

AN EFFICIENT ITERATIVE SOLUTION METHOD FOR THE CHEBYSHEV COLLOCATION OF ADVECTION-DOMINATED TRANSPORT PROBLEMS*

A. PINELLI[†], W. COUZY[‡], M.O. DEVILLE[‡], AND C. BENOCCHI[†]

Abstract. A new Chebyshev collocation algorithm is proposed for the iterative solution of advection-diffusion problems. The main features of the method lie in the original way in which a finite-difference preconditioner is built and in the fact that the solution is collocated on a set of nodes matching the standard Gauss–Lobatto–Chebyshev set only in the case of pure diffusion problems. The key point of the algorithm is the capability of the preconditioner to represent the high-frequency modes when dealing with advection-dominated problems. The basic idea is developed for a one-dimensional case and is extended to two-dimensional problems. A series of numerical experiments is carried out to demonstrate the efficiency of the algorithm. The proposed algorithm can also be used in the context of the incompressible Navier–Stokes equations.

Key words. advection-diffusion, collocation, Chebyshev, preconditioning, finite difference, staggered grid

AMS subject classifications. 65N35, 76D99

1. Introduction. The need to solve implicit equations is a basic requirement for spectral algorithms. For both steady problems, whose solution is sought through a time marching computation, and for unsteady calculations, spectral methods are often feasible only if an implicit or a semi-implicit procedure is introduced. A typical example is the solution of the unsteady Navier–Stokes equations. Many numerical solution algorithms for this problem consist of an implicit treatment of the diffusive terms in combination with an explicit scheme for the convective terms [1]. This approach results in a severe restriction to the admissible time step when the flow dynamics are dominated by convection. Alternatively, the (linearized) convective terms can be treated implicitly as well, yielding a large admissible time step. As an example of the latter family of methods for the Navier–Stokes equations on a domain Ω , we consider the following semidiscretized version of the projection method proposed by Shen [2].

$$\begin{aligned} (1) \quad & \frac{1}{\kappa} (\hat{u}^{n+1} - u^n) - \nu \Delta \hat{u}^{n+1} + (u^n \cdot \nabla) \hat{u}^{n+1} = b^{n+1}, \\ (2) \quad & \hat{u}_{\partial\Omega}^{n+1} = 0, \\ (3) \quad & \frac{1}{\kappa} (u^{n+1} - \hat{u}^{n+1}) + \nabla \phi^{n+1} = 0, \\ (4) \quad & \nabla \cdot u^{n+1} = 0, \\ (5) \quad & u^{n+1} \cdot \vec{n} = 0, \end{aligned}$$

where u is the velocity, ϕ is an approximation of the pressure, κ is the time step, \vec{n} is the normal vector to the boundary $\partial\Omega$, b is a force vector, and the superscript n represents the time iteration index. Equation (1) is a series of scalar advection-diffusion equations, each one having as unknown a velocity component.

Here, we will focus on a particular algorithmical aspect related to equation (1), when the spatial operators are discretized by the Chebyshev collocation technique [1]. To this end we

*Received by the editors August 13, 1993; accepted for publication (in revised form) March 2, 1995. This text presents research results of the Belgian Incentive Program “Information Technology–Computer Science of the Future,” initiated by the SPSS (Services du Premier Ministre, Programmation de la Politique Scientifique). All correspondence should be sent to M. Deville, Ecole Polytechnique Fédérale de Lausanne, DGM-IMHEF, 1015 Ecublens, Switzerland.

[†]Von Karman Institute for Fluid Dynamics, Chaussée de Waterloo 72, 1640 Rhode Saint Genèse, Belgium.

[‡]Université Catholique de Louvain, CESAME, Av. G. Lemaitre 4-6, 1348 Louvain-la-Neuve, Belgium. Present address: Ecole Polytechnique Fédérale de Lausanne, DGM-IMHEF, 1015 Ecublens, Switzerland (deville@imhfhfp28.epfl.ch).

will consider the following one-dimensional equation, meant to model the linearized equation (1):

$$(6) \quad -\epsilon \frac{d^2 u}{dx^2} + p(x) \frac{du}{dx} = b(x) \quad \text{on } \Omega = (-1, 1),$$

$$(7) \quad u = 0 \quad \text{on } \partial\Omega.$$

The two-dimensional formulation of (6)–(7) reads

$$(8) \quad -\epsilon \Delta u + p(x, y) \frac{\partial u}{\partial x} + q(x, y) \frac{\partial u}{\partial y} = b(x, y) \quad \text{on } \Omega = (-1, 1)^2,$$

$$(9) \quad u = 0 \quad \text{on } \partial\Omega,$$

where the given functions $p(x)$, $p(x, y)$, and $q(x, y)$ can be considered as representatives of the velocity field at the previous time step.

The collocation procedure applied to (6)–(7) or (8)–(9) yields the linear system

$$(10) \quad \mathbf{L}U = F.$$

Our goal is the design of an efficient iterative procedure to solve problem (10), such that the convergence does not depend on the ratio between local advection and the positive diffusion coefficient ϵ . An effective solution technique does not only depend on the choice of the iterative scheme but also on the efficiency of the preconditioner. We propose a particular preconditioned collocation scheme that relies on the scheme introduced by Funaro [3] for the Legendre collocation method. This method is based on the collocation of the equations on a particular set of points (termed as “staggered grid”) which is determined as a deviation from the original Gauss–Lobatto–Chebyshev (GLC) quadrature nodes in function of the local ratio between advection and diffusion. In this paper, we will extend the Funaro scheme to the Chebyshev collocation method, by introducing an original definition of the staggered grid that does not rely on the Sturm–Liouville problem associated with Legendre polynomials. We will show that such a grid exists and is unique. Moreover, the numerical procedure to compute the staggered grid is simplified, yielding important savings of CPU time.

2. One-dimensional model. In this paragraph we will give some heuristic considerations to highlight the difficulties that arise when one tries to precondition discrete advection–diffusion systems dominated by the advective terms. The same reasoning will introduce the basic ideas the proposed scheme relies on.

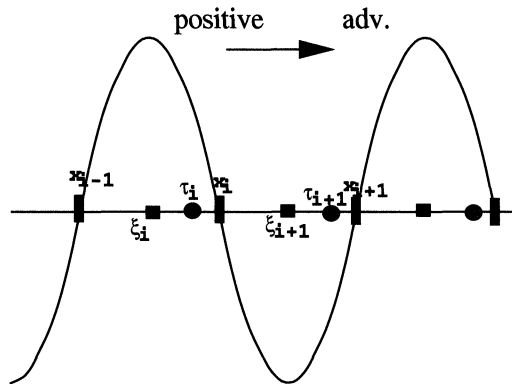
We start considering the one-dimensional case (6)–(7), with the given function $p(x)$ strictly positive on all the domain $\Omega = (-1, +1)$. The Chebyshev polynomial of order N is denoted by $T_N(x)$. The GLC points $\{x_i\}$ are defined as the zeros of $(x^2 - 1)T'_N(x)$ and are therefore given by

$$(11) \quad x_i = \cos\left(\frac{\pi i}{N}\right), \quad i = 0, \dots, N.$$

For future use, we introduce as well the set of corresponding midpoints $\{\xi_i\}$, defined as the zeros of $T'_N(x)$:

$$(12) \quad \xi_i = \cos\left(\pi \frac{2i+1}{2N}\right), \quad i = 0, \dots, N-1.$$

Next we consider a polynomial $z(x) = T'_N(x)$ (sketched in Figure 1) vanishing in all internal GLC points, according to (11). Such a polynomial represents a high-frequency mode, typical of a spectral approximation, that standard finite-difference operators fail to represent for advection-dominated problems. To explain this, we remark that the centered finite-

FIG. 1. The polynomial $z(x)$.

difference approximation of the second-order derivative in x_i (which is close to zero) is given by $\frac{h_{i+1}z(x_{i-1}) - (h_i + h_{i+1})z(x_i) + h_i z(x_{i+1}))}{0.5 h_i h_{i+1} (h_i + h_{i+1})} = 0$, with $h_i = x_i - x_{i-1}$. An approximation of the first-order derivative (which attains its maximum value in x_i) will also yield zero, since $\frac{z(x_{i+1}) - z(x_{i-1}))}{h_i + h_{i+1}} = 0$. This observation explains the degrading behavior of classical finite-difference schemes constructed on the original GLC nodes for advection-dominated problems. In such a case, it is essential to provide a better approximation for the first-order derivative. To this end, we notice that the first-order derivative in the midpoint ξ_i is much more accurately approximated ($z'(\xi_i) \approx 0$) using the values of $z(x_i)$ and $z(x_{i-1})$. Such a heuristic approach is somehow confirmed by the theorem of Funaro [4] for first-order operators. For advection-diffusion problems, we are therefore induced to believe that the finite-difference preconditioner has to be evaluated on a special “staggered grid,” although constructed using data at the GLC nodes. To set up this staggered grid, first we apply the operator (6) to the polynomial $z(x)$ that mimics the behavior of a high-frequency eigenfunction, and then we compute the zeros $\{\tau_i\}$ of the residual polynomial $f(x)$:

$$(13) \quad f(x) = -\epsilon z''(x) + p(x)z'(x).$$

For a purely second-order operator ($p(x) = 0$), the points $\{\tau_i\}$ coincide with the original set of GLC nodes, while for a first-order operator ($\epsilon = 0$) they drop on the GLC midpoints $\{\xi_i\}$. For a general advection-diffusion operator the following estimates hold:

$$(14) \quad \begin{aligned} \xi_i &< \tau_i < x_i \text{ if } p(x) > 0, \\ x_i &< \tau_i < \xi_{i+1} \text{ if } p(x) < 0. \end{aligned}$$

To implement the above theory, a polynomial $z(x)$ has to be found that mimics a high-frequency mode (i.e., $z(x_i) = 0$ and $z'(x_i)$ close to a local maximum or minimum at the interior collocation nodes x_i [3]). Because our target is the determination of a “staggered grid,” which has the same number of nodes as the original Gauss–Lobatto grid, we require that the corresponding function $f(x)$ (see (13)) vanishes once and only once in each interval defined by $[\xi_i, x_i]$ when $p > 0$ and $[x_i, \xi_{i+1}]$ when $p < 0$. For the Legendre case, the procedure of determining a function $z(x)$ that respects all the aforementioned requirements is simplified by the possibility of reducing the complexity of the formulation by using the associated Sturm–Liouville problem [3]. In the Chebyshev case on the contrary, no simplification can be taken on the same side. Moreover, in analogy with the Legendre case, if we let the first-order derivative of the N th Chebyshev mode play the role of the function $z(x)$, it can be shown that function $f(x)$ does not necessarily present a zero in the required interval $[\xi_i, x_i]$. For these reasons we

prefer to weaken the condition of $z(x)$ vanishing in the set $\{x_i\}$. Therefore, like Funaro [3] for the Legendre case, instead of requiring that

$$(15) \quad z(x_i) = 0, \quad z''(x_i) \approx 0, \quad \text{and} \quad \exists! \tau_i \in [\xi_i, x_i] : f(\tau_i) = 0,$$

we introduce a new strategy to determine the staggered grid by asking that

$$(16) \quad z(x_i) \approx 0, \quad z''(x_i) = 0, \quad \text{and} \quad \exists! \tau_i \in [\xi_i, x_i] : f(\tau_i) = 0.$$

In this respect, a good candidate for $z(x)$ is

$$(17) \quad z(x) = \int T_N(x) dx.$$

It is clear that $z''(x_i) = 0$. Although $z(x)$ is not exactly zero in the set of $\{x_i\}$, we will prove that it differs from zero only by terms of order $1/N^2$ and we will also show the existence and uniqueness of a zero for the function $f(x)$ in the required interval $[\xi_i, x_i]$.

First we prove that $z(x)$ is “almost” zero in the set of $\{x_i\}$.

THEOREM 2.1. *For all $x_i (i = 1, \dots, N-1)$ the polynomial $z(x)$ defined in (17) satisfies*

$$|z(x_i)| < \frac{1}{N^2 - 1}.$$

Proof. Because $T_N(x) = \frac{1}{2} \frac{1}{N+1} T'_{N+1}(x) - \frac{1}{2} \frac{1}{N-1} T'_{N-1}(x)$, one has

$$\begin{aligned} z(x_i) &= \int T_N(x) dx|_{x_i} \\ &= \frac{1}{2} \frac{1}{N+1} T_{N+1}(x_i) - \frac{1}{2} \frac{1}{N-1} T_{N-1}(x_i) \\ &= \frac{1}{2} \frac{1}{N+1} \cos\left(\frac{(N+1)\pi i}{N}\right) - \frac{1}{2} \frac{1}{N-1} \cos\left(\frac{(N-1)\pi i}{N}\right) \\ &= \frac{1}{2} \frac{1}{N+1} \left\{ \cos(\pi i) \cos\left(\frac{\pi i}{N}\right) - \sin(\pi i) \sin\left(\frac{\pi i}{N}\right) \right\} \\ &\quad - \frac{1}{2} \frac{1}{N-1} \left\{ \cos(\pi i) \cos\left(-\frac{\pi i}{N}\right) - \sin(\pi i) \sin\left(-\frac{\pi i}{N}\right) \right\} \\ (18) \quad &= \frac{1}{2} \frac{(-1)^i x_i}{N+1} - \frac{1}{2} \frac{(-1)^i x_i}{N-1} = \frac{(-1)^{i+1} x_i}{N^2 - 1}. \quad \square \end{aligned}$$

Next, we show that the zeros of $f(x)$ can be used to build the finite-difference grid. More precisely, we have the following theorem.

THEOREM 2.2. *For all $x_i (i = 1, \dots, N)$, $\epsilon > 0$, $p(x)$ piecewise constant on $[\xi_i, \xi_{i+1}]$, the following properties hold.*

If $p(x) > 0$ then there exists a unique $\tau_i \in [\xi_i, x_i]$ such that $f(\tau_i) = 0$.

If $p(x) < 0$ then there exists a unique $\tau_i \in [x_i, \xi_{i+1}]$ such that $f(\tau_i) = 0$.

Proof. First, we assume that $p(x)$ is a positive constant in $[\xi_i, \xi_{i+1}]$. In this case, $f(x)$ is a polynomial of degree N . If we prove that there is at least one zero in each interval $[\xi_i, x_i]$, the total number of zeros is N and hence every zero is unique since N corresponds to the degree of the polynomial $f(x)$. This leaves us to show that $f(\xi_i) \cdot f(x_i) < 0$. We have

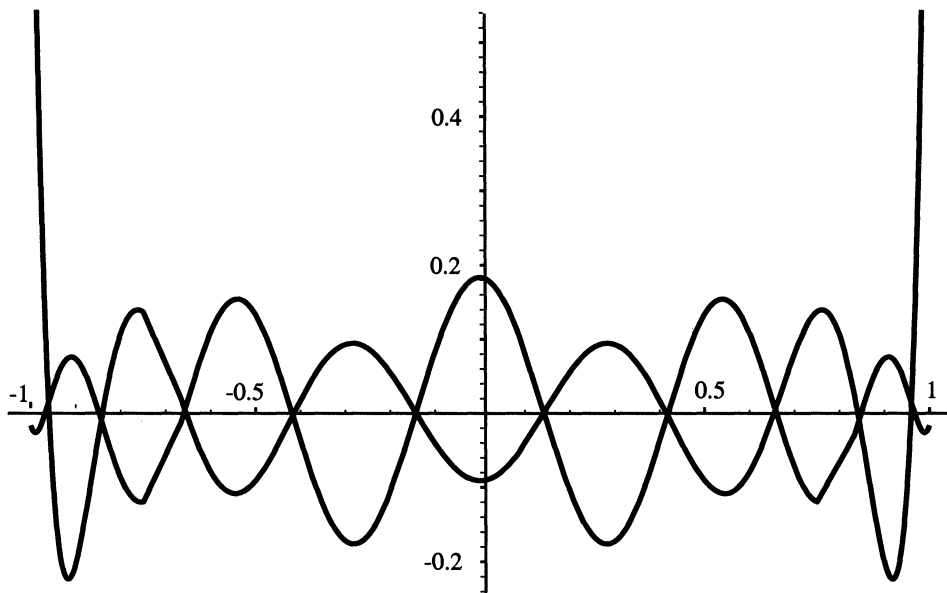


FIG. 2. $z(x)$ and $T'_N(x)/N^2$, for $N = 11$. $z(0) < 0$.

$f(\xi_i) = -\epsilon T'_N(\xi_i)$ and $f(x_i) = p(x_i)T_N(x_i)$. Because $p(x_i) > 0$, $\epsilon > 0$, it is enough to show that $T'_N(\xi_i) \cdot T_N(x_i) > 0$. This follows easily by noticing that

$$(19) \quad \begin{aligned} T'_N(\xi_i) &= \frac{N}{\sqrt{1-\xi_i^2}} \sin(N \arccos(\xi_i)) = \begin{cases} < 0 & i \text{ odd,} \\ > 0 & i \text{ even,} \end{cases} \\ T_N(x_i) &= \cos(N \arccos(x_i)) = \begin{cases} < 0 & i \text{ odd,} \\ > 0 & i \text{ even.} \end{cases} \end{aligned}$$

In the same way, we demonstrate that $f(x)$ has exactly one zero in each interval $[x_i, \xi_{i+1}]$ if $p < 0$. To prove the theorem for $p(x)$ piecewise constant, we construct the family of polynomials $f_i(x) = -\epsilon z''(x) + p(x_i)z'(x)$. Each polynomial f_i has exactly one zero in the interval $[\xi_i, x_i]$ (if $p(x_i) > 0$) or in the interval $[x_i, \xi_{i+1}]$ (if $p(x_i) < 0$). Because $f(x)$ has the property that its restriction on $[\xi_i, \xi_{i+1}]$ coincides with the one of $f_i(x)$, it has exactly one zero in each desired interval. \square

If $p(x)$ has a zero in an interval $[\xi_i, \xi_{i+1}]$, we map the finite-difference point on the Chebyshev node x_i .

In Figure 2, we display the function $z(x)$ and $T'_N(x)$. The latter function has $\{x_1, \dots, x_{N-1}\}$ as zeros and represents a high-frequency mode. We see that the zeros of the two polynomials practically coincide.

Based on the arguments presented in this section, we propose to use as a preconditioner the matrix associated with the centered finite-difference approximation to the problem (6)–(7) at the gridpoints $\{\tau_i\}$.

3. Two-dimensional formulation. Here, we consider the extension of the preconditioner to the two-dimensional problem (8)–(9). The GLC nodes are denoted by (x_i, y_j) and the midpoints by (ξ_i, η_j) ; cf. Figure 3. We propose to use the following function $z(x, y)$ that

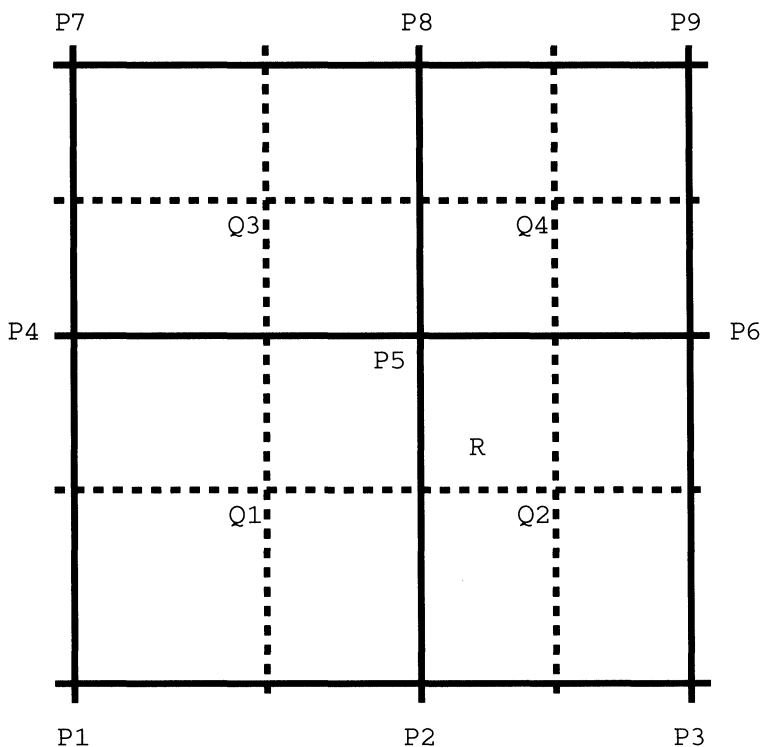


FIG. 3. The position of the staggered point $R = (\tau_{ij}, v_{ij})$ in the GLC grid (straight lines) and the Gauss-Chebyshev grid (also called midpoints; dashed lines). $P1 = (x_{i-1}, y_{j-1})$, $P5 = (x_i, y_j)$, $Q1 = (\xi_i, \eta_j)$, $Q4 = (\xi_{i+1}, \eta_{j+1})$, etc.

simulates a high-frequency mode in two dimensions:

$$(20) \quad z(x, y) = \iint T_N(x)T_N(y)dx dy.$$

This function can be regarded as the tensor product of two one-dimensional functions $z(x)$ and $z(y)$. We find for all $x, y \in \Omega$ and for all $i, j = 0, \dots, N$ that

$$(21) \quad z(x_i, y) \approx 0, \quad z_{xx}(x_i, y) = 0, \quad z(x, y_j) \approx 0, \quad \text{and} \quad z_{yy}(x, y_j) = 0.$$

Substitution of $z(x, y)$ in the two-dimensional analogue of (13) gives

$$(22) \quad f(x, y) = -\epsilon \Delta z(x, y) + p(x, y)z_x(x, y) + q(x, y)z_y(x, y).$$

The zeros $\{(\tau_{ij}, v_{ij})\}$ of the function $f(x, y)$ build the finite-difference grid for the preconditioner. In the following, we will assume p and q to be piecewise constant functions. Furthermore, we will focus on a rectangular domain Ω_{ij} ,

$$(23) \quad \Omega_{ij} = [\xi_i, \xi_{i+1}] \times [\eta_j, \eta_{j+1}],$$

on which $p(x, y) = p_{ij} > 0$ and $q(x, y) = q_{ij} > 0$. The zeros of $f(x, y)$ are given by the following relation:

$$(24) \quad f(x, y) = 0 \iff \{-\epsilon T'_N(x) + p_{ij}T_N(x)\} \int T_N(y)dy + \{-\epsilon T'_N(y) + q_{ij}T_N(y)\} \int T_N(x)dx = 0.$$

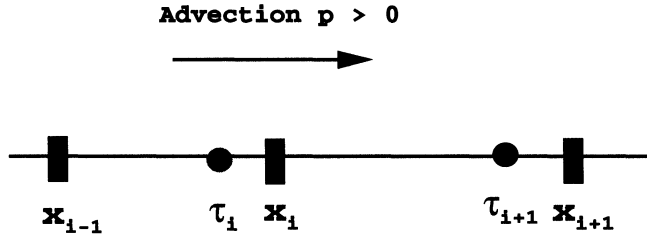


FIG. 4. The position of three consecutive GLC nodes x_{i-1} , x_i , x_{i+1} on which the second-order Lagrange polynomial is constructed. This polynomial is evaluated in τ_i .

From Theorem 2.1, we find

$$(25) \quad \exists! \tau_{ij} \in [\xi_i, x_i] : -\epsilon T'_N(\tau_{ij}) + p_{ij} T_N(\tau_{ij}) = 0,$$

$$(26) \quad \exists! v_{ij} \in [\eta_i, y_i] : -\epsilon T'_N(v_{ij}) + q_{ij} T_N(v_{ij}) = 0.$$

Hence, $\{(\tau_{ij}, v_{ij})\}$ are zeros of $f(x, y)$. Note that $f(x, y)$ is a polynomial of order $(N + 1)^2$, which implies that not all the zeros are found by solving (25)–(26). However, this is not a serious problem, since these additional zeros are not used to build up the finite-difference grid. If $p(x, y)$ assumes the value zero at one point at least in the rectangle Ω_{ij} , we will take x_i as the x -coordinate of the finite-difference node (analogously for $q(x, y)$).

4. Algorithmic aspects. We start focusing upon the one-dimensional model equation (6)–(7). The first step consists of determining the zeros of polynomial $f(x)$ (13). The computational cost to solve such a problem should be kept as low as possible. This issue is especially important when the preconditioner is used for nonlinear problems, like the Burgers or Navier–Stokes equations, in which the finite-difference grid has to be computed at each iteration. Therefore, we propose not to solve (13) by an expensive Newton method, as has been proposed by Funaro in [3]. Instead, we replace $f(x)$ by a piecewise parabolic function $d_i(x)$ on each $[\xi_i, x_i]$ (when $p > 0$) or on each $[x_i, \xi_{i+1}]$ (when $p < 0$). The zeros of these parabolas can be found by a simple and cheap formula. In the case of $p > 0$, we define the parabola $d_i(x)$ on the interval $[\xi_i, x_i]$ by the following three conditions.

$$(27) \quad d_i(\xi_i) = f(\xi_i), \quad d_i(x_i) = f(x_i), \quad \text{and} \quad (d_i)_x(\xi_i) = f_x(\xi_i).$$

In this way, the second-order polynomial d_i has exactly one zero in $[\xi_i, x_i]$. Because our preconditioner is designed for advection-dominated problems, the first-order derivative is matched in ξ_i and not in x_i (27), yielding a better approximation for the zero which is, in this case, close to ξ_i .

Once the zeros $\{\tau_i\}$ (i.e., the “staggered grid”) are found, for any i we approximate equation (6) with second-order Lagrange polynomials based on the data from three successive Chebyshev nodes x_{i-1} , x_i , x_{i+1} and collocated on the corresponding staggered point τ_i (see Figure 4). The general algorithm for the one-dimensional equation (6) with homogeneous Dirichlet boundary conditions (7) reads as follows.

- Select a guess solution U^0 .
- Compute the zeros $\{\tau_i\}$ of the polynomial $f(x)$ defined in (13).
- Construct the finite-difference preconditioner \mathbf{H} as

$$(28) \quad \mathbf{H} = \begin{bmatrix} b & c & 0 & & & \\ a & \ddots & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \ddots & \\ & 0 & a_i & b_i & c_i & 0 \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & \\ & & & & \ddots & \\ & & & & & \ddots & \\ & & & & & & \ddots & \\ & & & & & & & \ddots & \\ & & & & & & & & \ddots & \end{bmatrix}$$

with the following entries:

$$(29) \quad \begin{aligned} a_i &= p(\tau_i)l'_{i-1}(\tau_i) - \epsilon l''_{i-1}(\tau_i), \\ b_i &= p(\tau_i)l'_i(\tau_i) - \epsilon l''_i(\tau_i), \\ c_i &= p(\tau_i)l'_{i+1}(\tau_i) - \epsilon l''_{i+1}(\tau_i), \end{aligned}$$

as

$$(30) \quad l_i(x) = \frac{(x - x_{i-1})}{(x_i - x_{i-1})} \cdot \frac{(x - x_{i+1})}{(x_i - x_{i+1})}.$$

• Instead of $\mathbf{L}U = F$ solve

$$(31) \quad \mathbf{H}^{-1}\mathbf{L}U = \mathbf{H}^{-1}F$$

by an iterative algorithm. We remind that the system $\mathbf{L}U = F$ (see (10)) is the algebraic formulation of the spectral collocation method at the Chebyshev nodes $\{x_i\}$. Let us consider a simple preconditioned Richardson algorithm [1].

$$(32) \quad \begin{cases} \mathbf{H}\delta U = R\tilde{H}S, \\ U^{n+1} = U^n + \delta U, \end{cases}$$

with

$$(33) \quad RHS = \omega(F - \mathbf{L}U^n) = \mathbf{W}^{-1}R\tilde{H}S,$$

\mathbf{W} is a mapping operator that interpolates the residual computed on the collocation grid $\{x_i\}$ to the staggered grid $\{\xi_i\}$ and ω a relaxation factor. At each step a finite-difference problem is solved by inverting a tridiagonal matrix with a right-hand side that is the “spectral” residual at the previous step interpolated on the staggered grid.

• Iterate until convergence.

The previously described procedure has been extended to the two-dimensional advection-diffusion equation (8)–(9). Here, the staggered grid is given by the set of points $\{(\tau_{ij}, \eta_{ij})\}$ that are always determined as a deviation from the original Gauss-Lobatto grid in terms of the ratio between local advection $(p(x, y), q(x, y))$, and the diffusion coefficient ϵ . In particular, according to relationships (23) and (24), we approximate $f_1(x) := -\epsilon T'_N(x) + p_{ij}T_N(x)$ and $f_2(y) := -\epsilon T'_N(y) + q_{ij}T_N(y)$ locally by two one-dimensional parabolas. The zeros of these parabolas can be found following the procedure previously described for the one-dimensional case.

TABLE 1
Eigenvalues of the matrix $\mathbf{H}^{-1}\mathbf{L}$ for different values of ϵ .

Testcase	$\max \Re(\lambda)$	$\min \Re(\lambda)$	$\max \Im(\lambda)$
$\epsilon = 10^{-2}$ 21 nodes	1.09	0.65	0.56
$\epsilon = 10^{-4}$ 21 nodes	2.31	0.65	0.15
$\epsilon = 10^{-5}$ 21 nodes	2.31	0.32	0.09
$\epsilon = 10^{-5}$ 41 nodes	2.31	0.32	0.10

The mapping operator for the two-dimensional case \mathbf{W} is now chosen to be the local Taylor series expansion of the residual $r(x, y)$ of the current iteration:

$$(34) \quad \begin{aligned} r(\tau_{ij}, \eta_{ij}) \approx & r(x_i, y_j) + (\tau_{ij} - x_i) \frac{\partial r}{\partial x} + (\eta_{ij} - y_j) \frac{\partial r}{\partial y} + \frac{1}{2}(\tau_{ij} - x_i)^2 \frac{\partial^2 r}{\partial x^2} \\ & + \frac{1}{2}(\eta_{ij} - y_j)^2 \frac{\partial^2 r}{\partial y^2} + (\tau_{ij} - x_i)(\eta_{ij} - y_j) \left(\frac{\partial^2 r}{\partial x \partial y} \right), \end{aligned}$$

where the derivatives of the residual $r(x_i, y_j)$ are “spectrally” computed at the GLC nodes.

The finite-difference preconditioner \mathbf{H} is now based on a nine-points stencil and is always constructed using second-order Lagrange polynomials. The nine entries of the stencil corresponding to the point (x_i, y_j) are denoted by $h_{i,j}^{(1)}, \dots, h_{i,j}^{(9)}$ and are determined by imposing

$$(35) \quad \begin{aligned} & h_{i,j}^{(1)} u_{i-1,j-1} + h_{i,j}^{(2)} u_{i,j-1} + h_{i,j}^{(3)} u_{i+1,j-1} \\ & + h_{i,j}^{(4)} u_{i-1,j} + h_{i,j}^{(5)} u_{i,j} + h_{i,j}^{(6)} u_{i+1,j} \\ & + h_{i,j}^{(7)} u_{i-1,j+1} + h_{i,j}^{(8)} u_{i,j+1} + h_{i,j}^{(9)} u_{i+1,j+1} \\ & = -\epsilon \Delta u + p(x, y) \frac{\partial u}{\partial x} + q(x, y) \frac{\partial u}{\partial y} \quad \text{at } (\tau_{ij}, \eta_{ij}), \end{aligned}$$

meaning that equation (8) should be exactly satisfied for each Lagrange polynomial $u(x, y)$ of degree two in each variable. Such a procedure leads to a preconditioner with a bandwidth of $2N + 5$.

The iterative algorithm that we used for the two-dimensional case is an *orthomin*{5} (meaning that the maximum dimension of the Krylov space kept in memory is 5; in other words the algorithm is restarted every 5 iterations) [5] preconditioned with the above-described matrix \mathbf{H} .

5. Numerical experiments. To test the effectiveness of the developed preconditioner, we computed the eigenvalues of matrix $\mathbf{H}^{-1}\mathbf{L}$ for the one-dimensional case. The results are reported in Table 1 and refer to different values of the diffusion coefficient ϵ with the advection function $p(x)$ frozen at unity ($p(x) = 1$). The features of the preconditioned eigenvalues spectrum are summarized by giving the maximum and the minimum real parts together with the imaginary part. For the most severe test case at $\epsilon = 10^{-5}$ we also give the spectrum with two different sets of nodes (i.e., respectively, $N = 21$ and $N = 41$). This last result is intended to show the independence of the eigenvalues spectrum of the number of nodes. We recall here that in the nonpreconditioned case the conditioning number scales like $\sim N^2$ for the advective part and like $\sim N^4$ for the diffusive one [6]. In Figure 5, the eigenvalues spectrum is also shown for the cases $\epsilon = 10^{-2}$ and $\epsilon = 10^{-5}$.

To check the robustness of the method for the two-dimensional case, several tests were performed using different advective functions $p(x, y)$ and $q(x, y)$. In Table 2, the efficiency of the preconditioner in terms of number of iterations required to achieve machine precision (i.e.,

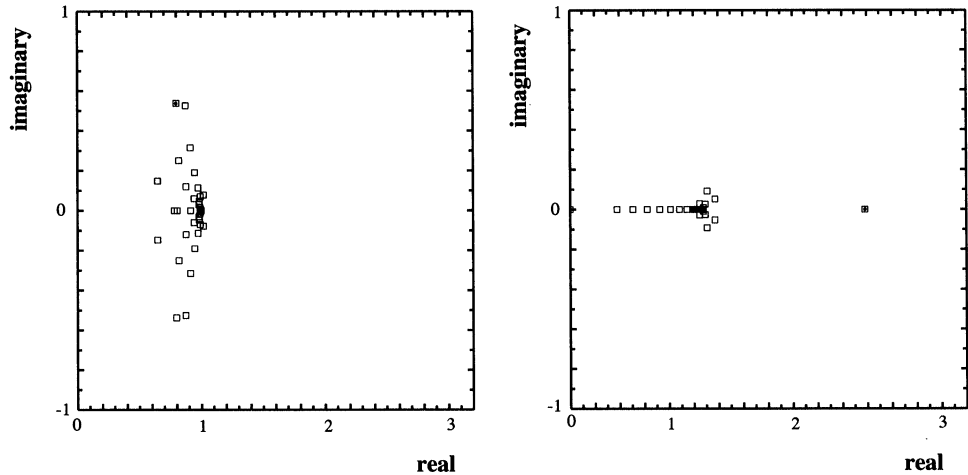


FIG. 5. Preconditioned eigenvalues spectrum, $\epsilon = 10^{-2}$ and $\epsilon = 10^{-5}$, respectively.

TABLE 2
Number of iterations to reduce the norm of the residual to 10^{-15} .

Testcase	Test 1	Test 2	Test 3	Test 4
Number of its.	65	45	56	82

L_∞ norm of the residuals at 10^{-15} for a double precision computation) is given at $\epsilon = 10^{-3}$ for, respectively,

(36)

Test 1.

Test 2.

Test 3.

Test 4.

$p(x, y) = 1,$

$p(x, y) = \sin(\pi x),$

$p(x, y) = 3x - y - 1,$

$p(x, y) = \sin(\pi x) \sin(\pi y),$

$q(x, y) = 1,$

$q(x, y) = 3x - y - 1,$

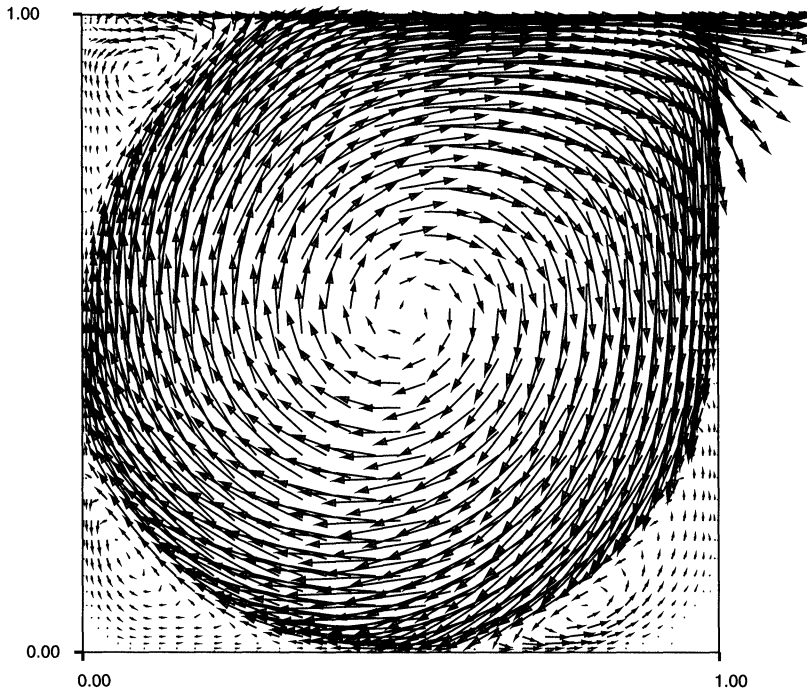
$q(x, y) = 3x^2 - y,$

$q(x, y) = \sin(\pi x) \sin(\pi y).$

As shown, the behavior is always satisfactory and the required number of iterations never exceeds by more than 30% the ones required for the constant advection case.

To test the developed preconditioner in a definite way, we considered equation (1) to be solved for the x -component of the velocity. The role of the advective functions $p(x, y)$ and $q(x, y)$ was played this time by an intermediate solution $u(x, y), v(x, y)$ of a two-dimensional regularized driven cavity at a Reynolds number of 10000 (see Figure 6). The whole procedure converged to machine accuracy (10^{-14}) in less than 100 iterations, against the 27 iterations that are required when only the diffusive terms are treated implicitly (i.e., with a classical treatment of the diffusive terms with Crank–Nicolson and of the advective ones with Adams–Bashforth). Of course, a huge increase of the maximum allowed time step is expected from the fully implicit treatment and represents the payoff of the present method.

6. Conclusions. The basic target of the present work was the design of an efficient algorithm for the iterative solution of the advection-diffusion equations. To this end, a finite-difference preconditioner has been introduced on a “staggered” grid. The staggered points are given as the zeros of a polynomial that depends on the ratio between the local advection and the diffusion. Existence and uniqueness of these zeros is assured and the capacity of the preconditioner to represent high-frequency modes is shown. A cheap procedure to compute the staggered points has also been introduced. This is especially important when dealing with linearized problems, where the grid has to be constructed for every time step.

FIG. 6. *The chosen flow field.*

The efficiency of the implicit method is confirmed by numerical results for advection-dominated problems. Moreover, the distribution of the eigenvalues of the preconditioned operator seems almost independent of the value of the diffusion coefficient. The algorithm has also been applied to the projection method for the two-dimensional unsteady Navier-Stokes equations. Even for large Reynolds numbers, a very moderate number of iterations is needed to converge to machine accuracy.

Finally, we remark that we have been concerned with the development of an efficient preconditioning scheme and that no attempt has been made to cure problems related to numerical instabilities due to large Peclet numbers. Hence, we do not pretend that the presented method produces accurate results for the well-known boundary-layer problems for the values of ϵ and N that have been used to solve the analytical problem (36). Nevertheless, the present technique produces a condition number for the preconditioned operator that is almost independent of the number of nodes N . One way to take advantage of this feature is to increase N to be able to accurately capture boundary layers, without achieving prohibitive operation counts.

REFERENCES

- [1] C. CANUTO, M. HUSSAINI, A. QUARTERONI, AND T. ZANG, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York, 1988.
- [2] J. SHEN, *On error estimates of projection methods for Navier-Stokes equation: First-order schemes*, SIAM J. Numer. Anal., 29 (1992), pp. 57-77.
- [3] D. FUNARO, *A new scheme for the approximation of advection-diffusion equations by collocation*, SIAM J. Numer. Anal., 30 (1993), pp. 1664-1676.
- [4] D. FUNARO AND E. ROTHMAN, *Preconditioning matrices for the pseudo-spectral approximation of first-order operators*, in *Proceedings Finite Element Analysis in Fluids*, T. J. Chung and G. R. Karr, eds., University of Alabama-Huntsville Press, Huntsville, AL, 1989.
- [5] S. EISENSTAT, H. ELMAN, AND M. SHULTZ, *Variational iterative methods for non-symmetric systems of linear equations*, SIAM J. Numer. Anal., 20 (1983), p. 345.
- [6] T. ZANG, C. STREETT, AND M. HUSSAINI, *Spectral Methods for Computational Fluid Dynamics*, Institute for Computer Applications in Science and Engineering, Report 89-13, Hampton, VA, 1989.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.