuous f.

PRECONDITIONING

We are interested in a good condition number of our spectral operator which does not increase too fast for efficient iterative solvers to be found. Here the maximum eigenvalues of the spectral Laplacian on the triangle increase proportionally to $O(N^8)$ (Fig. 8). On the square one has $O(N^4)$, which is certainly preferrable. We are looking for a preconditioner to improve the condition so that it grows proportionally to $O(N)$ or even independently of

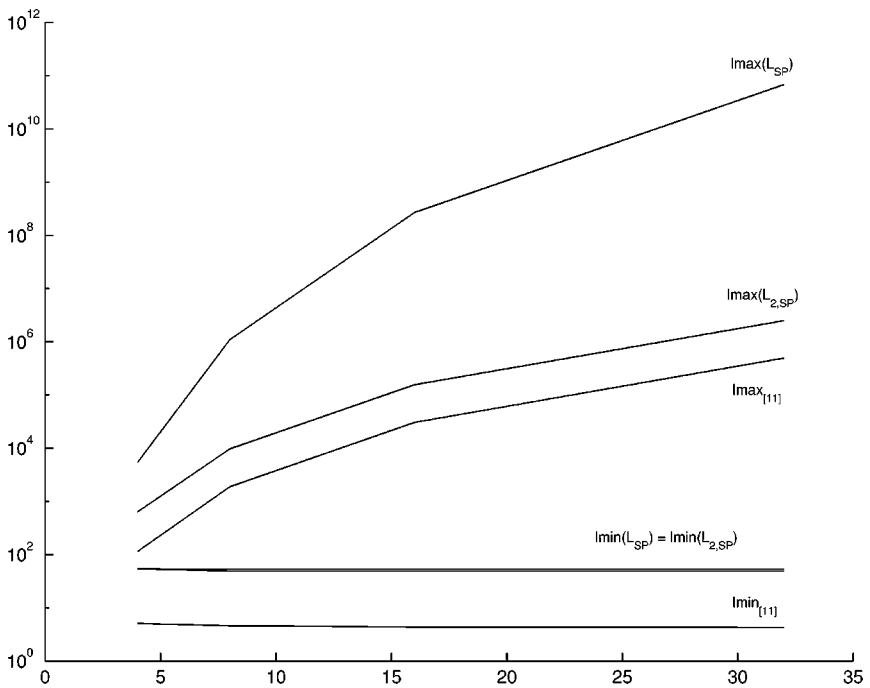

FIG. 8. λ_{max} and λ_{min} for L_{SP}, $L_{2,SP}$, and [11].

TABLE II
Condition Numbers

N	L_{SP}	$L_{2,SP}$	[11]
4	$1.01 \cdot 10^2$	$1.18 \cdot 10^1$	$2.26 \cdot 10^1$
8	$2.23 \cdot 10^4$	$1.83 \cdot 10^2$	$4.15 \cdot 10^2$
16	$5.49 \cdot 10^6$	$2.94 \cdot 10^3$	$7.00 \cdot 10^3$
32	$1.39 \cdot 10^9$	$4.71 \cdot 10^4$	$1.15 \cdot 10^5$

N. A good preconditioner also has to be a good approximation of the inverse of the spectral operator. It can be seen that the condition number is already reduced if we multiply the operator by $(1 - y_R)^2$. The partial derivatives contain this factor in the denominator. For y close to 1 the influence of the appropriate partial derivative is extremely high. The discretized operator is called $L_{2,SP}$. Its condition number increases proportionally to $O(N^4)$. Figure 8 shows $\lambda_{max} := \max\{|\lambda| \mid \lambda \text{ eigenvalue}\}$ and $\lambda_{min} := \min\{|\lambda| \mid \lambda \text{ eigenvalue}\}$ of L_{SP}, $L_{2,SP}$, and [11]. The condition numbers $cond \approx \lambda_{max}/\lambda_{min}$ can be found in Table II. Our results are comparable to those in [11].

We now study the finite difference preconditioner L_{FD}, which is the discretization of the Laplacian by second-order finite differences. The first and second derivatives are

$$w'(s_j) = \frac{1}{2}(-\gamma_{j-1} w(s_{j-1}) - (\gamma_j - \gamma_{j-1}) w(s_j) + \gamma_j w(s_{j+1})),$$
$$w''(s_j) = 2\delta_j(\gamma_{j-1} w(s_{j-1}) - (\gamma_j + \gamma_{j-1}) w(s_j) + \gamma_j w(s_{j+1}))$$

where

$$\delta_j = \frac{1}{s_{j+1} - s_{j-1}},$$
$$\gamma_j = \frac{1}{s_{j+1} - s_j} \quad \text{for } j = 1, \ldots, N-1 \text{ (see [11])}.$$

Table III shows the improved results.

Now we have obtained a condition number scaling as $O(N)$. We could now construct an effective iterative solver.

Figure 9 shows the positions of the eigenvalues for $N = 32$. Their imaginary parts are fairly small and the real parts are contained in [0.5, 3].

TABLE III
$(L_{FD})^{-1} L_{SP}$ and results in [11]

N	λ_{max}	λ_{min}	$cond$	λ_{max}	λ_{min}	$cond$
4	1.73	1.00	1.73	1.71	0.99	1.73
8	2.13	0.89	2.41	2.12	0.99	2.13
16	2.50	0.71	3.53	2.41	0.80	3.01
32	2.91	0.60	4.89	2.83	0.66	4.31

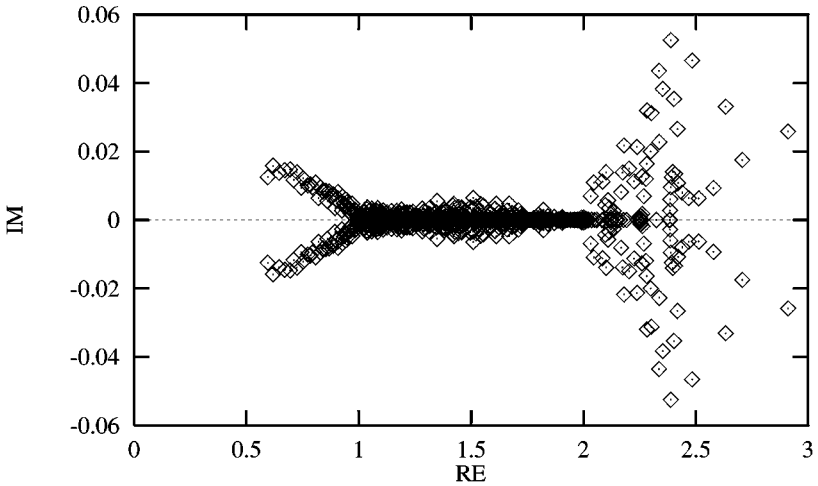

FIG. 9. Eigenvalues of $(L_{FD})^{-1}L_{SP}$ for $N = 32$.

One could apply higher order finite difference methods for an even better condition number. However, this would result in an extended effort for solving the FD problem.

In summary, this transformation between triangle and square gives comparable or better results than the transformation by polar coordinates in [11].

THE CONVECTION–DIFFUSION EQUATION

Modeling of purely convectional or convection dominated processes is a central problem in areas such as meteorology or investigation of aerodynamical or geophysical flows. A model boundary value problem is the convection–diffusion equation

$$\begin{aligned} -\epsilon\Delta u + au_x + bu_y &= f \quad \text{in } T, \\ u &= 0 \quad \text{on } \partial T, \end{aligned}$$

which can be used for describing the expansion of temperature in a fluid. Temperature diffuses uniformly in every direction and we can express this by $-\epsilon\Delta u$. It is also spread by current (called convection) and is described by $au_x + bu_y$ (a and b being the velocities in the x and in y direction).

As usual, ϵ is the viscosity of our material and represents a measure of interior friction. As the partial differential equation is of different type for $\epsilon > 0$ and $\epsilon = 0$ (in the first case it is elliptic and in the latter it is hyperbolic) we talk about singular behavior. In the interior of our domain u_ϵ and u_0 are close together; however when getting nearer to the boundary they differ significantly.

Homogeneous Dirichlet boundary conditions are not, however, applicable to hyperbolic problems. As a result, we now have to deal with boundary layers. Boundary layers are environments where derivatives of u_ϵ scale as $O(\frac{1}{\epsilon})$. Those systems are also called stiff systems. Unphysical oscillations occur in the numerical solution and the discretization is unstable. Figure 10 shows the situation in one dimension (see [9], [10]).

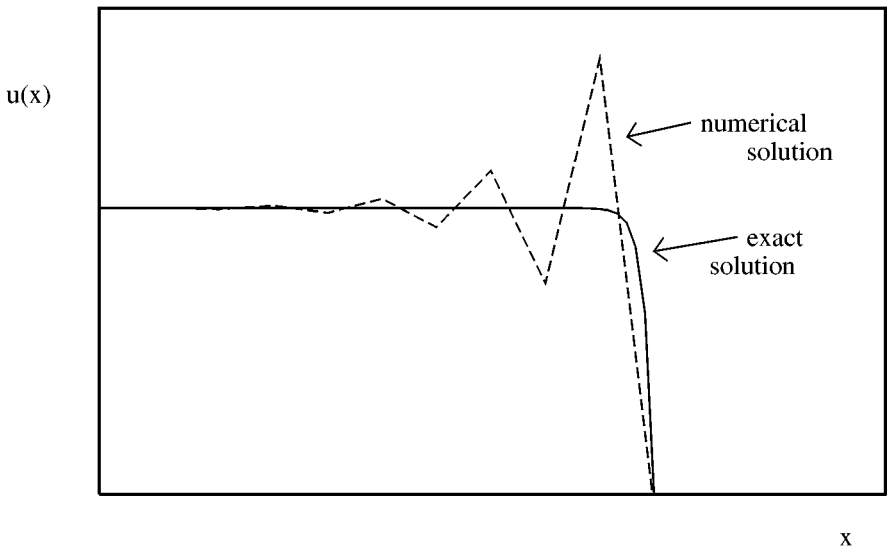

FIG. 10. Boundary layer.

We are looking for a method to resolve the boundary layers. There are schemes that use spectral methods such as adding artificial viscosity, spectral viscosity, or streamline diffusion. However, here we only choose odd N. Oscillations always arise and increase for even N with $\epsilon \ll N^{-2}$ while this is not the case if N is odd.

Table IV contains the discrete L_2 error for decreasing ϵ which develops when discretizing the convection–diffusion equation by spectral collocation. Here we choose $(a, b) = (1, 1)$ and $(-1, 1)$ because these two cases are good representatives of other choices of (a, b). We have tested the algorithm with Example 1. In the case of pure convection ($\epsilon = 0$) the method is unstable. With decreasing ϵ the singular behavior increases and one has to choose a finer grid (larger N) to obtain results comparable to $\epsilon = 1$.

As already mentioned, we now choose N odd, which usually leads to a decreased error. This behavior was analyzed in [6] in one dimension on the square. It can be transferred to the triangle with a few restrictions concerning the choice of parameters. If N is even there exists an interpolation polynomial which fulfills the boundary conditions and whose derivative

TABLE IV
Error for the Convection–Diffusion Equation

		$E2$				
(a, b)	N	$\epsilon = 1$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-6}$	$\epsilon = 0$
(1, 1)	3	$2.75 \cdot 10^{-4}$	$2.21 \cdot 10^{-3}$	$2.15 \cdot 10^{-3}$	$2.15 \cdot 10^{-3}$	$2.15 \cdot 10^{-3}$
	7	$8.75 \cdot 10^{-10}$	$4.08 \cdot 10^{-9}$	$7.77 \cdot 10^{-9}$	$7.77 \cdot 10^{-9}$	$7.77 \cdot 10^{-9}$
	15	$3.77 \cdot 10^{-16}$	$5.32 \cdot 10^{-17}$	$1.81 \cdot 10^{-16}$	$1.50 \cdot 10^{-16}$	$1.14 \cdot 10^{-16}$
	31	$5.08 \cdot 10^{-16}$	$1.72 \cdot 10^{-16}$	$2.27 \cdot 10^{-16}$	$5.64 \cdot 10^{-16}$	$3.67 \cdot 10^{-16}$
(−1, 1)	3	$2.56 \cdot 10^{-4}$	$3.75 \cdot 10^{-3}$	$1.69 \cdot 10^{-1}$	$1.68 \cdot 10^{1}$	$1.09 \cdot 10^{13}$
	7	$8.76 \cdot 10^{-10}$	$5.16 \cdot 10^{-9}$	$1.86 \cdot 10^{-7}$	$1.86 \cdot 10^{-5}$	$1.59 \cdot 10^{8}$
	15	$1.06 \cdot 10^{-16}$	$7.53 \cdot 10^{-17}$	$1.86 \cdot 10^{-16}$	$2.70 \cdot 10^{-14}$	$5.75 \cdot 10^{-1}$
	31	$4.25 \cdot 10^{-16}$	$2.27 \cdot 10^{-16}$	$4.74 \cdot 10^{-16}$	$2.90 \cdot 10^{-14}$	$4.36 \cdot 10^{0}$

vanishes at the collocation points. This polynomial is responsible for the instability. In contrast, if N is odd one finds the proof in [6] that this polynomial does not exist. Apparently, there are parameters (a, b) for which the spectral method is unstable even for odd N. For the stable case (1, 0) we actually have the regular operator $\frac{\partial}{\partial x}$ on the square multiplied by a factor. For $(-1, 1)$ we have such a combination of the first derivatives on the square that there are at least two equal rows in the derivative matrix. The partial derivatives are based on the matrix D_N. Because the collocation points on the square are symmetric (for every positive node we find a corresponding negative one) cancelation occurs in the derivative matrix. The following example for $N = 3$ shows the connection: $u_y = 2\frac{x_R+1}{1-y_R}u_{x_R} + 2u_{y_R}$ yields the derivative matrix

$$\begin{pmatrix} \frac{-s_1}{(1-s_1)^2} - \frac{s_1}{(1-s_1^2)} & \frac{-2(s_1+1)}{(1-s_1)(s_1-s_2)} & \frac{-2}{s_1-s_2} & 0 \\ \frac{-2(s_2+1)}{(1-s_1)(s_2-s_1)} & \frac{-s_2}{(1-s_1)(1-s_2)} - \frac{s_1}{1-s_1^2} & 0 & \frac{-2}{s_1-s_2} \\ \frac{-2}{s_2-s_1} & 0 & \frac{-s_1}{(1-s_2)(1-s_1)} - \frac{s_2}{1-s_2^2} & \frac{-2(s_1+1)}{(1-s_2)(s_1-s_2)} \\ 0 & \frac{-2}{s_2-s_1} & \frac{-2(s_2+1)}{(1-s_2)(s_2-s_1)} & \frac{-s_2}{(1-s_2)^2} - \frac{s_2}{1-s_2^2} \end{pmatrix}.$$

Because $s_1 = -s_2$ (symmetry) the second and third rows of the matrix are equal and so the matrix is singular. The same behavior is displayed at $(-1, 1)$.

For (1, 1) we do not have cancelations and the method is stable. Table IV displays the results; (0, 1) can be stabilized by using the vertical transformation where x and y are exchanged.

Next a constant right-hand side is considered. The differential equation and its boundary condition are not compatible here, i.e.

$$\begin{aligned} -\epsilon\Delta u + au_x + bu_y &= 1 \quad \text{in } T, \\ u &= 0 \quad \text{on } \partial T. \end{aligned}$$

Table V shows the difference ER of u_{36} and u_N at $P_1(0, 0)$. $P_1(0, 0)$ is in the center of the triangle and therefore far away from any boundary. It is the only collocation point (out of P_1–P_5) where stability is achieved for (1, 1) for small ϵ.

TABLE V
Error for Constant f in P_1

		ER				
(a, b)	N	$\epsilon = 1$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-6}$	$\epsilon = 0$
(1, 1)	4	$1.34 \cdot 10^{-4}$	$1.99 \cdot 10^{-1}$	$9.53 \cdot 10^{-2}$	$2.72 \cdot 10^{-2}$	$4.11 \cdot 10^{-1}$
	8	$1.86 \cdot 10^{-6}$	$5.99 \cdot 10^{-2}$	$7.82 \cdot 10^{-2}$	$4.83 \cdot 10^{-3}$	$1.14 \cdot 10^{-1}$
	16	$1.12 \cdot 10^{-8}$	$1.24 \cdot 10^{-3}$	$6.20 \cdot 10^{-2}$	$7.03 \cdot 10^{-3}$	$8.06 \cdot 10^{-1}$
	32	$8.91 \cdot 10^{-11}$	$1.68 \cdot 10^{-7}$	$1.26 \cdot 10^{-2}$	$1.58 \cdot 10^{-3}$	$2.89 \cdot 10^{-2}$
$(-1, 1)$	4	$6.80 \cdot 10^{-5}$	$2.20 \cdot 10^{-1}$	$2.34 \cdot 10^{1}$	$2.35 \cdot 10^{3}$	$1.79 \cdot 10^{15}$
	8	$1.56 \cdot 10^{-6}$	$6.71 \cdot 10^{-2}$	$1.43 \cdot 10^{0}$	$1.41 \cdot 10^{2}$	$1.27 \cdot 10^{15}$
	16	$1.11 \cdot 10^{-8}$	$2.10 \cdot 10^{-3}$	$1.75 \cdot 10^{-3}$	$2.45 \cdot 10^{1}$	$1.27 \cdot 10^{15}$
	32	$8.88 \cdot 10^{-11}$	$1.31 \cdot 10^{-5}$	$6.21 \cdot 10^{-2}$	$2.17 \cdot 10^{0}$	$2.06 \cdot 10^{15}$

TABLE VI
Error for Discontinuous f in P_1

		ER				
(a, b)	N	$\epsilon = 1$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-6}$	$\epsilon = 0$
(1, 1)	4	$1.53 \cdot 10^{-3}$	$1.68 \cdot 10^{-1}$	$1.08 \cdot 10^{-1}$	$2.00 \cdot 10^{-2}$	$3.69 \cdot 10^{-1}$
	8	$4.29 \cdot 10^{-4}$	$2.00 \cdot 10^{-2}$	$7.42 \cdot 10^{-2}$	$1.66 \cdot 10^{-2}$	$8.56 \cdot 10^{-2}$
	16	$1.36 \cdot 10^{-4}$	$3.64 \cdot 10^{-4}$	$4.14 \cdot 10^{-2}$	$2.37 \cdot 10^{-2}$	$8.89 \cdot 10^{-1}$
	32	$9.35 \cdot 10^{-5}$	$5.88 \cdot 10^{-4}$	$1.89 \cdot 10^{-2}$	$1.73 \cdot 10^{-3}$	$3.08 \cdot 10^{-2}$
(−1, 1)	4	$1.70 \cdot 10^{-3}$	$1.39 \cdot 10^{-1}$	$5.46 \cdot 10^{0}$	$5.48 \cdot 10^{2}$	$3.43 \cdot 10^{14}$
	8	$4.94 \cdot 10^{-4}$	$6.86 \cdot 10^{-2}$	$1.93 \cdot 10^{0}$	$1.96 \cdot 10^{2}$	$9.23 \cdot 10^{14}$
	16	$1.53 \cdot 10^{-4}$	$4.65 \cdot 10^{-3}$	$2.35 \cdot 10^{-1}$	$3.61 \cdot 10^{1}$	$5.63 \cdot 10^{14}$
	32	$1.02 \cdot 10^{-4}$	$1.79 \cdot 10^{-3}$	$3.64 \cdot 10^{-2}$	$3.08 \cdot 10^{-1}$	$8.96 \cdot 10^{14}$

A discontinuous right-hand side

$$f(x, y) = \begin{cases} -1 & \text{for } y - x > 0, \\ 0 & \text{for } y - x \le 0 \end{cases}$$

yields an even slower convergence rate than the last example (Table VI).

PRECONDITIONING

For the construction of an effective iterative solver we now examine the condition number of the spectral operator $L_{2,\epsilon}$ of $(1 - y_R)^2(-\epsilon\Delta + au_x + bu_y)$.

Figures 11 and 12 show the positions of the eigenvalues for $\epsilon = 1$ and $\epsilon = 10^{-6}$ for $(a, b) = (1, 1)$, $N = 15$.

Table VII gives λ_{max}, λ_{min}, and *cond* and demonstrates that there is an eigenvalue close to 0 for $(-1, 1)$.

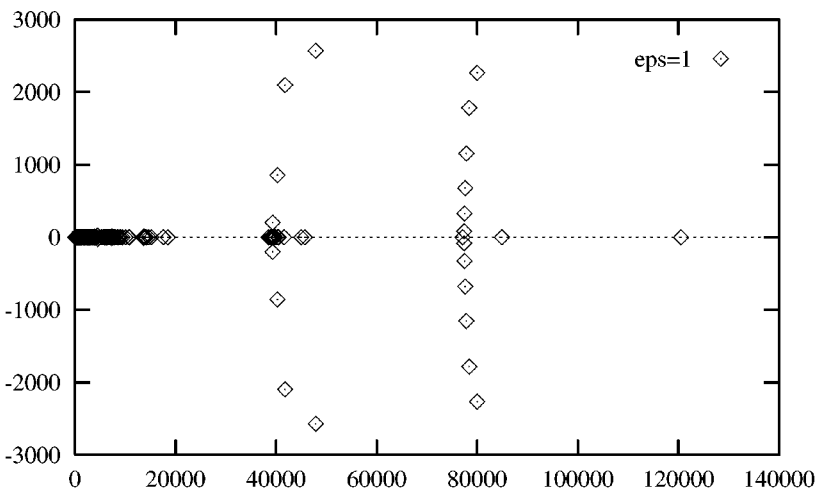

FIG. 11. Eigenvalues of $L^{\epsilon}_{2,SP}$ for $N = 15$, $(a, b) = (1, 1)$.

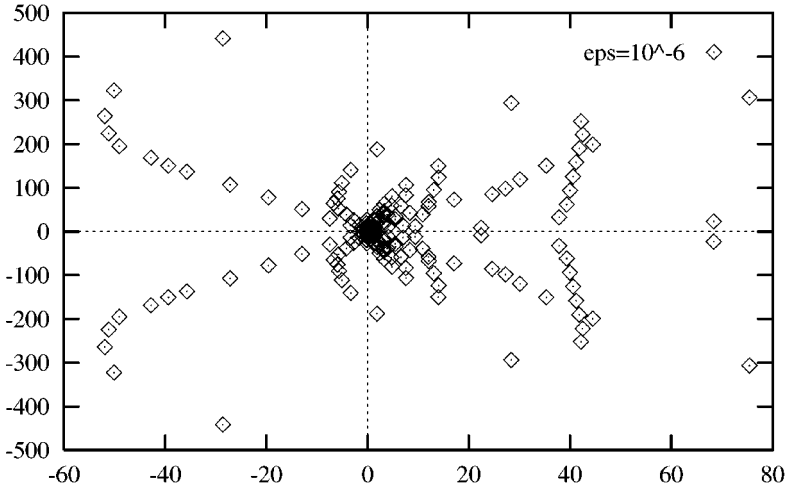

FIG. 12. Eigenvalues of $L^{\epsilon}_{2,SP}$ for $N = 15$, $(a, b) = (1, 1)$.

Applying the inverse of the finite difference operator L^{ϵ}_{FD} as preconditioner, we observe decreased condition number if $\epsilon = 1$ while for small ϵ, λ_{max} is unbounded for (1, 1). This preconditioner obviously does not stabilize.

Figures 13 and 14 show the positions of the eigenvalues. For small ϵ they are relatively densely positioned with few peak values.

In general, finite difference methods applied to singular disturbance problems are stable if the step size $h_i < 2\epsilon$. However, if $h_i \gg \epsilon$ they are unstable. To obtain stability one could increase the number of collocation points, which reduces the step size. A more promising attempt is the application of the upwind method. The first derivatives $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ (the convectional part) are discretized by one-sided stream-directed finite differences while the diffusive part is treated with central differences. We lose one order in convergence but stability is achieved.

We have

$$a \cdot u_x = a \cdot \frac{4}{1 - y_R} \cdot u_{x_R}$$

TABLE VII

$L^{\epsilon}_{2,SP}$

		$\epsilon = 1$			$\epsilon = 0$		
(a, b)	N	λ_{max}	λ_{min}	*cond*	λ_{max}	λ_{min}	*cond*
(1, 1)	3	$2.20 \cdot 10^2$	$5.91 \cdot 10^1$	$3.72 \cdot 10^0$	$8.76 \cdot 10^0$	$2.43 \cdot 10^0$	$3.60 \cdot 10^0$
	7	$5.71 \cdot 10^3$	$5.36 \cdot 10^1$	$1.06 \cdot 10^2$	$8.63 \cdot 10^1$	$7.02 \cdot 10^{-1}$	$1.23 \cdot 10^2$
	15	$1.20 \cdot 10^5$	$5.32 \cdot 10^1$	$2.26 \cdot 10^3$	$4.43 \cdot 10^2$	$1.73 \cdot 10^{-2}$	$2.57 \cdot 10^3$
	31	$2.20 \cdot 10^6$	$5.30 \cdot 10^1$	$4.15 \cdot 10^4$	$1.96 \cdot 10^3$	$4.24 \cdot 10^{-2}$	$4.62 \cdot 10^4$
(−1, 1)	3	$2.22 \cdot 10^2$	$5.82 \cdot 10^1$	$3.82 \cdot 10^0$	$3.27 \cdot 10^0$	$0.00 \cdot 10^0$	
	7	$5.75 \cdot 10^3$	$5.36 \cdot 10^1$	$1.07 \cdot 10^2$	$5.03 \cdot 10^1$	$2.60 \cdot 10^{-16}$	$1.93 \cdot 10^{17}$
	15	$1.21 \cdot 10^5$	$5.32 \cdot 10^1$	$2.27 \cdot 10^3$	$2.86 \cdot 10^2$	$6.61 \cdot 10^{-16}$	$4.33 \cdot 10^{17}$
	31	$2.20 \cdot 10^6$	$5.30 \cdot 10^1$	$4.15 \cdot 10^4$	$1.29 \cdot 10^3$	$6.52 \cdot 10^{-16}$	$1.98 \cdot 10^{18}$

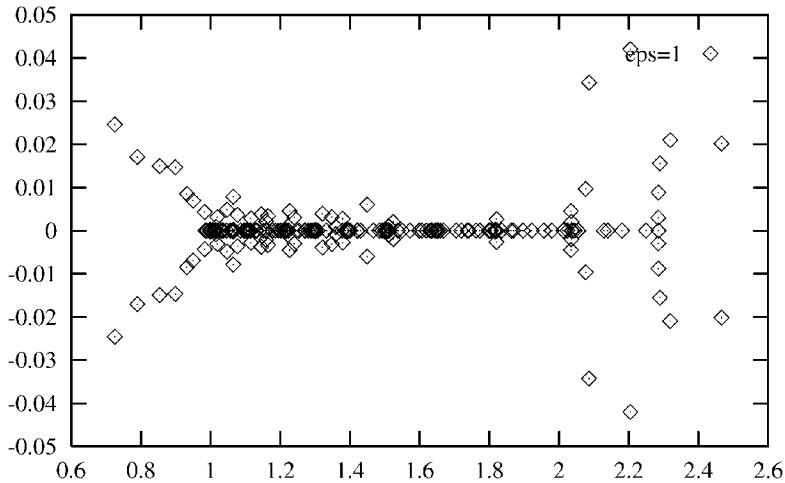

FIG. 13. Eigenvalues of $L_{FD}^{-1}L_{SP}$ for $N = 15$, $(a, b) = (1, 1)$.

and

$$b \cdot u_y = b \cdot \left(2\frac{x_R + 1}{1 - y_R} \cdot u_{x_R} + 2u_{y_R} \right).$$

Depending on the coefficients the derivatives u_{x_R} and u_{y_R} are discretized by left- or right-differences in the stream direction,

$$u_{x_R}(x_i, y_j) \cong \begin{cases} \dfrac{u(x_{i+1}, y_j) - u(x_i, y_j)}{x_{i+1} - x_i} & \text{if } a \geq 0, \\ \dfrac{u(x_i, y_j) - u(x_{i-1}, y_j)}{x_i - x_{i-1}} & \text{if } a < 0 \end{cases}$$

for $i = 0, \ldots, N-1$ or $i = 1, \ldots, N$. We follow a similar process for u_{y_R}. The upwind method is not uniformly convergent. An adaptive refinement might help in this case.

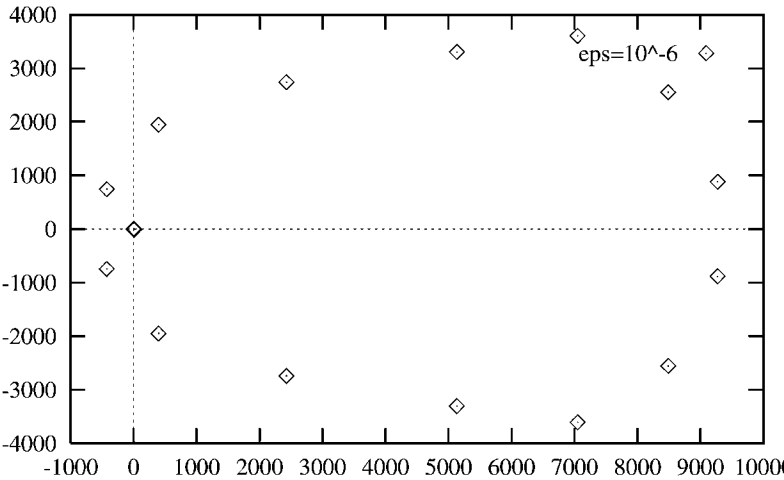

FIG. 14. Eigenvalues of $L_{FD}^{-1}L_{SP}$ for $N = 15$, $(a, b) = (1, 1)$.

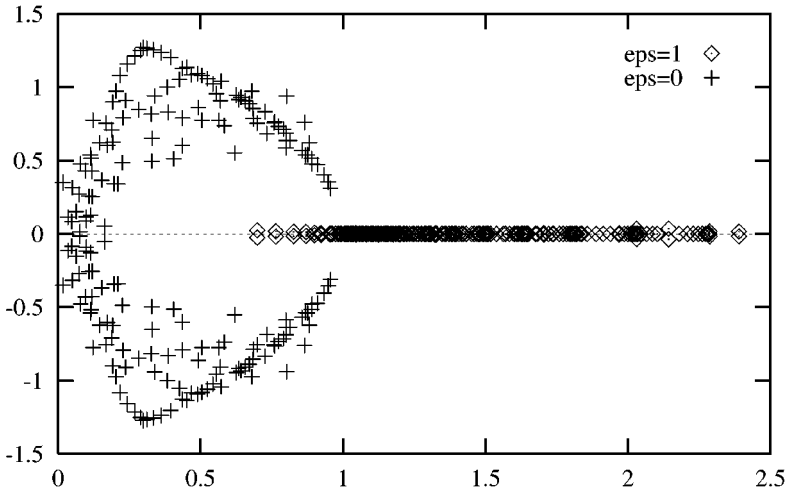

FIG. 15. Eigenvalues of the upstream operator for $N = 15$, $(a, b) = (1, 1)$.

Figures 15 and 16 show that applying the upstream method completely changes the positions of the eigenvalues for small ϵ. They are now complex, bounded, and symmetric. Table VIII gives the numerical results for the upstream scheme.

It is unsatisfactory that there are cases (e.g., $(-1,\ 1)$) in which we cannot achieve stability. One solution may be to introduce an additional collocation point. The system is then overdetermined. This method has been examined and successfully applied on the square in [6]. An alternative method may be the use of staggered grids, which possibly lead to $\lambda_{min} > 0$. Two different sets of grids are used—one for the solution and the other for its derivative. For the advection–diffusion equation, positive results were found in [7].

One of the potential problems with using a tensorial expansion based on nonorthogonal coordinates, as adopted in this work, is that the convection operator has a condition number that grows proportional to N^4. The growth of the condition number can be a problem

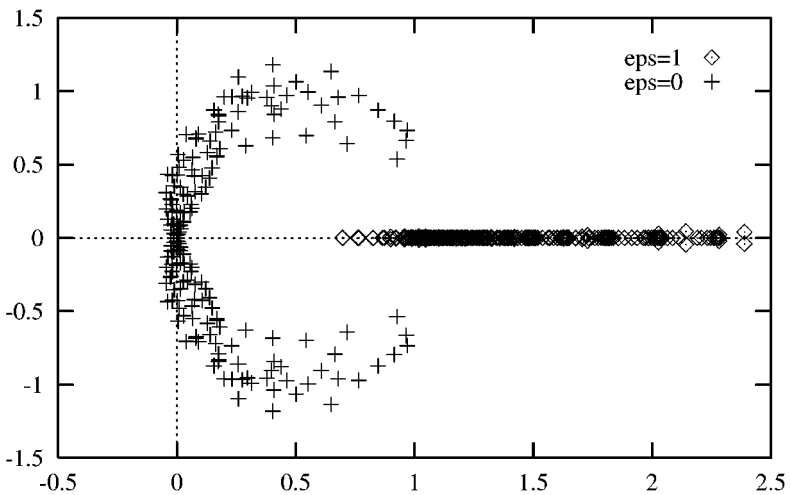

FIG. 16. Eigenvalues of the upstream operator for $N = 15$, $(a, b) = (-1, 1)$.

TABLE VIII
Upstream Method

	(a, b)	$(1, 1)$			$(-1, 1)$		
ϵ	N	λ_{max}	λ_{min}	*cond*	λ_{max}	λ_{min}	*cond*
1	3	1.40	$9.28 \cdot 10^{-1}$	1.51	1.41	$9.35 \cdot 10^{-1}$	1.50
	7	2.06	$8.61 \cdot 10^{-1}$	2.39	2.06	$8.58 \cdot 10^{-1}$	2.40
	15	2.39	$6.99 \cdot 10^{-1}$	3.42	2.39	$6.97 \cdot 10^{-1}$	3.43
	31	2.85	$5.91 \cdot 10^{-1}$	4.82	2.85	$5.91 \cdot 10^{-1}$	4.82
10^{-2}	3	$6.04 \cdot 10^{-1}$	$3.04 \cdot 10^{-1}$	1.98	$2.86 \cdot 10^{-1}$	$1.11 \cdot 10^{-1}$	2.58
	7	1.37	$2.99 \cdot 10^{-1}$	4.56	1.46	$2.67 \cdot 10^{-1}$	5.49
	15	2.18	$4.09 \cdot 10^{-1}$	5.34	2.19	$4.81 \cdot 10^{-1}$	4.55
	31	2.37	$4.34 \cdot 10^{-1}$	5.47	2.37	$5.08 \cdot 10^{-1}$	4.67
10^{-4}	3	$6.66 \cdot 10^{-1}$	$3.15 \cdot 10^{-1}$	2.11	$3.33 \cdot 10^{-1}$	$1.41 \cdot 10^{-3}$	$2.35 \cdot 10^{2}$
	7	1.14	$1.50 \cdot 10^{-1}$	7.61	1.19	$3.10 \cdot 10^{-3}$	$3.84 \cdot 10^{2}$
	15	1.29	$8.44 \cdot 10^{-2}$	15.3	1.30	$8.37 \cdot 10^{-3}$	$1.56 \cdot 10^{2}$
	31	1.82	$8.08 \cdot 10^{-2}$	22.5	1.82	$1.74 \cdot 10^{-2}$	$1.05 \cdot 10^{2}$
10^{-6}	3	$6.67 \cdot 10^{-1}$	$3.15 \cdot 10^{-1}$	2.11	$3.33 \cdot 10^{-1}$	$1.42 \cdot 10^{-5}$	$2.34 \cdot 10^{4}$
	7	1.15	$1.50 \cdot 10^{-1}$	7.65	1.20	$3.11 \cdot 10^{-5}$	$3.85 \cdot 10^{4}$
	15	1.31	$7.79 \cdot 10^{-2}$	16.8	1.31	$8.41 \cdot 10^{-5}$	$1.55 \cdot 10^{4}$
	31	1.41	$5.72 \cdot 10^{-2}$	24.7	1.34	$1.79 \cdot 10^{-4}$	$7.50 \cdot 10^{3}$
0	3	$6.67 \cdot 10^{-1}$	$3.15 \cdot 10^{-1}$	2.11	$3.33 \cdot 10^{-1}$	$2.25 \cdot 10^{-17}$	$1.48 \cdot 10^{16}$
	7	1.15	$1.50 \cdot 10^{-1}$	7.65	1.20	$1.84 \cdot 10^{-18}$	$6.48 \cdot 10^{17}$
	15	1.31	$7.78 \cdot 10^{-2}$	16.8	1.31	$6.61 \cdot 10^{-18}$	$1.98 \cdot 10^{17}$
	31	1.41	$5.69 \cdot 10^{-2}$	24.8	1.34	$5.21 \cdot 10^{-17}$	$2.58 \cdot 10^{16}$

due to prohibitive time-step restrictions when explicitly treating the convection operator, which may be necessary in nonlinear advective problems. Dubiner [5] and Sherwin and Karniadakis [15] have already considered this by using expansion bases with $N(N+1)/2$ degrees of freedom rather than our N^2 expansion. The Dubiner basis is useful for time-dependent problems since the improved condition number makes the impact of the time-step limits less severe. Turning from boundary value problems to time-dependent Navier–Stokes flow, explicit time schemes then lead to extremely small time steps. This is typical for singular mappings of this kind.

DOMAIN DECOMPOSITION

We are now interested in applying the spectral method to more complex domains. We use the patching method (see [8]) where the domain is separated into square or triangular subdomains on which Gauss–Lobatto nodes are defined. The differential equation is solved at the interior nodes. At the interface we require continuity of the solution and its normal derivative. We consider the Poisson equation with the Dirichlet boundary condition

$$\Delta u = f \quad \text{in } \Omega,$$
$$u = g \quad \text{on } \partial\Omega.$$

At the interface Γ between two subdomains, information is exchanged until continuity is reached. In one direction Dirichlet information is transferred and in the other direction Neumann information. We use an interface relaxation as proposed in [8]; i.e., at the Dirichlet

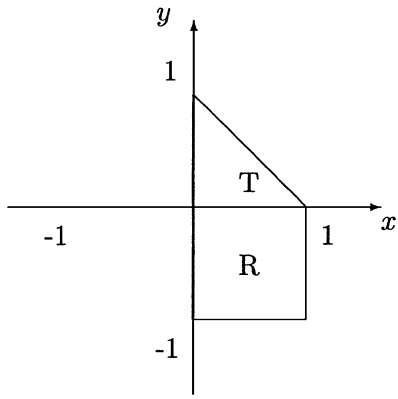

FIG. 17. Domain Ω.

side we hand over a weighted sum of subsolutions at the interface. We iterate until the error at the interface is smaller than 10^{-14}. Therefore we solve a sequence of Dirichlet–Neumann problems.

We begin with a domain composed of one patched triangle and square $\Omega = T \cup R$ where

$$T = \{(x, y) \mid 0 < x, y < 1, \quad \text{and} \quad x + y < 1\}$$

and

$$R = \{(x, y) \mid 0 < x < 1 \quad \text{and} \quad -1 < y < 0\}.$$

The interface is $\Gamma = (0, 1) \times \{0\}$ (Fig. 17). Initial conditions are $u_1^0 = u_2^0 \equiv 0$ on Ω and $u_1^1 = g$ on Γ. We then iterate

$$\begin{aligned} \Delta u_1^m &= f && \text{in } T, \\ u_1^m &= g && \text{on } \partial T \setminus \Gamma, \\ u_1^m &= \delta^{m-1} u_2^{m-1} + (1 - \delta^{m-1}) u_1^{m-1} && \text{on } \Gamma \end{aligned}$$

and

$$\begin{aligned} \Delta u_2^m &= f && \text{in } R, \\ u_2^m &= g && \text{on } \partial R \setminus \Gamma, \\ \frac{\partial}{\partial y} u_2^m &= \frac{\partial}{\partial y} u_1^m && \text{on } \Gamma. \end{aligned}$$

TABLE IX
Ω with (3)

N	Iterations	$E2_T$	$E2_R$
4	15	$1.36 \cdot 10^{-2}$	$1.28 \cdot 10^{-2}$
8	17	$2.42 \cdot 10^{-5}$	$1.64 \cdot 10^{-5}$
16	17	$8.01 \cdot 10^{-13}$	$4.24 \cdot 10^{-13}$
32	17	$1.00 \cdot 10^{-14}$	$1.60 \cdot 10^{-14}$

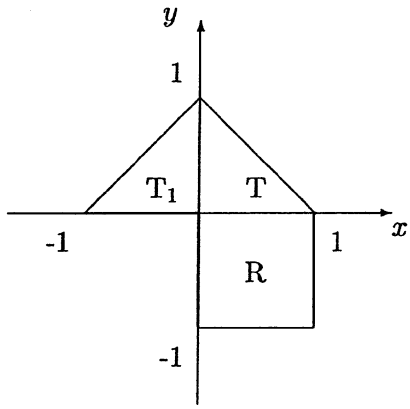

FIG. 18. Domain Ω_1.

Here δ^m denotes the relaxation parameter, which is chosen dynamically. This dynamical choice usually accelerates the convergence. The unique number $\delta^m = \delta$ minimizes $\|z_m(\delta) - z_{m-1}(\delta)\|_2^2$, where $z_m(\delta) = \delta u_2^m + (1-\delta)u_1^m \cdot \delta^m$ is calculated by

$$\delta^m = \frac{\left(e_1^m, e_1^m - e_2^m\right)}{\left\|e_1^m - e_2^m\right\|_2^2},$$

where $(.,.)$ denotes the discrete L_2 inner product and

$$e_i^m = u_i^m - u_i^{m-1} \quad \text{for } i = 1, 2$$

is the difference of two consecutive iterates on the two subdomains. Here δ^m should be in $(0, 1]$. We cannot use Example 1 since this function vanishes at the interface. Therefore no new information is exchanged, which makes an iterative method superfluous as it converges after the first step. Thus we introduce the following oscillating example:

$$u(x, y) = \sin(\pi x) \sin\left(\pi y + \frac{\pi}{4}\right). \tag{3}$$

Table IX shows the number of iterations and the discrete L_2 error on the defined square and triangle (Fig. 17). We reach the tolerance after relatively few steps. The convergence is

TABLE X
Ω_1 with (4)

N	Iterations	$E2_T$	$E2_R$	$E2_{T1}$
4	65	$1.04 \cdot 10^{-2}$	$9.62 \cdot 10^{-3}$	$1.25 \cdot 10^{-2}$
8	83	$6.98 \cdot 10^{-6}$	$5.67 \cdot 10^{-6}$	$2.68 \cdot 10^{-5}$
16	84	$6.85 \cdot 10^{-13}$	$4.01 \cdot 10^{-13}$	$1.02 \cdot 10^{-12}$
32	90	$2.31 \cdot 10^{-13}$	$8.30 \cdot 10^{-14}$	$9.90 \cdot 10^{-14}$

TABLE XI
Ω_1 with (4) and $\delta^m = 0.5$

N	Iterations	$E2_T$	$E2_R$	$E2_{T1}$
4	34	$4.89 \cdot 10^{-4}$	$2.82 \cdot 10^{-4}$	$1.51 \cdot 10^{-3}$
8	41	$9.29 \cdot 10^{-9}$	$2.30 \cdot 10^{-9}$	$6.82 \cdot 10^{-8}$
16	46	$1.78 \cdot 10^{-13}$	$3.70 \cdot 10^{-14}$	$1.83 \cdot 10^{-13}$
32	87	$9.68 \cdot 10^{-13}$	$8.40 \cdot 10^{-14}$	$1.01 \cdot 10^{-12}$

fairly slow because of the oscillatory behavior of the solution. The number of iterations is constant and independent of N. Machine accuracy is reached for $N = 16$.

The second domain Ω_1 to be studied consists of Ω and an additional triangle T_1 attached to the already existing triangle (Fig. 18). We begin with triangle T with interface boundaries $\Gamma_1 = (0, 1) \times \{0\}$ and $\Gamma_2 = \{0\} \times (0, 1)$ and $\Gamma = \Gamma_1 \cup \Gamma_2$. We then solve on R and T_1. This should be done on a parallel computer. The algorithm reads

$$\begin{aligned}
\Delta u_1^m &= f && \text{in } T,\\
u_1^m &= g && \text{on } \partial T \setminus \Gamma,\\
u_1^m &= \delta_1^{m-1} u_2^{m-1} + \left(1 - \delta_1^{m-1}\right) u_1^{m-1} && \text{on } \Gamma_1,\\
u_1^m &= \delta_2^{m-1} u_3^{m-1} + \left(1 - \delta_2^{m-1}\right) u_1^{m-1} && \text{on } \Gamma_2,
\end{aligned}$$

and

$$\begin{aligned}
\Delta u_2^m &= f && \text{in } R,\\
u_2^m &= g && \text{on } \partial R \setminus \Gamma_1,\\
\frac{\partial}{\partial y} u_2^m &= \frac{\partial}{\partial y} u_1^m && \text{on } \Gamma_1,
\end{aligned}$$

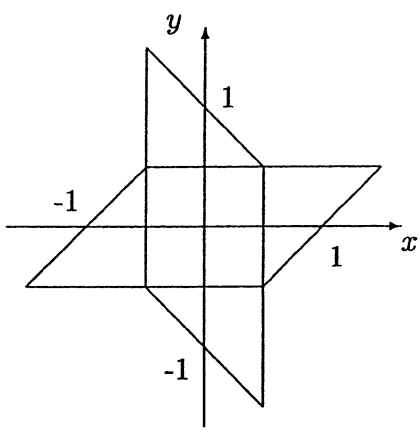

FIG. 19. "Wind wheel" Ω_2.

TABLE XII
Ω_2 with (5)

N	Iterations	$E2_R$	$E2_{T1}$	$E2_{T2}$	$E2_{T3}$	$E2_{T4}$
8	75	$9.00 \cdot 10^{-2}$	$5.86 \cdot 10^{-2}$	$7.55 \cdot 10^{-2}$	$4.89 \cdot 10^{-2}$	$7.13 \cdot 10^{-2}$
16	72	$4.33 \cdot 10^{-5}$	$2.08 \cdot 10^{-5}$	$4.70 \cdot 10^{-5}$	$1.79 \cdot 10^{-5}$	$4.63 \cdot 10^{-5}$
32	69	$3.83 \cdot 10^{-13}$	$5.10 \cdot 10^{-14}$	$1.06 \cdot 10^{-13}$	$6.60 \cdot 10^{-14}$	$8.60 \cdot 10^{-14}$

and

$$\begin{aligned} \Delta u_3^m &= f \qquad \text{in } T_1, \\ u_3^m &= g \qquad \text{on } \partial T_1 \setminus \Gamma_2, \\ \frac{\partial}{\partial x} u_3^m &= \frac{\partial}{\partial x} u_1^m \quad \text{on } \Gamma_2. \end{aligned}$$

Initial values are analogous to the last example. We apply this algorithm to the example

$$u(x, y) = \sin\left(\pi x + \frac{\pi}{4}\right) \sin\left(\pi y + \frac{\pi}{4}\right). \tag{4}$$

The results are listed in Table X. The number of iterations is increased significantly if a further triangle is added. Unfortunately, δ_i^m tends to leave the interval (0, 1]. Whenever this happens, the following approximation is worse than the one directly before. Nevertheless, the method finally converges. This dynamical choice of δ_i^m is not optimal. We have derived results for fixed $\delta_i^m = \frac{1}{2}$ in Table XI. The number of iterations is smaller and there are no backward steps any more.

Next, we study the domain $\Omega_2 = R \cup T_1 \cup T_2 \cup T_3 \cup T_4$ (T_i triangles), which is symmetric to the origin (Fig. 19). We consider the following example:

$$u(x, y) = \sin\left(3\pi x + \frac{\pi}{4}\right) \sin\left(3\pi y + \frac{\pi}{4}\right). \tag{5}$$

The algorithm is analogous to the last one and we first solve on the square and then on the triangles. The results in Table XII are fairly good for reasons of symmetry considering

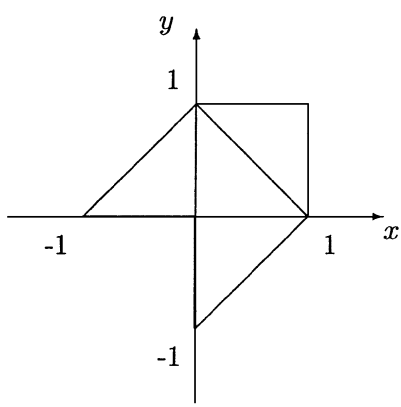

FIG. 20. Domain Ω_3.

TABLE XIII
Ω_3 with (5)

N	Iterations	$E2_T$	$E2_{T1}$	$E2_{T2}$	$E2_{T3}$
4	48	$4.51 \cdot 10^{-1}$	$4.07 \cdot 10^{-1}$	$1.48 \cdot 10^{-1}$	$2.17 \cdot 10^{-1}$
8	50	$9.80 \cdot 10^{-2}$	$7.64 \cdot 10^{-2}$	$8.42 \cdot 10^{-2}$	$6.21 \cdot 10^{-2}$
16	46	$3.73 \cdot 10^{-5}$	$2.23 \cdot 10^{-5}$	$4.90 \cdot 10^{-5}$	$4.14 \cdot 10^{-5}$
32	47	$2.23 \cdot 10^{-13}$	$2.88 \cdot 10^{-13}$	$2.60 \cdot 10^{-14}$	$7.00 \cdot 10^{-14}$

that we are now dealing with five subdomains. The number of iterations is constant and machine accuracy is reached for $N = 16$.

Finally, to demonstrate that this method delivers satisfactory results for other cases, we present domain $\Omega_3 = T \cup T_1 \cup T_2 \cup T_3$ (Fig. 20). As we now have only triangles, considerably fewer of iterations are required and the results are comparable to the wind wheel. Figure 21 shows the L_2 error on collocation nodes.

We have not considered time-dependent problems since our method requires very short time step limits because of the large condition number. Time-dependent problems cannot be treated efficiently using this method.

In summary, we can state that this spectral method is also effective for time-independent domain decomposition problems provided no cross points occur. Cross points are common corners of at least four spectral elements. If there are cross points, the number of iterations is quite large and so this method has to be improved. We can now also deal with partial differential equations on complex domains using spectral methods as long as those domains can be separated into rectangular and triangular elements. Delves *et al.* [4] found that domain decomposition is effective when there are lines of discontinuity in the middle of the domain or nasty corner singularities and they included maps of mixtures of squares and triangles in their global approach.

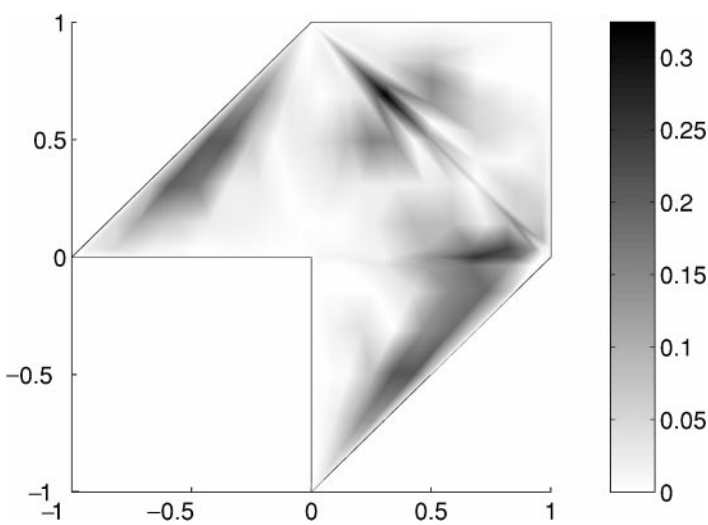

FIG. 21. Error distribution in Ω_3 with (4) and $N = 8$.

REFERENCES

1. J. P. Boyd, Polynomial series versus sinc expansions for functions with corner or endpoint singularities, *J. Comput. Phys.* **64**, 266 (1986).
2. J. P. Boyd, The asymptotic Chebyshev coefficients for functions with logarithmic endpoint singularities, *Appl. Math. Comput.* **29**, 49 (1989).
3. C. Canuto, M. Hussaini, A. Quarteroni, and T. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, Berlin, 1988).
4. L. M. Delves, A. McKerrell, and S. A. Peters, Performance of GEM2 on the ELLPACK problem population, *Int. J. Numer. Meth. Eng.* **23**, 229 (1986).
5. M. Dubiner, Spectral methods on triangles and other domains, *J. Sci. Comput.* **6**(4), 345 (1991).
6. H. Eisen and W. Heinrichs, A new method of stabilization for singular perturbation problems with spectral methods, *SIAM J. Numer. Anal.* **29**, 107 (1992).
7. D. Funaro, A fast solver for elliptic boundary-value problems in the square, *Comput. Meth. Appl. Eng*. **116**, 253 (1994).
8. D. Funaro, A. Quarteroni, and P. Zanolli, An iterative procedure with interface relaxation for domain decomposition methods, *SIAM J. Numer. Anal.* **25**, 1213 (1988).
9. M. Griebel, T. Dornseifer, and T. Neunhoeffer, *Numerische Simulation in der Strömungsmechanik* (Vieweg, Lehrbuch, 1995).
10. W. Heinrichs, Defect correction for convection-dominated flow, *SIAM J. Sci. Comput.* **17**, 1082 (1996).
11. W. Heinrichs, Spectral collocation on triangular elements, *J. Comput. Phys.* **145**, 743 (1998).
12. G. Karniadakis and S. J. Sherwin *Spectral/hp Element Methods for CFD. Numerical Mathematics and Computation* (Oxford Univ. Press, London, 1999).
13. C. Mavriplis and J. Van Rosendale, *Triangular Spectral Elements for Incompressible Fluid Flow*, ICASE Rep. 93-100 (1993).
14. A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations* (Springer-Verlag, Berlin, 1994).
15. S. Sherwin and G. Karniadakis, A triangular spectral element method; applications to the Navier–Stokes equations. *Comput. Methods Appl. Mech. Engrg.* 123, pp. 189–229 (1995).
16. S. Sherwin and G. Karniadakis, Triangular and tetrahedral spectral elements, in *ICOSAHOM'95: Proceedings of the Third International Conference on Spectral and High Order Methods*, 1996 Houston J. Math.