DIFFERENCE METHODS FOR SOLVING CONVECTION-DIFFUSION EQUATIONS

G. BIRKHOFF,1 E. C. GARTLAND JR,2 and R. E. LYNCH3

1Mathematics Department, Harvard University, Cambridge, MA 02138, U.S.A.
2Mathematics Department, Kent State University, Kent, OH 44242, U.S.A.
3Computer Science and Mathematics Departments, Purdue University, West Lafayette, IN 47907, U.S.A.

Abstract—Difference methods for solving the convection-diffusion equation are discussed. The superiority of Allen's approximation over central or upwind differences for one-dimensional problems is confirmed, the superiority being greatest when the boundary layer is very thin. Higher order methods give improved accuracy with negligible increase in cost. A new iterative scheme is proposed for the two-dimensional problem, which requires orders of magnitude fewer arithmetic operations than existing procedures currently available in the literature.

1. INTRODUCTION

The equilibrium (i.e. time-independent) convection-diffusion equation,

\[-\nabla \cdot [D(x)\nabla u] + V(x) \cdot \nabla u = f(x), \quad D(x) > 0,
\]

arises in many applications, such as convective heat and chemical transport. However, methods for solving it numerically have been studied much less intensively than methods for solving another linear elliptic differential equation (DE), the self-adjoint DE

\[-\nabla \cdot [D(x)\nabla u] + q(x)u = f(x), \quad D(x) > 0, \quad q(x) \geq 0;
\]

see for example Birkhoff and Lynch [1]. The present paper is concerned with constructing accurate difference methods for approximating one- and two-dimensional convection-diffusion equations of form (1) and also with finding efficient methods for solving the resulting linear algebraic systems.

Physical intuition often helps in suggesting accurate and efficient numerical methods. Therefore we recall that, in the preceding equations, u(x) can be thought of as representing the concentration of a chemical in a solution flowing with convective velocity V(x). Then D(x) represents a diffusion coefficient, q(x) an absorption coefficient, and f(x) a source term.

Some of the interesting complications that arise in solving convection-diffusion problems numerically already occur in connection with the simple one-dimensional equation

\[-Du'' + Vu' = f(x), \quad 0 < x < a,
\]

with D and V positive constants, and we shall study it first, for given u(0) = u0 and u(a) = u1. By dividing equation (2) by V, and introducing the dimensionless variables \(\xi = x/a\) and \(\epsilon = D/aV > 0\), equation (2) is transformed to

\[-\epsilon u'' + u' = F(\xi), \quad \text{on } [0, 1], \quad F(\xi) = af(a\xi)/V.
\]

Clearly, equations (2) and (3) are singular perturbations of first-order DE's [2, 3]. If one sets \(\epsilon = 0\), then the second-order DE (3) becomes first-order, and its solution is determined by the value of u at a single endpoint. Furthermore, when \(\epsilon \ll 1\), the solution of this first-order equation that satisfies the upstream boundary condition \(u(0) = u_0\), approximates the true solution of equation (3) everywhere except in an \(O(\epsilon)\) interval (e.g. 1 - 10\(\epsilon < \xi < 1\)), which may be called the (downstream) boundary layer. Evidently, the functions 1 and \(e^{\epsilon\xi}\) form a basis of solutions of the homogenous DE \(-\epsilon u'' + u' = 0\), obtained by setting \(F = 0\) in equation (3). Hence for small \(\epsilon\) (the most interesting case), variations in \(u = A + Be^{\epsilon\xi - 1/\epsilon}\) are concentrated in the interval \(1 - 10\epsilon < \xi < 1\), where \(e^{\epsilon(1-1/\epsilon)}\) changes by a factor of 22,000.

For these reasons equation (3) is said to be convection-dominated when \(\epsilon \ll 1\), and diffusion-dominated when \(\epsilon \gg 1\). It is the convection-dominated case that gives rise to numerical difficulties, and we will concentrate on it.

147
A similar classification applies more generally to one-dimensional convection–diffusion equations $-D u'' + V(x) u' = f(x)$ with moderately varying $V(x)$ which occur in chemical engineering applications when $V(x)$ has a small derivative and does not change sign. Introducing $v(\xi) = V(a\xi)/V(a)$ and letting $x$ replace $\xi$, we obtain the one-dimensional convection–diffusion equation in normal form

$$-\varepsilon u'' + v(x) u' = F(x), \quad 0 < x < 1, \quad v(1) = 1, \quad 0 < v_{\min} \leq v(x) = O(1),$$

if we orient the $x$-axis so that the direction of flow is to the right.

2. TWO DIFFERENCE APPROXIMATIONS

We first review two standard difference approximations widely used to solve convection–diffusion equation (4), explaining their limitations, before describing some better methods in Sections 3 and 4.

Let a uniform of $[0, 1]$ by given by $x_i = ih, i = 0, \ldots, n$, with $h = 1/n$. At each interior mesh-point $x_i$, approximate equation (4) using central difference quotients. This gives the 3-point approximating difference equation (AE)

$$-\varepsilon h^2 (u_{i-1} - 2u_i + u_{i+1}) + \frac{v_i}{2h} (-u_{i-1} + u_{i+1}) = F(x_i).$$

(5)

More precisely, we have

$$-\varepsilon h^2 (u_{i-1} - 2u_i + u_{i+1}) + \frac{v_i}{2h} (-u_{i-1} + u_{i+1}) = F(x_i) + T_i,$$

(5')

where the truncation error $T_i$ is defined by equation (5'), and is equal to

$$\frac{v_i h^2}{12} u'''(\eta) + \frac{v_i h^2}{6} u''(\sigma) = O(h^2),$$

for some $\eta, \sigma \in (x_{i-1}, x_{i+1})$; see Conte and de Boor [4, pp. 298–299]. Multiplying equation (5) through by $h^2/\varepsilon$, we get a tridiagonal system of $n - 1$ linear algebraic equations,

$$2U_i = (1 + \sigma_i) U_{i-1} + (1 - \sigma_i) U_{i+1} + \frac{h^2}{\varepsilon} F(x_i), \quad \sigma_i = \frac{hv_i}{2\varepsilon} = \frac{haV_i}{2D}.$$  (6)

For given $U_0 = u(0)$ and $U_n = u(1)$, this system can be quickly solved by band elimination.

The preceding scheme gives satisfactory approximate solutions of equation (4) when $hv_i < 2\varepsilon$, i.e. when $\max_i(hv_i)$ is less than one-fifth the boundary layer thickness $10\varepsilon$. Under these circumstances, the coefficients of $U_{i-1}$ and $U_{i+1}$ are both positive with sum 2, so that when $F(x) = 0$, $U_i$ is a weighted mean of $U_{i-1}$ and $U_{i+1}$. For sufficiently small $h$, one gets accurate approximations $U_i$ to the values $u(x_i)$ of the solution at interior mesh-points. Actually, the error is $O(h^2)$: there is a constant $K$ independent of $h$, such that $|u_i - U_i| \leq Kh^2$ for all sufficiently small $h$. This follows because the truncation error is $O(h^2)$ and, like the DE (1.4), the AE (2.2) is of monotone type [i.e. increasing the $f(x_i)$ for fixed $U_0$ and $U_n$ increases all $U_i, i = 1, \ldots, n - 1$ when $h < 2\varepsilon/v_{\max}$.

However, if $hv_i > 2\varepsilon$, one “weight” is negative and the $U_i$ do not approximate the $u_i$ even qualitatively. To see this, consider the (homogeneous) constant-coefficient case, with $v(x) \equiv 1$ and $F(x) \equiv 0$. The $U_i$ given by equation (6) are then

$$U_i = A + B[(1 + \sigma)/(1 - \sigma)]^i, \quad \sigma = h/2\varepsilon.$$

When $\sigma > 1$, the variations $U_i - U_{i-1}$ computed from equation (6) alternate in sign with increasing amplitude, whereas the true $u(x_i)$ vary monotonically from $u(0)$ to $u(1)$.

**Upwind differencing**

The oscillations just described, which are unrealistic physically, can be avoided for arbitrary $h > 0$ by using the following scheme of upwind differencing. Instead of approximating the convection term $u'$ by the central difference quotient, $(u_{i+1} - u_{i-1})/2h$, use the “upwind” (or “upstream” or “backward”) approximation $u' \equiv (u_i - u_{i-1})/h$. After simplifying, this gives the
approximating $\Delta E$ of monotone type

$$(2 + 2\sigma_j)U_i = (1 + 2\sigma_j)U_{i-1} + U_{i+1} + h^2 F_i/\epsilon, \quad \sigma_j = \epsilon v_i/2\epsilon.$$ (7)

Upwind differencing is widely used in engineering computations when $\pm 5\%$ accuracy is all that is demanded [5, p. 64]. However, its asymptotic error as $h \downarrow 0$ is $O(h)$, instead of $O(h^2)$, as is easily seen. Therefore, we now turn our attention to a much better scheme, due to R. N. de G. Allen.$^*$

3. EXPONENTIAL FITTING

Allen began by constructing a difference approximation which is exact if $v(x) = V = 1$ and $F(x) = K$ are constant in equation (4). We then have $-\epsilon u'' + u' = K$. Its general solution is $A + Be^{\epsilon v}/K$, which satisfies exactly the $\Delta E$

$$(2 \cosh \sigma_i)U_i = e^\sigma U_{i-1} + e^{-\sigma} U_{i+1} + (2h \sinh \sigma_i)K, \quad \sigma = h/2\epsilon.$$ (8)

Notice that the exponentially small coefficient associated with the downwind stencil point, $x_{i+1}$, implies a reduction by a factor of 22,000 in the effect of a change $Au$ in the downstream boundary value $u(1) = u_1$, outside a boundary layer of thickness $10\epsilon$. This agrees with the behavior of the exact solution, as indicated in Section 1.

Allen then proposed using the following analogue of equation (8)

$$(2 \cosh \alpha_i)U_i = e^{\alpha} U_{i-1} + e^{-\alpha} U_{i+1} + (2h \sinh \alpha_i)F_i/\epsilon, \quad \sigma = h/2\epsilon,$$ (9)

also in the general variable coefficient case of equation (4). Allen's scheme has several advantageous properties. It is of monotone type, and $U \equiv 1$ is an exact solution of the homogeneous $\Delta E$, for all $\sigma$. It also has $O(h^2)$ local accuracy, as may be shown by expanding all functions in Taylor series about $x = x_i$. Note finally that, after dividing both sides by $e^{\epsilon v}$ and then letting $\epsilon$ tend to zero, the equation becomes

$$U_i = U_{i-1} + hF_i/\epsilon,$$

which is an $O(h)$ discretization of the "reduced" equation $v(x)u' = F$, obtained from equation (4) by setting $\epsilon$ equal to zero.

Self-adjoint form

Multiplying through by the integrating factor $e^{-x/\epsilon}$, equation (3) assumes the self-adjoint form

$$-(e^{-x/\epsilon} u')' = e^{-x/\epsilon} F(x)/\epsilon.$$

Curiously, if this is approximated with divided central differences as

$$(2 \cosh \sigma_i)U_i = e^\sigma U_{i-1} + e^{-\sigma} U_{i+1} + h^2 F_i/\epsilon, \quad \sigma = h/2\epsilon$$ (10)

the coefficients of the unknowns $U_i$ are the same as in Allen's scheme (8). However, in contrast to Allen's scheme, equation (10) can grossly underestimate the influence of $F$. When one lets $\epsilon$ tend to zero with $h$ and $F$ fixed, the solution of equation (10) tends to a solution of the homogeneous equation $u' = 0$.

Test problem

To test the superiority of Allen's scheme over those discussed in Section 2, we tried all three methods out on the following test problem:

$$v(x) = \frac{2}{x+1}, \quad F(x) = \left(-\epsilon + \frac{2}{x+1}\right)e^x, \quad u(0) = 1 + 2^{-2\epsilon}, \quad u(1) = e + 2.$$

We chose this problem because it had an easily calculated known exact solution for the DE $-\epsilon u'' + v(x)u' = F(x)$, namely

$$u = e^x + (x + 1) \left[\frac{(x + 1)^{2\epsilon}}{2}\right].$$

$^*$See Allen and Southwell [6], in which the method is attributed to Allen.
This test problem was solved on the Southern Methodist University's IBM 3081 computer by the difference approximations described above. Table 1 lists values of $E_n$, the maximum of the errors, $|u_i - U_i|$, for $n = 1/h$; $42(-4)$ denotes $42 \times 10^{-4}$.

The superiority of Allen's scheme is evident. Indeed, as was explained in Section 2, when the problem and $\epsilon$ are fixed the asymptotic error is $O(h)$ for the upwind approximation and $O(h^2)$ for the other methods. In Table 1, values of $p_\epsilon = \log(E_n/E_{2^n})/\log 4$ are listed. Observe that these do tend to 1 and to 2 as $h \downarrow 0$ for the $O(h)$ and $O(h^2)$ methods, respectively, as one would expect for relative errors more than $10^{-10}$, since floating point arithmetic accurate to one part in $10^{14}$ was used. The data in Table 1 confirm these predictions though, also as expected, $n$ must be large when $\epsilon$ is small for the $p_\epsilon$ to display this behavior. For example, when $\epsilon = 0.001$ and $h = 1/256$, there are only two mesh-points in the boundary layer.

The evidence indicates that Allen's scheme is the most accurate for all $n$; for this problem, it is about 30 times more accurate than the divided central difference approximation. Since the maximum value of the solution is about 5, 1% accuracy is obtained when the error is 0.05 or less. Specifically, 1% relative accuracy is obtained with it for $n \geq 4$ when $\epsilon = 1$ or 0.1, and for $n \geq 16$ when $\epsilon = 0.01$ or 0.001.

Considering $\epsilon$ as a parameter, then for each method we have $E_n \cong K(\epsilon) h^p$ for sufficiently small $h$ (neglecting roundoff). From the data in Table 1, $K(\epsilon)$ seems to increase as $\epsilon \downarrow 0$ roughly like $O(\epsilon^{-q})$, where $q = 2, 1, 1$ and 2, for the four methods, respectively.

### 4. HIGHER-ORDER ACCURACY

We now explain how one can obtain higher than $O(h^2)$ accuracy in solving equation (4) by using more than one value of $F(x)$ on the right-hand side of the $\Delta v$. For example, when $v(x) \equiv 0$, one has the classic Stormer–Numerov formula for integrating DE’s of the form $-u'' = F(x, u)$. Writing $F_i = F(x_i, U_i)$, this formula is

$$
\frac{1}{h^2} (-U_{i-1} + 2U_i - U_{i+1}) = \frac{1}{12} (F_{i-1} + 10F_i + F_{i+1}),
$$

(11)

it is well-known to have $O(h^4)$ accuracy. In the linear case, when $F(x, u) = a(x) + b(x)u$, equation (11) reduces to a linear tridiagonal system; it was very thoroughly studied in Birkhoff and Gulati [7]. We show how to obtain similar $O(h^4)$ accuracy for convection–diffusion equations including a term in $u'$ which may even be dominant.

<table>
<thead>
<tr>
<th>$\epsilon$</th>
<th>$n$</th>
<th>Central differences</th>
<th>Upwind difference</th>
<th>Allen's scheme</th>
<th>Self-adjoint form</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_n$</td>
<td>$p_\epsilon$</td>
<td>$E_n$</td>
<td>$p_\epsilon$</td>
<td>$E_n$</td>
<td>$p_\epsilon$</td>
</tr>
<tr>
<td>1.000</td>
<td>4</td>
<td>42(-4)</td>
<td>63(-3)</td>
<td>14(-5)</td>
<td>21(-5)</td>
</tr>
<tr>
<td>16</td>
<td>26(-5)</td>
<td>18(-3)</td>
<td>0.9</td>
<td>99(-7)</td>
<td>1.9</td>
</tr>
<tr>
<td>64</td>
<td>16(-6)</td>
<td>2.0</td>
<td>47(-4)</td>
<td>1.0</td>
<td>62(-6)</td>
</tr>
<tr>
<td>256</td>
<td>10(-7)</td>
<td>2.0</td>
<td>12(-4)</td>
<td>1.0</td>
<td>39(-9)</td>
</tr>
<tr>
<td>1024</td>
<td>64(-9)</td>
<td>2.0</td>
<td>30(-5)</td>
<td>1.0</td>
<td>25(-10)</td>
</tr>
<tr>
<td>0.100</td>
<td>4</td>
<td>48(-2)</td>
<td>48(-2)</td>
<td>48(-3)</td>
<td>32(-2)</td>
</tr>
<tr>
<td>16</td>
<td>29(-3)</td>
<td>2.0</td>
<td>21(-2)</td>
<td>0.6</td>
<td>36(-4)</td>
</tr>
<tr>
<td>64</td>
<td>18(-4)</td>
<td>2.0</td>
<td>62(-3)</td>
<td>0.9</td>
<td>23(-5)</td>
</tr>
<tr>
<td>256</td>
<td>11(-5)</td>
<td>2.0</td>
<td>16(-3)</td>
<td>1.0</td>
<td>14(-6)</td>
</tr>
<tr>
<td>1024</td>
<td>69(-7)</td>
<td>2.0</td>
<td>41(-4)</td>
<td>1.0</td>
<td>89(-8)</td>
</tr>
<tr>
<td>0.010</td>
<td>4</td>
<td>79(-1)</td>
<td>21(-2)</td>
<td>1.4</td>
<td>31(-2)</td>
</tr>
<tr>
<td>16</td>
<td>11(-1)</td>
<td>1.4</td>
<td>31(-2)</td>
<td>-0.3</td>
<td>36(-3)</td>
</tr>
<tr>
<td>64</td>
<td>18(-2)</td>
<td>1.3</td>
<td>37(-2)</td>
<td>-0.1</td>
<td>39(-4)</td>
</tr>
<tr>
<td>256</td>
<td>95(-4)</td>
<td>2.1</td>
<td>13(-2)</td>
<td>0.8</td>
<td>26(-5)</td>
</tr>
<tr>
<td>1024</td>
<td>59(-5)</td>
<td>2.0</td>
<td>35(-3)</td>
<td>0.9</td>
<td>16(-6)</td>
</tr>
<tr>
<td>0.001</td>
<td>4</td>
<td>84(0)</td>
<td>21(-2)</td>
<td>1.4</td>
<td>31(-2)</td>
</tr>
<tr>
<td>16</td>
<td>52(-1)</td>
<td>2.0</td>
<td>79(-3)</td>
<td>0.5</td>
<td>48(-3)</td>
</tr>
<tr>
<td>64</td>
<td>15(-1)</td>
<td>0.9</td>
<td>13(-2)</td>
<td>-0.4</td>
<td>12(-3)</td>
</tr>
<tr>
<td>256</td>
<td>69(-2)</td>
<td>0.6</td>
<td>37(-2)</td>
<td>-0.8</td>
<td>20(-4)</td>
</tr>
<tr>
<td>1024</td>
<td>56(-3)</td>
<td>1.7</td>
<td>26(-2)</td>
<td>0.3</td>
<td>17(-5)</td>
</tr>
<tr>
<td>4096</td>
<td>37(-4)</td>
<td>2.1</td>
<td>82(-3)</td>
<td>0.8</td>
<td>11(-6)</td>
</tr>
<tr>
<td>16384</td>
<td>23(-5)</td>
<td>2.0</td>
<td>22(-3)</td>
<td>1.0</td>
<td>69(-8)</td>
</tr>
</tbody>
</table>

$p_\epsilon = \log(E_{n-1}/E_n)/\log(4)$; $42(-4)$ denotes $42 \times 10^{-4}$. 

G. Birkhoff et al.
Constant velocity

We first consider the case of constant \( v = 1 \), setting \(-cu'' + u' = F(x)\) as in equation (3). Using the HODIE method of Lynch and Rice [8, 9] (see also Doedel [10]) we look for coefficients \( \alpha_j \) and \( \beta_j \) which make the 3-point formula

\[
\alpha_{-1} U_{i-1} + \alpha_0 U_i + \alpha_1 U_{i+1} = \beta_{-1} F_{i-1} + \beta_0 F_i + \beta_1 F_{i+1}
\]

have the highest order of accuracy possible. This is done by selecting the \( \alpha \) and \( \beta \) so that the solution \( U \) of the difference equation is the same as the solution \( u \) of the DE whenever \( u \) is in specified subspace of functions.

Thus, by setting \( \alpha_{-1} = -e^\sigma \), \( \alpha_1 = -e^{-\sigma} \) and \( \alpha_0 = 2 \cosh \sigma \), where \( \sigma = h/2\epsilon \), one gets the same \( \alpha_j \) as in Allen's scheme (8). When \( F \equiv 0 \), (4.2) has the same solution as the DE, which is of the form \( A + B e^{\epsilon x} \).

Next we choose the \( \beta_j \) so that not only all \( A + B e^{\epsilon x} \), but also \( x \), \( x^2 \) and \( x^3 \), satisfy equation (4.2) exactly for the corresponding \( F(x) = -cu'' + u' \)—namely, for \( 1 \), \( -2\epsilon + 2x \), and \( -6\epsilon x + 3x^2 \), respectively. This will make equation (12) exact for all solutions of equation (3) with quadratic \( F(x) \).

For the \( \beta_j \) specified, substitution into equation (12) yields a system of three linear equations for the \( \beta_j \) whose solutions are (see also Table 2)

\[
\begin{align*}
\beta_{-1} &= h [(1 - \sigma^2/3) (\sinh \sigma)/\sigma - (1 - \sigma)e^\sigma]/2\sigma, \\
\beta_0 &= -h [(1 - 4\sigma^2/3)(\sinh \sigma)/\sigma - \cosh \sigma]/\sigma, \\
\beta_1 &= h [(1 - \sigma^2/3)(\sinh \sigma)/\sigma - (1 + \sigma)e^{-\sigma}]/2\sigma.
\end{align*}
\]

Optimal evaluation points

In addition to the freedom in choosing basis functions for the HODIE approximation scheme, one has freedom in choosing the points of evaluation of \( F \) in equation (12). One can select arbitrarily \( J \) evaluation points, \( \tau_{i,1}, \ldots, \tau_{i,J} \), in each subinterval \([x_{i-1}, x_{i+1}]\), and replace the right-hand side of equation (12) with

\[
\sum_{j=1}^{J} \beta_{i,j} F(\tau_{i,j}).
\]

Each increase by one in \( J \) allows one to include at least one additional basis function.

As in Lynch and Rice [9], one can use “Gauss evaluation points” in the HODIE method applied to ordinary DE’s, so as to obtain higher-order accuracy with the same number of evaluation points. It was shown there that for each \( J \), there exist \( J \) “Gauss points” whose use maximizes the order of accuracy as in the Gauss quadrature: the points are zeros of a \( J \)th degree orthogonal polynomial with respect to the local Green’s function of the operator (i.e. the Green’s function for an interval of length \( 2h \)). When \( \epsilon \ll h \), this Green’s function is strongly skewed in the upstream direction; hence so are these optimal evaluation points.

In this case, we find the location \( \tau h \) of a single “Gauss” evaluation point which allows exact results for all solutions of the form

\[ u(x) = A + Be^{\epsilon x} + Cx + Dx^2. \]

By translation invariance, and because the coefficients of Allen’s scheme give a \( \Delta E \) which is exact when \( F(x) \equiv K \), it suffices for \( O(h^4) \) accuracy to make the \( \Delta E \)

\[(2 \cosh \sigma) U_0 = e^{\epsilon x} U_{-1} + e^{-\epsilon} U_1 + \beta F(\tau h), \quad \sigma = h/2\epsilon,\]

Table 2. HODIE coefficients for equation (12); \( v = 1 \)

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>( h_{-1}/s )</th>
<th>( h_0/s )</th>
<th>( h_1/s )</th>
<th>( \beta_{-1}/s )</th>
<th>( \beta_0/s )</th>
<th>( \beta_1/s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>-50.5</td>
<td>100.0</td>
<td>-49.5</td>
<td>0.0833</td>
<td>0.833</td>
<td>0.0833</td>
</tr>
<tr>
<td>0.10</td>
<td>-5.52</td>
<td>10.0</td>
<td>-4.52</td>
<td>0.0917</td>
<td>0.833</td>
<td>0.0751</td>
</tr>
<tr>
<td>1.00</td>
<td>-1.16</td>
<td>1.31</td>
<td>-0.157</td>
<td>0.167</td>
<td>0.823</td>
<td>0.0101</td>
</tr>
<tr>
<td>10.00</td>
<td>-1.00</td>
<td>1.00</td>
<td>-21 (-10)</td>
<td>0.369</td>
<td>0.712</td>
<td>-0.0808</td>
</tr>
<tr>
<td>( \infty )</td>
<td>-1.00</td>
<td>1.00</td>
<td>0</td>
<td>0.417</td>
<td>0.667</td>
<td>-0.0833</td>
</tr>
</tbody>
</table>

\( \tau h = \beta_{-1} + \beta_0 + \beta_1. \)
exact for \( u(x) = x \) and \( u(x) = x^2 \). Because \( U_0 \) is zero for both these functions, we get the two equations

\[
    h(-e^x + e^{-x}) + \beta = 0, \quad h^2(e^x + e^{-x}) + 2\beta (th - c) = 0.
\]

Hence

\[
    \beta = 2h \sinh \sigma \quad \text{and} \quad \tau = \frac{c}{h} \coth \sigma = \frac{1}{2} \left[ \frac{1}{\sigma} - \coth \sigma \right],
\]

and thus \(-1/2 < \tau < 0\), and \( \tau \to 0 \) as \( \sigma \to 0 \), and \( \tau \to -1/2 \) as \( \sigma \to \infty \). In the limit as \( \epsilon \downarrow 0 \) with \( h \) fixed, the difference equation becomes

\[
    U_{i+1} = U_i + hF(x_i - h/2),
\]

which is the midpoint rule for solving the reduced equation \( u' = F(x) \) with local \( O(h^2) \) accuracy.

**Variable velocity**

In the more general case of variable velocity \( v(x) \), Dennis [11] has shown that one can obtain \( O(h^4) \) accuracy by making the substitution \( u = w \exp((2\epsilon)^{-1} \int v \, dx) \) for the independent variable. This transforms the DE to \( w'' + R(x)w = S(x)f(x) \) so that one can first construct the Stormer–Numerov approximation, and then revert to the original unknown \( u \) to obtain a variant of Allen’s scheme. We now compare his method with that given by the general HODIE procedure.

When \( v(x) \) is variable, one cannot find analytically an exact solution of \(-\epsilon u'' + v(x)u' = F(x)\) for general \( F(x) \). Instead, one first chooses a basis of functions \( u_k(x) \) for which equation (12) is to be made exact, and then determines analytically the input functions \( F_k(x) \), which, for given \( \epsilon \) and \( v(x) \), makes \(-\epsilon u_k'' + v(x)u_k' = F_k(x)\). One gets for each mesh-point \( x_i \) and each \( k \), values \( F_{k,i} \), and corresponding \( U_{k,i} = u_k(x_i) \) to substitute into equation (12). To construct difference approximation (12), the computer solves the linear system

\[
    a_{i-1} u_{k,i-1} + a_{i} u_{k,i} + a_{i+1} u_{k,i+1} = \beta_{i-1} F_k(x_{i-1}) + \beta_{i} F_k(x_{i}) + \beta_{i+1} F_k(x_{i+1}), \quad k = 0, \ldots, 4, \quad (14)
\]

for the \( \alpha \)s and \( \beta \)s; here we have normalized with the choice \( \beta_{i,0} = 1 \).

For example, when one takes solutions

\[
    u_k = (x - x_i)^k \quad \text{for} \quad k = 0, \ldots, 3, \quad \text{and} \quad u_4(x) = \exp((v_0[x - x_i]/\epsilon),
\]

then the input functions are

\[
    F_k(x) = -ek(k - 1)(x - x_i)^k - v(x)k(x - x_i)^{k-1}, \quad \text{for} \quad k = 0, \ldots, 3,
\]

\[
    F_4(x) = v_1(-v_i + v(x)) \exp((v_0[x - x_i])/\epsilon).
\]

System (12) is

\[
    \alpha_{i-1} + \alpha_{i,0} + \alpha_{i,1} = 0,
\]

\[
    -h\alpha_{i-1} + h\alpha_{i,1} = v_{i-1} \beta_{i-1} + v_{i} + v_{i+1} \beta_{i,1},
\]

\[
    h^2\alpha_{i-1} + h^2\alpha_{i,1} = -2(\epsilon v_i + v_{i-1}) \beta_{i-1} - 2(\epsilon v_i - v_{i+1}) h \beta_{i,1},
\]

\[
    -h^2\alpha_{i-1} + h^2\alpha_{i,1} = (6 \epsilon v_i h + 3 v_{i+1} h^2) \beta_{i-1} - (6 \epsilon v_{i} h - 3 v_{i-1} h^2) \beta_{i,1},
\]

\[
    -\epsilon k \alpha_{i-1} + \alpha_{i,0} + \epsilon k \alpha_{i,1} = (-v_i + v_{i-1}) e^{-\epsilon k \beta_{i-1}/\epsilon_i} - (v_{i-1} + v_{i+1}) e^{\epsilon k \beta_{i,1}/\epsilon_i},
\]

where \( \epsilon_i = \epsilon/v(x_i) \).

We obtain improved accuracy for small \( \epsilon \) when the three evaluation points are equally spaced in \( x_{i-1} \leq x \leq x_i \) rather than in \( x_{i-1} \leq x \leq x_{i+1} \), or when the function \( u_i = (x - x_i)^3 \) above is replaced with \( (x - x_i) \exp((x - x_i)/\epsilon_i) \).

**Uniform methods**

Difference schemes having uniform \( Kh^p \) error bounds, with \( K \) uniformly bounded as the singular perturbation parameter \( \epsilon \) tends to zero, have obvious theoretical advantages. Standard polynomial-based schemes (on uniform meshes), such as equations (5) and (6), as well as Swartz’s [12, p. 304]
Solution of convection–diffusion equations

### Table 3. Maximum errors and experimental orders of accuracies

<table>
<thead>
<tr>
<th>ε</th>
<th>n</th>
<th>(E_n)</th>
<th>(p_n)</th>
<th>(E_n)</th>
<th>(p_n)</th>
<th>(E_n)</th>
<th>(p_n)</th>
<th>(E_n)</th>
<th>(p_n)</th>
<th>(E_n)</th>
<th>(p_n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.100</td>
<td>4</td>
<td>17 (−3)</td>
<td>62 (−3)</td>
<td>93 (−3)</td>
<td>14 (−4)</td>
<td>10 (−4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>70 (−5)</td>
<td>31 (−5)</td>
<td>21 (−5)</td>
<td>4.4</td>
<td>1.8</td>
<td>76 (−7)</td>
<td>3.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>27 (−8)</td>
<td>12 (−7)</td>
<td>4.0</td>
<td>70 (−7)</td>
<td>2.0</td>
<td>31 (−9)</td>
<td>4.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>13 (−10)</td>
<td>49 (−10)</td>
<td>4.0</td>
<td>44 (−8)</td>
<td>2.0</td>
<td>12 (−11)</td>
<td>4.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>61 (−13)</td>
<td>19 (−12)</td>
<td>4.0</td>
<td>27 (−9)</td>
<td>2.0</td>
<td>21 (−13)</td>
<td>2.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.010</td>
<td>4</td>
<td>15 (−2)</td>
<td>43 (−6)</td>
<td>33 (−3)</td>
<td>12 (−8)</td>
<td>2.0</td>
<td>10 (−11)</td>
<td>4.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>23 (−3)</td>
<td>1.4</td>
<td>91 (−2)</td>
<td>2.6</td>
<td>21</td>
<td>39 (−5)</td>
<td>1.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>22 (−5)</td>
<td>3.3</td>
<td>299 (−4)</td>
<td>3.3</td>
<td>1.7</td>
<td>57 (−7)</td>
<td>3.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>42 (−8)</td>
<td>4.1</td>
<td>33 (−5)</td>
<td>3.9</td>
<td>1.9</td>
<td>27 (−9)</td>
<td>3.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>10 (−10)</td>
<td>4.0</td>
<td>12 (−8)</td>
<td>4.0</td>
<td>2.0</td>
<td>10 (−11)</td>
<td>4.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.001</td>
<td>4</td>
<td>38 (−2)</td>
<td>43 (−2)</td>
<td>14 (−2)</td>
<td>1.9</td>
<td>1.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>18 (−2)</td>
<td>0.5</td>
<td>19 (1)</td>
<td>2.3</td>
<td>1.9</td>
<td>50 (−5)</td>
<td>1.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>19 (−3)</td>
<td>1.6</td>
<td>23 (−1)</td>
<td>3.2</td>
<td>2.1</td>
<td>33 (−6)</td>
<td>2.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>59 (−5)</td>
<td>2.5</td>
<td>53 (−2)</td>
<td>0.8</td>
<td>2.2</td>
<td>13 (−7)</td>
<td>2.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1024</td>
<td>31 (−7)</td>
<td>3.8</td>
<td>10 (−2)</td>
<td>1.2</td>
<td>1.7</td>
<td>11 (−9)</td>
<td>3.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(\dagger\) Basis 1, \(x, x^2, x^3, \exp(\epsilon \int v(x) dx);\) evaluation points \(x_i - h, x_i, x_i + h.\)

\(\ddagger\) Basis 1, \(x, x^2, \exp(\epsilon \int v(x) dx), x \exp(\epsilon \int v(x) dx);\) evaluation points \(x_i - h, x_i, x_i + h.\)

| Error significantly larger than 100%. |

### O(\(h^4\)) interpolation scheme, do not have this desirable property. However, Allen’s scheme (8) is uniformly \(O(h)\) (see Il’in [13], Kellogg and Tsan [14], and the last paragraph of Section 3), and the same is true of the HODIE scheme above with three evaluation points, which has an order of accuracy \(O(h^4)\) for fixed \(\epsilon\) (vs \(O(h^2)\) for Allen’s scheme). The \(O(h^2)\) scheme of El Mistikawy and Werle [15] is uniformly \(O(h^4)\) [16].

Actually, by constructing a difference approximation which is exact locally on the family of functions

\[1, x, \ldots, x^J, \exp(\epsilon \int v(x) dx), x \exp(\epsilon \int v(x) dx), \ldots, x^J \exp(\epsilon \int v(x) dx), \] \(15\)

one can obtain for any \(J\) an exponentially fitted HODIE scheme that is uniformly \(O(h^4)\); see Gartland [17].

Experimental results for the test problem of Section are listed in Table 3. For the first HODIE scheme, when \(h\) is very small the error seems to be roughly proportional to \(h^4/\epsilon^3\); one can prove that it is \(O(h^4)\) for any fixed \(\epsilon\), as \(h \to 0\). For the second HODIE scheme, with basis (15) and \(J = 2\), the data indicate that the error is roughly proportional to \(h^4/\epsilon^2\), as in Gartland’s theoretical results.

### 5. THE BOUNDARY LAYER

In the convection dominated case (\(\epsilon \ll 1\)), as was mentioned in Section 1, there is a thin “boundary layer” of width the order of \(10\epsilon\) at the downstream end of the flow defined by equation (4). Since the flow velocity \(v(x)\) usually varies little in this layer, a fairly good first approximation in the boundary layer itself is ordinarily given by the model problem with constant velocity, \(v(x) = 1\).

#### Outer solution

Outside the boundary layer, we can obtain an asymptotic expansion (valid as \(\epsilon \to 0\)) in powers of \(\epsilon\):

\[u(x) \approx z(x; \epsilon) - z_0(x) + \epsilon z_1(x) + \epsilon^2 z_2(x) + \cdots, \] \(16\)

using standard methods of singular perturbation theory (see O’Malley [2] or Carrier and Pearson [3]). Moreover, the \(z_n(x)\) can be computed recursively, as we shall now explain.

In the constant coefficient case, \(v(x) \equiv 1\), we have simply

\[u(x) \approx z_0(x) = u(0) + \int_{0}^{x} F(\xi) \, d\xi. \] \(17\)

This is intuitively obvious: if the diffusivity \(\epsilon\) is zero, we have a simple problem in pure convection, and equation (3) reduces to \(u' = F(x)\). Substituting the series for \(z\) for \(u\) in \(-\epsilon u'' + u' = F(x)\),
rearranging and equating coefficients of powers of $\epsilon$ to zero, we get equations for $z_m$:

\[ z_1 = z_0, \quad z_1(0) = 0; \quad \ldots, \quad z_m = z_m, \quad z_m(0) = 0; \quad \ldots \]

This gives

\[ z_m(x) = \int_0^x z_{m-1}(\xi) \, d\xi, \]

which leads to

\[ u(x) \approx z(x; \epsilon) \approx u(0) + \int_0^x F(\xi) \, d\xi + \epsilon[F(x) - F(0)] + \epsilon^2[F''(x) - F''(0)] + \cdots + \epsilon^M[F^{(M-1)}(x) - F^{(M-1)}(0)] + \cdots \quad (18) \]

More generally, for any $v(x) > 0$, (4) reduces to $v(x)u' = F(x)$ with solution

\[ u(x) = z_0(x) = u(0) + \int_0^x \frac{F(\xi)}{v(\xi)} \, d\xi. \]

Typically, $F(x)$ and $v(x)$ are slowly varying functions of $x$ and numerical quadrature must be used to compute accurate values of $u_0(x)$ at all mesh-points $x_i$. The equations

\[ z_1 = z_0' = \frac{[F(x)/v(x)]'}{v(x)} \quad z_2 = z_1' = \frac{[F(x)/v(x)]'}{[v(x)]'} / v(x), \quad \ldots, \quad (19) \]

which give the other terms in the asymptotic series (16) can be treated similarly.

**Boundary layer correction**

As $\epsilon \downarrow 0$, the upstream solution depends exponentially little on the downstream boundary value $u(1)$. The first few terms of the asymptotic expansion (16) also give more accurate representations of the solution. Inside the boundary layer, expansion (16) fails to give an accurate approximation. There one must either add to $z(x; \epsilon)$ a boundary layer correction, $w(x; \epsilon)$, or find an inner solution which blends smoothly into the outer solution, $z(x; \epsilon)$.

We write

\[ u(x) = z(x; \epsilon) + w(x; \epsilon). \]

Because the problem is linear, the boundary layer correction $w$ satisfies the homogeneous DE

\[ -\epsilon w''(x; \epsilon) + v(x)w'(x; \epsilon) = 0 \]

subject to the boundary conditions

\[ w(0) = 0, \quad w(1) = u(1) - z(1; \epsilon). \]

For the special case the $v(x) \equiv 1$, we have the explicit formula

\[ w(x; \epsilon) = [u(1) - z(1; \epsilon)](1 - e^{1/\epsilon})/(1 - e^{1/\epsilon}) = [u(1) - z(1; \epsilon)]e^{(1-1/\epsilon)} + O(e^{-1/\epsilon}). \]

In the variable coefficient case when $v(x)$ varies only slightly in the boundary layer, then $w(x; \epsilon)$ can also be used, added to $z(x; \epsilon)$.

**Inner solution**

We can also compute approximate values of the inner solution by any of the finite difference methods discussed in Sections 2-4. One picks a point $X$ outside the boundary layer but close to its edge, say $X = 1 - 10\epsilon$, and solves

\[ -\epsilon w'' + v(x)w' = F(x), \quad \text{on} \quad X < x < 1, \quad w(X) = z(X; \epsilon), \quad w(1) = u(1). \]

Because the interval $[X, 1]$ is so short, only a few (say less than 20) mesh-points in it are required. We used this method to solve the test problem of Section 3 with $\epsilon = 1/100$. We took $1 - X = 10\epsilon = 1/10$ and used the trapezoid rule for the initial value problem and the divided central difference approximation with mesh-lengths $h_{\text{out}} = X/N_{\text{out}}$ and $h_{\text{in}} = (1 - X)/N_{\text{in}}$, respectively. Some results are listed below, where $M$ denotes the number of terms of the asymptotic expansion used to compute $u_{\text{out}}$ [see expression (5.3)]. The problem was solved on a VAX 11/780, with single precision arithmetic (about seven significant decimal digits of accuracy).
Table 4 lists the following maximum errors:

\[ E_{\text{out}} = \max_{j \in \{0, \ldots, N_{\text{out}}\}} |U_j - u(j/N_{\text{out}})|, \quad E_{\text{in}} = \max_{k \in \{0, \ldots, N_{\text{in}}\}} |U_{N_{\text{out}} + k} - u(X + k/N_{\text{in}})|. \]

In this problem, both maximum errors are attained at \( x = X \); this explains why both tabulated errors approach the same value as the number of subintervals increases.

The error of the trapezoidal method decreases by a factor of about four each time the number of subintervals, \( N_{\text{out}} \), is doubled, and for 320 subintervals, the error is nearly at roundoff level. Thus the listed values for 320 subintervals are close to the error \( |u(X) - u_{\text{out}}(X)| \) when using \( M \) terms of the asymptotic series for \( u_{\text{out}} \); this error is about 0.018, 0.00049 and 0.000061, for \( M = 1, 2 \) and 3, respectively. Since the maximum of \( u \) is a little less than five, the last value, 0.000061, is close to roundoff level and it seems unlikely that more terms would improve the accuracy with single precision VAX arithmetic.

Note that with \( M = 1 \) and \( N_{\text{out}} = 5, N_{\text{in}} = 10 \), the error is 0.08, which is less than 2\% (0.08/5). Doubling the number of subintervals, \( (N_{\text{out}}, N_{\text{in}}) \), to \( (10, 20) \), the error is reduced to 0.6\% and one more doubling gives nearly the smallest error possible, 0.018 or 0.4\%, with \( M = 1 \). With \( M = 3 \), we obtain errors a little larger than 0.1\% with the pair of interval numbers \( (5, 10) \)—i.e. by solving one tridiagonal system with nine unknowns.

Instead of using one uniform spacing in the outer region and a different uniform spacing in the inner region, one can use unequal spacings throughout. For an adaptive procedure to optimize such spacings, see Pearson [18].

6. A MODEL TWO-DIMENSIONAL PROBLEM

We conclude by taking up a model two-dimensional convection–diffusion problem: the specialization of equation (1) to the case of flow parallel to the walls of a rectangular channel. The relevant DE is

\[ \nu u_x = \epsilon (u_{xx} + u_{yy}) + F(x, y), \]

where \( \nu = \nu(x, y) \), with \( \nu(1, y) = O(1) \). For this elliptic equation, we assume Dirichlet boundary conditions. Thus we set

\[ u(0, y) = g_0(y), \quad u(1, y) = g_1(y), \]

and (taking the channel width as \( b \))

\[ u(x, 0) = \phi(x), \quad u(x, b) = \psi(x). \]

Typically, \( \phi(x) \equiv \psi(x) = 0 \). Moreover, the exact solution of equation (20) has boundary layers of thickness \( O(\epsilon^{1/2}) \) along the sides of the channel, \( y = 0 \) and \( y = b \), as well as a downstream boundary layer of thickness \( O(\epsilon) \) at \( x = 1 \); see Carrier and Pearson [3, p. 287] and Eckhaus [19, p. 255]. See Johnson–Schatz–Wahlbin [20], and the references they cite, for finite element methods for treating the boundary layers at the channel edges; see also Wahlbin [21].

The standard 5-point central difference approximation to equation (20) is

\[ 2(1 + \beta^2)U_{i,j} = (1 + \sigma_{i,j})U_{i-1,j} + (1 - \sigma_{i,j})U_{i+1,j} + \beta^2(U_{i,j-1} + U_{i,j+1}) + h^2 F_{i,j}/\epsilon, \]

\[ \beta = h/k \quad \sigma_{i,j} = hv_{i,j}/2\epsilon. \]
This is quite satisfactory if \( v_i h \ll 2 \varepsilon \), for reasons similar to those given in Section 2 for the one-dimensional case. For \( \Delta x = h = 1/J \) and \( \Delta y = k = b/J \), the unknowns \( U_{i,j} \) can be conveniently listed lexicographically by columns of the mesh: \((i,j) < (i',j')\) [in words: \((i,j)\) precedes \((i',j')\)] if and only if \( i < i' \) or \( i = i' \) and \( j < j' \). Considering \( U_{i,j} \) as an approximation to \( u(x_i, y_j) = u(ih, jk) \), and solving system (21) by band elimination, one then gets the desired approximate solution in about \((I - 1)(J - 1)^3\) multiplications (and about as many additions), or about \( J^2 \) per mesh-point, and with about \( J^2 \) words of memory.

When \( h \) is large compared with the boundary layer thickness, one should proceed in two stages. First solve an appropriate \( \Delta E \) accurately in the main flow ("outer region"), as efficiently as possible. This gives approximate values \( U_{i-1,j} \) at the beginning of the last interval, \( 1 - h < x < 1 \), which are used as "upstream" boundary conditions for the equations on a refined mesh in this last interval.

For very small \( \varepsilon \) (convection-dominated case), with a fine mesh (but with \( \sigma_{i,j} = h \sigma_{i,j}/2 \varepsilon > 1 \)), an adequate approximation outside the downstream boundary layer is often obtained by neglecting streamwise diffusion (the term \( \varepsilon u_{xx} \)). This replaces equation (20) by the parabolic diffusion equation

\[
v(x, y)u_x = \varepsilon u_{yy} + F(x, y).
\]

Many discretizations of the DE (22) are discussed in Richtmyer and Morton [22]. Of these, the explicit Courant–Friedrichs–Lewy scheme and the implicit Crank–Nicolson scheme are probably the ones most commonly used. We shall not discuss these well-known methods here, nor the error committed by neglecting the \( \varepsilon u_{xx} \) term.

A new procedure

Instead, we describe a new procedure for solving the two-dimensional convection–diffusion model problem under discussion. First, we propose combining Allen's approximation of \(- \varepsilon u_{xx} + \varepsilon u_x\) with the divided central second difference approximation of \(- \varepsilon u_{yy}\). In the constant coefficient case, the system can be written as

\[
\frac{v}{h} \left[ -\frac{1}{1 - e^{-\theta}} U_{i-1,j} + \frac{1 + e^{-\theta}}{1 - e^{-\theta}} U_{i,j} - \frac{1}{e^\theta - 1} U_{i,j+1} \right] + \frac{\varepsilon}{k^2} \left[ -U_{i,j-1} + 2U_{i,j} - U_{i,j+1} \right] = F_{i,j},
\]

where \( \theta = \varepsilon h/\varepsilon = 2\sigma \) and \( i = 1, \ldots, I - 1, j = 1, \ldots, J - 1 \). At boundary mesh-points, \( U \) is equal to the given values of \( u \). In the variable coefficient case, \( v \) and \( \theta \) in equation (23) are replaced with \( v_{i,j} \) and \( \theta_{i,j} \).

When discussing the hand relaxation solution of this system, Allen [23, p. 20] remarks: "Relaxation using equation (23) is always easy and, surprisingly, becomes easier as \( \theta \) increases in value." We will describe a rapidly convergent automated procedure of solution, in the convection-dominated case (\( \varepsilon < 1 \)) when \( \theta > 1 \); that is, with \( h \) large compared with the downstream boundary layer thickness. Formula (23) gives accurate values of \( u \) outside the boundary layer. Having computed these, as was stated above, one can solve this system again on a refined mesh inside the boundary layer, on \([I - 1]h, 1] \times [0, b] \), using the outer values \( U_{i-1,j} \) as boundary values.

Let \( U_i = (U_{i,1}, U_{i,2}, \ldots, U_{i,J-1})^T \) be the vector of unknowns along the mesh-line \( x_i = ih \). System (23) can be written in block form as

\[
-R U_{i-1} + C U_i - S U_{i+1} = F_i + b_i,
\]

where \( (F_i) = F_{i,j} \), and \( b_i = -(\varepsilon/k^2)(u(ih, 0), 0, \ldots, 0, u(ih, b))^T \). Here \( R \) and \( S \) are diagonal matrices and \( C \) is a tridiagonal \((J - 1) \times (J - 1)\) matrix. When \( v \) is constant,

\[
R_{i,j} = \frac{v}{h(1 - e^{-\theta})}, \quad S_{i,j} = S = \frac{v}{h(e^\theta - 1)},
\]

and

\[
C_{j-1,j} = C_{j,j+1} = -\frac{\varepsilon}{k^2}, \quad C_{j,j} = \frac{v(1 + e^{-\theta})}{h(1 - e^{-\theta})} + \frac{2\varepsilon}{k^2} = r + s + \frac{2\varepsilon}{k^2}.
\]

Because \( \theta = \varepsilon h/\varepsilon > 1 \), the matrix \( S = sl \) in equation (24) is exponentially smaller than the others, and outside the boundary layers \( U_{i,j} \) approximates a solution of \( \varepsilon u_x = \varepsilon u_{yy} + F \). Furthermore, the
tridiagonal matrix $C$ is very strongly diagonally dominant. Therefore, the line Gauss–Seidel method converges very rapidly (and line SOR even more rapidly).

Accordingly, we consider the following line Gauss–Seidel scheme. After choosing initial guesses for the $U_{i}^{(n)}$, $i = 1, \ldots, I-1$, say zero, or a blending function (see Birkhoff and Lynch [1, Chap. 7, Section 9]) which interpolates the boundary data, solve the tridiagonal systems

$$CU_{i}^{(n+1)} = RU_{i-1}^{(n+1)} + h F_{i} + b,$$

in the order $i = 1, \ldots, I-1$, for each of $m = 0, 1, \ldots$. In this constant coefficient case, the $LU$-factorization of $C$ needs to be computed only once.

The spectral radius of this scheme for the constant coefficient case is derived in the Appendix; it is

$$\rho = \frac{4rs \cos^2(\pi/J)}{2\varepsilon [1 - \cos(\pi/J)]/k^2 + r + s}.$$

In any norm $\| \cdot \|$, the iteration error $E^{(m)} = U^{(m)} - U$ satisfies $\| E^{(m)} \| \leq K \rho^m$ asymptotically [1, p. 129]; thus the error reduction factor $R^{(m)} = \| E^{(m-1)} \| / \| E^{(m)} \|$ per step satisfies

$$R^{(m)} \approx 1/\rho > 0.25(e^{\frac{h}{\varepsilon}} + 2 + \varepsilon^{-h/\varepsilon}) = \cosh^2(\frac{h}{2\varepsilon}) \equiv 1/\rho_u,$$

asymptotically when $v = 1$; here we have defined the lower bound $1/\rho_u$. For $h/\varepsilon = 0.01, 0.1, 1, 10$, and $100$, the value of the lower bound, $1/\rho_u$, of the error reduction factor is

$$1.000025, \ 1.0025, \ 1.27, \ 5.5 \times 10^3, \ 6.7 \times 10^{43},$$

respectively. We have done numerous experiments with the line Gauss–Seidel method solving equation (23) on $[0, 1] \times [0, 1]$ with $h = k$ for $h/\varepsilon$ between 3 and 250. These results confirm that after a few iterations the error reduction factor is given by $1/\rho$, with $\rho$ as in equation (28).

For another analysis of a Gauss–Seidel method applied to the conduction convection equations, see Ill’in and Kellogg [24].

**SOR**

It is well-known that in the diffusion-dominated case, SOR converges an order of magnitude faster than the Gauss–Seidel scheme. The situation in the convection-dominated case is quite different. In either case, the optimal overrelaxation factor for SOR is $\omega^* = 2/(1 + [1 - \rho]^{1/2})$ and its spectral radius is $\rho^* = \omega^* - 1$ [1, Chap. 4]. In the convection-dominated case, $\rho$ is very small; hence $\rho^* = 1 + \rho/4 + O(\rho^2)$ and thus $1/\rho_u \geq (e^{h/\varepsilon} + 2 + e^{-h/\varepsilon})$, so the asymptotic reduction factor for optimal line SOR is four times that for line Gauss–Seidel, when $h$ is large compared with $\varepsilon$. (In contrast, when $h$ is small compared with $\varepsilon$, $\rho = 1 - \eta$ is close to unity, and then $\rho_b = 1 - 2(2\eta)^{1/2} + O(\eta^{3/2})$ [1, p. 139].) In our numerical experiments, the Gauss–Seidel scheme converged to within the discretized error in one or two iterations.

**Operation count**

About $7m + 3/I$ operations (+, *, /) per unknown are required for $m$ iterations; evaluation of the right-hand sides and the back substitutions account for $7m$ of these and $3/I$ are required for the single $LU$-factorization of the $(J - 1) \times (J - 1)$ tridiagonal matrix $C$.

The iteration error is reduced by a factor much greater than $10^6$ in a single iteration when $\varepsilon = 1/1000$ and $h \leq 1/32$. Thus one iteration is sufficient and the “work” per unknown is about 8. This is significantly less than than the work (between 68 and 2045) reported by Eisenstat et al. [25] when various gradient-type methods are applied to an $O(h^2)$ variant of the upwind approximation to equation (20), due to Axelsson and Gustafsson [26].

For $\varepsilon = 1/100$ and $h = 1/8, 1/16$, and $1/32$, the number of iterations to reduce the error by a factor of $10^6$ is 2, 3 and about 11, respectively, so that the work per unknown is about 15, 22 and 78, respectively. The ranges reported by Eisenstat et al. [25] are 96–345, 143–640 and 220–1136, respectively.
Higher accuracy

We obtained higher accuracy by replacing the right-hand side of equation (23) with

$$\beta_{-1} F_{i-1,j} + \beta_0 F_{i,j} + \beta_1 F_{i+1,j},$$

where

$$\beta_{-1} = \frac{\theta - 2 + (\theta + 2)e^{-\theta}}{2\theta(1 - e^{-\theta})^2}, \quad \beta_1 = \beta_{-1} e^{-\theta}, \quad \beta_0 = 1 - \beta_{-1} - \beta_1, \quad \theta = v/h/e.$$  \hspace{1cm} (30)

The coefficients were determined so that when \( v(x) \equiv \text{constant} \), the scheme is exact (see Section 4) when

$$u(x, y) = A + Bx + Cx^2 + (D + Ex) e^{\beta x} + Fy + Gy^2;$$  \hspace{1cm} (31)

it is a result of the symmetry of the operator that the scheme is also exact when \( Hx^2 \exp(vx/e) + Iy^3 \) is added to the right-hand side of equation (31). As in the one-dimensional case, the local Green's function is strongly skewed in the upstream direction; see Gartland [27, 28].

Experimental results are given below for the solution of equation (20) on the unit square, when \( F \) and the boundary conditions are taken so that the solution is

$$u(x, y) = e^{x} + (x + 1) [1 + \sin(y(1 - y))].$$  \hspace{1cm} (32)

For \( \epsilon = 0.001 \) and for the mesh spacings we used, the iterative method described above converged to the level of the discretization error in one iteration; namely to

0.00356, 0.0000909, 0.0000199, 0.00000308,

for \( 1/h = 8, 16, 32, 64 \), respectively.

Channel-edge boundary layers

When boundary layers of thickness \( O(\epsilon^{1/2}) \) are present at the edges \( (y = 0 \text{ and } Y = b) \), one can replace the divided central difference approximation to \( \epsilon u_{yy} \) in equation (23) with one which is exact on the set of basis elements 1, \( \exp(ay^L - \frac{1}{2}) \) and \( \exp(-ay^L - \frac{1}{2}) \):

$$-\epsilon u_{yy} \approx a^2 [-u_{j-1} + 2u_{j} - u_{j+1}] / [(e^\kappa - 2 + e^{-\kappa})], \quad \kappa = ah/\epsilon^{1/2},$$

where \( a \) can be made to change with \( x \) when the width of the boundary layer changes with \( x \). The line Gauss-Seidel method of solving the resulting system of difference equations converges even faster than for equation (23) because the matrix \( C \) is even more strongly diagonally dominant for this approximation. We have also made some preliminary experiments which indicate that similar methods are highly efficient and accurate for this case. We plan to report on this approximation and approximations near corners with Bessel and parabolic cylindrical functions, as well as more accurate 9-point difference approximations, in a subsequent paper.

We have recently extended the results reported here, after noticing that stable discretizations of convection–diffusion equations are naturally associated with diagonally dominant \( M \)-matrices. Additional numerical experiments, concerned with solving extensions of equation (23) to general uniform rectangular meshes on rectangular domains, show that downstream line Gauss–Seidel iteration converges in relatively few steps for convection-dominated problems, but that downstream line SOR with optimum \( \omega \) converges even more rapidly. These new results will be reported in a paper to appear in Numerische Mathematik.

REFERENCES

APPENDIX

We determine the spectral radius of the line Gauss–Seidel scheme (27) in the case of constant coefficients. The iteration error $E^{n+1} = U_n - U^{n+1}$ satisfies

$$-RE^{n+1} + CE^{n+1} - SE^n = 0, \quad R = RI, \quad S = sI,$$

(A.1)

with zero values on the boundary. The symmetric tridiagonal $(J-1) \times (J-1)$ matrix $C$, given in equation (26), has a complete orthogonal set of eigenvectors $\varphi_p$, $p = 1, \ldots, J-1$, with components

$$(\varphi_p)_j = \sin(jpn/J), \quad j = 1, \ldots, J-1$$

and corresponding eigenvalues

$$2\zeta_p = \frac{2\epsilon}{\epsilon^2}(1 - \cos(p\pi/J)) + r + s, \quad p = 1, \ldots, J-1.$$

We express the errors in terms of these eigenvectors:

$$E^{n+1} = \sum \beta^{(n+1)}_p \varphi_p.$$

Substitution into equation (A.1) and use of the orthogonality yields

$$-r\beta^{(n+1)}_1 + \zeta_p\beta^{(n+1)}_p - s\beta^{(n+1)}_{p+1} = 0,$$

(A.2)

This gives a linear transformation $M_1 \beta^{(n+1)} = N_1 \beta^{(n)}$ which takes $\beta^{(n)}_p = (\beta_{1,p}, \ldots, \beta_{J-1,p})^T$ to $\beta^{(n+1)}_p$, where the $(J-1) \times (J-1)$ matrices $M$ and $N$ have zero entries, except for

$$M_{i,i} = \zeta_p, \quad M_{i,i-1} = -r, \quad N_{i,i+1} = -s.$$

Since $\beta^{(n+1)} = M^{-1}N\beta^{(n)}$, the spectral radius of the scheme is the spectral radius of the iteration matrix $G = M^{-1}N$. Thus we solve the eigenproblem $G\lambda = \lambda\varphi$ or, equivalently, $M\lambda = s\lambda$. In component form this is

$$-\lambda r\varphi_{i-1} + \zeta_p\varphi_i - s\varphi_{i+1} = 0.$$
This difference equation is made symmetric by the change \( \gamma_i = (\lambda r/s)^{1/2} \delta_i \), which gives

\[-\delta_{i-1} + \lambda^{1/2} (rs)^{-1/2} \bar{\xi}_i \delta_i - \delta_{i+1} = 0. \] (A.3)

Since \( \delta_0 = \delta_L = 0 \), it is evident that \( \delta_i = \sin(qn/L) \) is a solution of equation (A.3) for \( q = 1, \ldots, L-1 \). Substitution and simplification then gives

\[-2\cos(qn/L) + \lambda^{1/2} (rs)^{-1/2} \bar{\xi}_i = 0,\]

hence

\[
\lambda = \frac{4rs \cos^2(qn/L)}{\bar{\xi}_i^2} = \frac{4rs \cos^2(qn/L)}{[2e(1 - \cos(pn/J))/k^2 + r + s]^2},
\]

where

\[
r = \frac{v}{h(1 - e^{-i})}, \quad s = \frac{v}{h(e^{-i} - 1)}, \quad z = \frac{hv}{c}.
\]

Consequently, the spectral radius of the line Gauss–Seidel method is

\[
\rho = \frac{4rs \cos^2(qn/L)}{[2e(1 - \cos(pn/J))/k^2 + r + s]^2} \leq \frac{4rs}{(r + s)^2} = \frac{4}{e^i + 2 + e^{-i}}.
\]