Numerical Analysis Project
Manuscript NA-90-01

February 1990

Line Iterative Methods for Cyclically Reduced Discrete Convection-Diffusion Problems

by

Howard C. Elman
Gene H. Golub

Numerical Analysis Project
Computer Science Department
Stanford University
Stanford, California 94305

UMIACS-TR-90-16 February 1990
CS-TR-2403

Line Iterative Methods for Cyclically Reduced Discrete Convection-Diffusion Problems

Howard C. Elman[1]
Department of Computer Science
and Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
Gene H. Golub[2]
Department of Computer Science
Stanford University
Stanford, CA 94305

Abstract

We perform an analytic and empirical study of line iterative methods for solving the discrete convection-diffusion equation. The methodology consists of performing one step of the cyclic reduction method, followed by iteration on the resulting reduced system using line orderings of the reduced grid. Two classes of iterative methods are considered: block stationary methods, such as the block Gauss-Seidel and SOR methods, and preconditioned generalized minimum residual methods with incomplete LU preconditioners. New analysis extends convergence bounds for constant coefficient problems to problems with separable variable coefficients. In addition, analytic results show that iterative methods based on incomplete LU preconditioners have faster convergence rates than block Jacobi relaxation methods. Numerical experiments examine additional properties of the two classes of methods, including the effects of direction of flow, discretization, and grid ordering on performance.

Abbreviated Title. Line Iterative Methods for Convection-Diffusion Problems.

Key words. Iterative methods, line orderings, reduced system, convection-diffusion, elliptic operators.

AMS(MOS) subject classification. Primary: 65F10, 65N20. Secondary: 15A06.

[1]The work of this author was supported by the National Science Foundation under grants DMS-8607478 and ASC-8958544, and by the U. S. Army Research Office under grant DAAL-0389-K-0016

[2]The work of this author was supported by the National Science Foundation under grant DCR-8412314.

1. Introduction.

Consider the convection-diffusion equation

$$-[(pu_x)_x + (qu_y)_y] + ru_x + su_y = f \text{ on } \Omega \tag{1.1a}$$

$$\alpha u + \beta u_n = g \text{ on } \partial\Omega, \tag{1.1b}$$

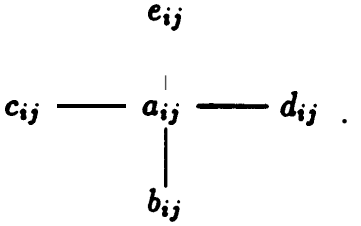
where Ω is a smooth domain in $\mathbf{R}^2$ and $p > 0$, $q > 0$ on Ω. Discretization of (1.1) produces a linear system of equations

$$Au = f, \tag{1.2}$$

where u and f are now vectors in a finite dimensional space, and A is a nonsymmetric matrix when r and s are nonzero. We are concerned with discretizations (principally, finite difference methods) for which each equation in (1.2) is centered at some mesh point (x_i, y_j), and the associated unknown u_{ij} depends only on its neighbors in the horizontal and vertical directions. That is, the equation centered at (x_i, y_j) has the form

$$a_{ij}u_{ij} = f_{ij} - b_{ij}u_{i,j-1} - c_{ij}u_{i-1,j} - d_{ij}u_{i+1,j} - e_{ij}u_{i,j+1}. \tag{1.3}$$

In this case, we say that (1.2) has a ***computational molecule*** of the form



When the system (1.2) has this property, the mesh points $\{(x_i, y_j)\}$ and unknowns $\{u_{ij}\}$ can be ordered with a red-black ordering so that every equation centered at a "red" point depends only on "black" unknowns, and every equation centered at a "black" point depends only on "red" unknowns. An example of a red-black ordering of a 6 x 5 grid is shown in Fig. 1.1. If u_{ij} is a black unknown, then by adding appropriate linear combinations of the equations for $u_{i\pm1,j}$ and $u_{i,j\pm1}$ to the equation for u_{ij}, we can eliminate the dependence of u_{ij} on its red neighbors. When this is done for every black equation, the result is a smaller linear system

$$A^{(b)}u^{(b)} = g^{(b)}, \tag{1.4}$$

where $u^{(b)}$ is the set of unknowns associated with black mesh points.[1]

[1] In matrix notation, the rows and columns of A can be ordered so that (1.2) has the form

$$\begin{pmatrix} D & C \\ E & F \end{pmatrix} \begin{pmatrix} u^{(r)} \\ u^{(b)} \end{pmatrix} = \begin{pmatrix} f^{(r)} \\ f^{(b)} \end{pmatrix}$$

where D and F are nonsingular diagonal matrices. Matrices of this type are said to possess *Property A* [26], or to be *two-cyclic* [23]. Decoupling of the red points $u^{(r)}$ is equivalent to producing the system (1.4), where $A^{(b)} = F - ED^{-1}C$ and $g^{(b)} = f^{(b)} - ED^{-1}f^{(r)}$.

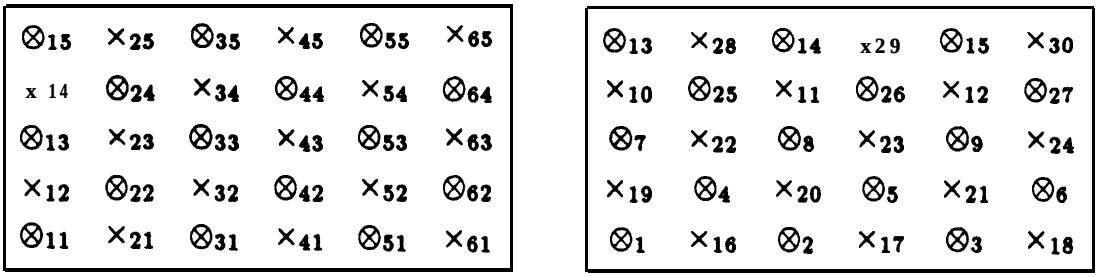


Fig. 1.1: A 6 × 5 grid and a red-black ordering. Grid indices are shown on the left, and vector indices for a red-black ordering are shown on the right. Red points are denoted by "⊗" and black points by "×."

In [7], [8], we analyzed the convergence behavior of block iterative methods for solving the reduced system (1.4) derived from discretizations of (1.1). We considered block Jacobi, Gauss-Seidel and successive over-relaxation (SOR) methods [23], [26], where the blockings (of the rows and columns of $A^{(b)}$) are derived from certain *line orderings* of the underlying reduced (black) grid. In particular, the unknown grid values $u^{(b)}$ can be grouped together either by individual lines of the grid, producing a class of *one-line* orderings, or by pairs of lines, producing *two-line* orderings (see §2). These orderings produce matrices with block Property A, so that the classical analysis of Gauss-Seidel and SOR methods [23], [26] can be used. The results of [7], [8] apply to problems with the constant coefficients $p(x, y) = q(x, y) = 1$, $r(x, y) = \sigma$, $s(x, y) = \tau$. They show that convergence is often very fast; in particular, for non-self-adjoint problems (σ or τ nonzero), convergence is typically faster than for self-adjoint problems. They also show that convergence rates for solving the reduced system are often faster than for solving the full system (1.2) by analogous line methods. These observations are in agreement with asymptotic results in [18] and the algebraic analysis of [11]. Related results for point iterative methods are given in [16].

In this paper, we extend the analysis of [7], [8] to separable problems, and we also use it to derive bounds on convergence behavior for stationary methods based on incomplete factorizations [15]. In addition, in a series of numerical experiments, we examine the effect of physically significant properties of the problem (1.1) on the performance of iterative methods applied to (1.4). Here, we consider both block relaxation methods and the preconditioned generalized minimum residual method (GMRES) [21], with preconditioning by incomplete factorizations [15]. We focus on the following issues:

1. For constant coefficient problems, the effect of the signs and magnitudes of r and s in (1.1). These quantities determine the direction and rate of flow associated with the convection in the model. The analysis of [7], [8] is sensitive to magnitudes but not to signs.
2. The effect of variable coefficients r and s. We consider problems both with and without turning points.
3. The effects of the choice of discretization on performance; we consider centered and upwind finite difference discretizations.
4. The first three issues do not address the issue of accuracy of the discrete solution. We also examine the effect of methods designed to improve accuracy in the presence of boundary layers, in particular, local mesh refinement and defect correction methods [10], [13].

An outline of the paper is as follows. In §2, we describe the reduced matrix $A^{(b)}$, and we present the ordering strategies and iterative methods used to solve (1.4), including some block red-black strategies of use for vector and parallel computations. In §3, we extend the analysis of [7], [8] to separable problems and incomplete factorizations. In §4, we describe the results of numerical experiments with constant coefficient problems. For several ordering strategies, we examine how performances of block stationary methods and preconditioned GMRES are affected by direction and rate of flow, choice of difference scheme, and use of local mesh refinement to resolve boundary layers. In §5, we compare experimental results with analytic bounds on convergence, for separable problems. In §6, we consider performance for some problems with nonseparable variable coefficients, i.e. where the flow varies in both direction and magnitude in Ω. Here we consider both centered and upwind finite differences, as well as a difference scheme used to implement defect correction methods. Finally, in §7 we make some concluding remarks.

2. The Reduced System and Line Iterative Methods.

Consider the two equations from (1.2) centered at the (x_i, y_j) (as in (1.3)) and (x_i, y_{j-1}) mesh points:

$$a_{ij}u_{ij} + b_{ij}u_{i,j-1} + c_{ij}u_{i-1,j} + d_{ij}u_{i+1,j} + e_{ij}u_{i,j+1} = f_{ij},$$
$$a_{i,j-1}u_{i,j-1} + b_{i,j-1}u_{i,j-2} + c_{i,j-1}u_{i-1,j-1} + d_{i,j-1}u_{i+1,j-1} + e_{i,j-1}u_{ij} = f_{i,j-1}.$$

Solving the second equation for $u_{i,j-1}$ and then substituting into the first equation gives the new equation

$$\left[a_{ij} - \frac{b_{ij}e_{i,j-1}}{a_{i,j-1}}\right] u_{ij} + c_{ij}u_{i-1,j} + d_{ij}u_{i+1,j} + e_{ij}u_{i,j+1}$$
$$\frac{b_{ij}b_{i,j-1}}{a_{i,\,j-1}}u_{i,j-2} - \frac{b_{ij}c_{i,j-1}}{a_{i,\,j-1}}u_{i-1,j-1} - \frac{b_{ij}d_{i,j-1}}{a_{i,j-1}}u_{i+1,j-1} = f_{ij} - \frac{b_{ij}f_{i,j-1}}{a_{i,j-1}}.$$

Unknowns $u_{i-1,j}$, $u_{i+1,j}$ and $u_{i,j+1}$ are eliminated in a similar manner, using this equation and the ones centered at the other neighbors of (x_i, y_j). Thus, for all black mesh points not next to the boundary $\partial\Omega$, the computational molecule for the reduced matrix $A^{(b)}$ has the form shown in Fig. 2.1. The value "*" in the center is

$$a_{ij} - \frac{b_{ij}e_{i,j-1}}{a_{i,\,j-1}} - \frac{c_{ij}d_{i-1,j}}{a_{i-1,j}} \quad \frac{d_{ij}c_{i+1,j}}{a_{i+1,j}} \quad \frac{e_{ij}b_{i,j+1}}{a_{i,\,j+1}},$$

and the right hand side is perturbed by an average of neighboring values,

$$g_{ij}^{(b)} = f_{ij} - \frac{b_{ij}f_{i,j-1}}{a_{i,j-1}} - \frac{c_{ij}f_{i-1,j}}{a_{i-1,j}} \quad \frac{d_{ij}f_{i+1,j}}{a_{i+1,j}} - \frac{e_{ij}}{a_{i,j+1}}\,f_{i,\,j+1}.$$

We will be concerned with finite difference discretizations of (1.1). On a uniform grid with mesh size h, let standard second order differences [9] be used for the second derivative

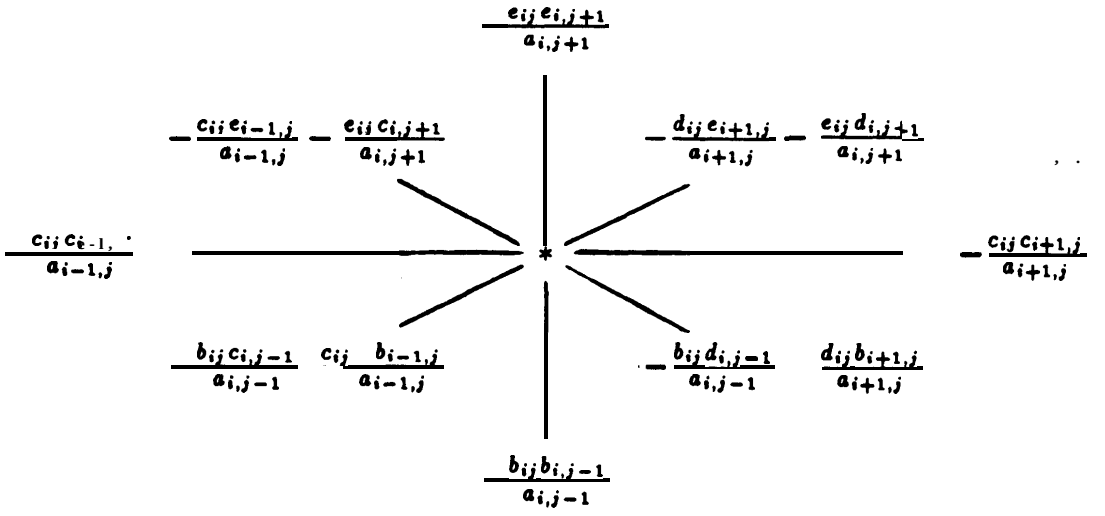


Fig. 2.1. The computational molecule for the reduced system.

terms. If centered differences are used for the first derivative terms, then after scaling by h^2, the values in the computational molecule are given by

$$a_{ij} = p(x_{i+1/2}, y_j) + p(x_{i-1/2}, y_j) + q(x_i, y_{j+1/2}) + q(x_i, y_{j-1/2}),$$
$$b_{ij} = -(q(x_i, y_{j-1/2}) + s(x_i, y_j)h/2), \qquad d_{ij} = -(p(x_{i+1/2}, y_j) - r(x_i, y_j)h/2),$$
$$c_{ij} = -(p(x_{i-1/2}, y_j) + r(x_i, y_j)h/2), \qquad e_{ij} = -(q(x_i, y_{j+1/2}) - s(x_i, y_j)h/2).$$

If upwind differencing is used for the fist derivatives, then (for the case $r(x_i, y_j) > 0$, $s(x_i, y_j) > 0$), the values are

$$a_{ij} = p(x_{i+1/2}, y_j) + p(x_{i-1/2}, y_j) + q(x_i, y_{j+1/2}) + q(x_i, y_{j-1/2}) + r(x_i, y_j)h + s(x_i, y_j)h,$$
$$b_{ij} = -(q(x_i, y_{j-1/2}) + s(x_i, y_j)h), \qquad d_{ij} = -p(x_{i+1/2}, y_j),$$
$$c_{ij} = -(p(x_{i-1/2}, y_j) + r(x_i, y_j)h), \qquad e_{ij} = -q(x_i, y_{j+1/2}).$$

If instead, $s(x_i, y_j) < 0$, then $b_{ij} = -q(x_i, y_{j-1/2})$, $e_{ij} = -(q(x_i, y_{j+1/2}) - s(x_i, y_j)h)$, and $s(x_i, y_j)h$ is replaced by $-s(x_i, y_j)h$ in the expression for a_{ij}. The case $r(x_i, y_j) < 0$ is handled in an analogous manner.

The line ordering strategies for the reduced grid are outlined as follows, see [7], [8] for further details. *In* the ***natural one-line*** ordering, points of the reduced grid are grouped together by diagonal lines, e.g. oriented in the NW-SE direction. The left side of Fig. 2.2 shows an example for a 6 x 5 grid. Here, the E'th line consists of all points with grid indices (i, j) such that $i + j = 2k+1$. (Compare with the left side of Fig. 1.1) Thus, in Fig. 2.2, the **first line** consists of the points $\{1,2\}$, the second line consists of the points $\{3,4,5,6\}$, etc. In the ***natural two-line*** ordering, points are grouped together by pairs of either horizontal or vertical lines. The right side of Fig. 2.2 shows an example of a horizontal grouping

for a 6 x 5 grid. **The** points in the k'th group are those with grid indices (i, j) such that $k - 1 < j/2 \leq k$. If the number of lines is odd, the last group consists of a single line, as in the group $\{13, 14, 15\}$. For both these strategies, $A^{(b)}$ is a block tridiagonal matrix; let D denote its block diagonal. For the one-line ordering, each block of D is a tridiagonal matrix, and for the two-line ordering, each block of D is a pentadiagonal matrix (except possibly the last block, which may be tridiagonal). It is also useful (e.g. for parallel computations, *see* [8]) to define ***line red-black*** variants of these orderings, in which alternating lines (or line-pairs) are assigned opposite colors. For example, for the one-line version, let the sets $\{1, 2\}, \{7, 8, 9, 10, 11\}$ and $\{15\}$ be denoted **as** "red" **lines,** and the others as "black" lines. Then every equation centered at **a** point in a red line depends only on that red line and the neighboring black lines; an analogous statement holds for equations centered on black lines. For the red-black one-line ordering, all red lines are ordered first, followed by all black lines. The red-black two-line ordering is defined in similar fashion.

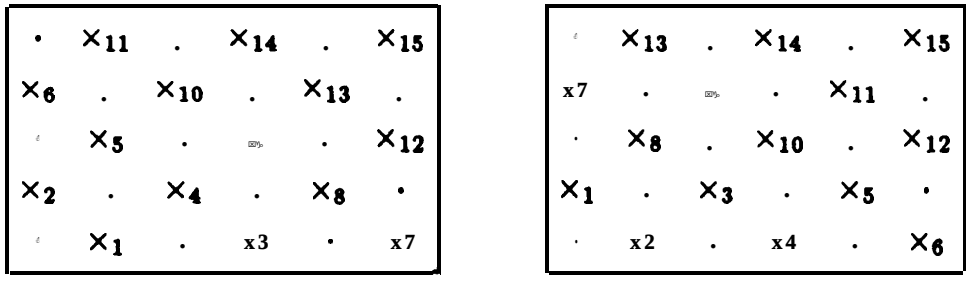


Fig. 2.2: Natural one-line (left) and two-line (right) orderings of the reduced 6×5 grid.

For any of these line orderings, let

$$A^{(b)} = D - C = (D - L) - U, \tag{2.1}$$

where D is the block diagonal part of $A^{(b)}$ and L and U are the lower and upper triangular parts, respectively, of the block off-diagonal part of $A^{(b)}$. We consider several block stationary methods based on the splittings (2.1). The block Jacobi iteration is given by

$$u_{k+1}^{(b)} = D^{-1}Cu_k^{(b)} + D^{-1}g^{(b)},$$

and the block SOR iteration is

$$u_{k+1}^{(b)} = (D - \omega L)^{-1}[(1 - \omega)D + \omega U]u_k^{(b)} + \omega(D - \omega L)^{-1}g^{(b)}. \tag{2.2}$$

The block **Gauss-Seidel iteration corresponds to the case w =** 1 in (2.2). In all cases, $A^{(b)}$ has block **Property A, so that** [26]

$$\rho((D - L)^{-1}U) = [\rho(D^{-1}C)]^2, \tag{2.3}$$

where p(X) denotes **the** spectral radius of **a** matrix X.

In addition, we consider the use of the IC(0) incomplete factorization [15] applied to $A^{(b)}$ for each of the orderings. This factorization is defined as

$$M = (\hat{D} - \hat{L})\hat{D}^{-1}(\hat{D} - \hat{U}), \tag{2.4}$$

where $\hat{D}$ is a diagonal matrix; $\hat{L}$ and $\hat{U}$ are strictly lower triangular and upper triangular, respectively; the nonzero structure of $\hat{D} - \hat{L} - 6$ is the same as that of $A^{(b)}$; and the entries of M are the same as the corresponding entries of $A^{(b)}$ wherever the latter are nonzero. We will examine the use of this factorization as a preconditioner for GMRES.

3. Analysis of separable problems and the IC(0) factorization.

If Ω is a rectangular domain and the coefficients of (1.1a) satisfy

$$P = p(x), \quad q = q(y), \quad r = f(3)' \quad s = s(y),$$

then the differential operator of (1.1) is *separable* [24]. In this case, the discrete coefficients of (1.3) satisfy

$$\begin{aligned} a_{ij} &= a_i^{(x)} + a_j^{(y)} \\ b_{ij} = b_j,\ c_{ij} &= c_i,\ d_{ij} = d_i,\ e_{ij} = e_j. \end{aligned} \tag{3.1}$$

Our convergence analysis is based on symmetrizing the reduced matrix $A^{(b)}$ by a diagonal similarity transformation. The following result gives circumstances under which $A^{(b)}$ can be symmetrized when it comes from a separable operator. In the analysis, matrix entries are referenced using indices from the underlying reduced grid. That is, every nonzero entry of the row of $A^{(b)}$ associated with the (i, j) grid point is referenced using subscripts i and j. For example, the entry corresponding to the point southwest of the center of the computational molecule (see Fig. 2.1) is denoted by

$$-b_j c_i \left(\frac{1}{a_{i,j-1}} + \frac{1}{a_{i-1,j}} \right),$$

where the numerator is expressed using the notation of (3.1).

THEOREM 1. *If the operator of* (1.1) *is separable and* $c_i d_{i-1}$ *and* $b_j e_{j-1}$ *have the same sign for all i and j, then the reduced matrix* $A^{(b)}$ *can be symmetrized with a real* diagonal *similarity transformation.*

Proof, We seek a diagonal matrix Q such that $Q^{-1}A^{(b)}Q$ is symmetric. Let $A^{(b)}$ be ordered by the natural one-line ordering, so that its rows and columns are grouped into l blocks corresponding to l individual lines. Let Q be ordered the same way.

First consider the block diagonal D, which is a tridiagonal matrix. Any two successive rows of a block of D, corresponding to the (i, j) and (i − 1, j + 1) mesh points, contain the 2 x 2 sub-block

$$\begin{pmatrix} * & -c_i e_j \left(\frac{1}{a_{i-1,j}} + \frac{1}{a_{i,j+1}} \right) \\ -b_{j+1} d_{i-1} \left(\frac{1}{a_{i-1,j}} + \frac{1}{a_{i,j+1}} \right) & * \end{pmatrix},$$

where "*" denotes a diagonal entry. If q_{ij} is known, then $q_{i-1,j+1}$ must be chosen so that

$$q_{i-1,j+1}^{-1} b_{j+1} d_{i-1} \left(\frac{1}{a_{i-1,j}} + \frac{1}{a_{i,j+1}} \right) q_{ij} = q_{ij}^{-1} c_i e_j \left(\frac{1}{a_{i-1,j}} + \frac{1}{a_{i,j+1}} \right) q_{i-1,j+1}.$$

Thus, within the blocks of Q, successive entries must satisfy

$$q_{i-1,j+1} = \left(\frac{b_{j+1} d_{i-1}}{c_i e_j} \right)^{1/2} q_{ij}. \tag{3.2}$$

For symmetrizing D, the first entry of each block of Q may be arbitrary.

To symmetrize the off-diagonal blocks of $A^{(b)}$, we require

$$Q_k^{-1} A_{k,k-1}^{(b)} Q_{k-1} = (Q_{k-1}^{-1} A_{k-1,k}^{(b)} Q_k)^T, \tag{3.3}$$

where k is a block (or line) index, $2 \le k \le l$. There are three cases, corresponding to $2 \le k < l/2 + 1$, $k = l/2 + 1$ (l even) and $l/2 + 2 < k$. In the case $2 \le k \le l$, a careful specification of the entries of Q and $A^{(b)}$ shows that (3.3) is equivalent to the following three scalar relations:

$$q_{ij} = \left(\frac{c_i c_{i-1}}{d_{i-1} d_{i-2}} \right)^{1/2} q_{i-2,j}, \tag{3.4}$$

$$q_{i-1,j+1} = \left(\frac{b_{j+1} c_{i-1}}{d_{i-2} e_j} \right)^{1/2} q_{i-2,j}, \tag{3.5}$$

$$q_{i-2,j+2} = \left(\frac{b_{j+1} b_{j+2}}{e_j e_{j+1}} \right)^{1/2} q_{i-2,j}. \tag{3.6}$$

These relations specify three successive entries of Q_k in terms of a single entry of Q_{k-1} (where $k = (i + j - 1)/2$). Since the first entry of Q_k is arbitrary, (3.4) can be used to define it. However, once this entry is defined, all subsequent entries are determined by (3.2). Thus, it is necessary to show that (3.4) – (3.6) are consistent with (3.2). But application of (3.2) and (3.4) in either order results in (3.5), showing that both (3.4) and (3.5) are consistent with (3.2). Similarly, (3.6) follows directly from (3.2) and (3.5).

The arguments for the cases $k = l/2 + 1$ (l even) and $l/2 + 2 < k$ are essentially the same and we omit the details. A sufficient condition to guarantee that all the required square roots are well-defined is that $c_i d_{i-1}$ and $b_j e_{j-1}$ have the same sign for all i and j.

Finally, note that this analysis is not restricted to the natural one-line ordering: If $A^{(b)}$ is symmetrically permuted into some other order, giving the permuted matrix $\tilde{A}^{(b)}$, then for an analogous permutation of Q to $\tilde{Q}$, $\tilde{Q}^{-1} \tilde{A}^{(b)} \tilde{Q}$ is also symmetric. □

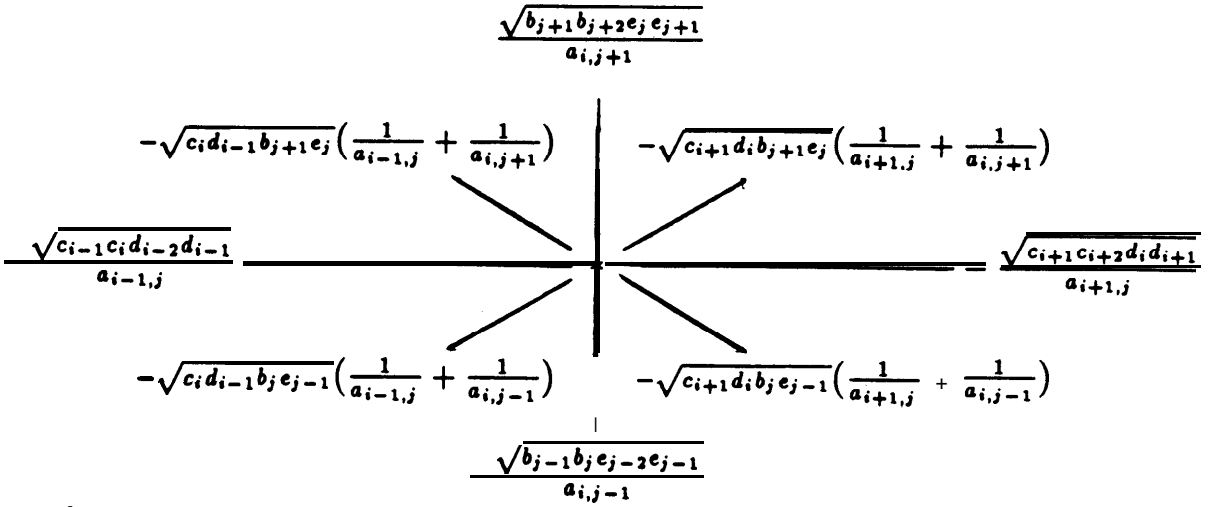


Fig. 3.1. **The computational molecule for the symmetrized reduced system in the separable case.**

REMARK 1. For the centered difference discretization, necessary and sufficient conditions to ensure that all $c_i d_{i-1}$ and $b_j e_{j-1}$ have the same sign are that either

(3.7)
$$\max_i \left[\max \left(\left| \frac{r(x_i)h}{2p(x_{i-1/2})} \right|, \left| \frac{r(x_{i-1})h}{2p(x_{i-1/2})} \right| \right) \right] < 1 \text{ and } \max_j \left[\max \left(\left| \frac{s(y_j)h}{2q(y_{j-1/2})} \right|, \left| \frac{s(y_{j-1})h}{2q(y_{j-1/2})} \right| \right) \right] < 1;$$

or

$$\min_i \left[\min \left(\left| \frac{r(x_i)h}{2p(x_{i-1/2})} \right|, \left| \frac{r(x_{i-1})h}{2p(x_{i-1/2})} \right| \right) \right] > 1 \text{ and } \min_j \left[\min \left(\left| \frac{s(y_j)h}{2q(y_{j-1/2})} \right|, \left| \frac{s(y_{j-1})h}{2q(y_{j-1/2})} \right| \right) \right] > 1.$$

In contrast, the full system (1.2) can be symmetrized by a diagonal similarity transformation if and only if the conditions (3.7) hold. For upwind differences, it is always the case that $c_i d_{i-1} > 0$ and $b_j e_{j-1} > 0$ for all i, j.

Let $\hat{A}^{(b)} = Q^{-1} A^{(b)} Q$ denote the symmetrized reduced matrix, when it exists, for any of the strategies under consideration. Fig. 3.1 shows the resulting computational molecule. Let

$$\hat{A}^{(b)} = \hat{D} - \hat{C}$$

denote the block Jacobi splitting, where $\hat{D} = Q^{-1} D Q$, $\hat{C} = Q^{-1} C Q$. Note that $\hat{D}^{-1}\hat{C} = Q^{-1} D^{-1} C Q$, so that the eigenvalues of $D^{-1}C$ are the same as those of $\hat{D}^{-1}\hat{C}$, and in particular they are real. Let $\mathcal{L}_\omega = (D - \omega L)^{-1}[(1 - \omega)D + \omega U]$ denote the block SOR iteration matrix. The following result is then a straightforward application of the analysis of the block SOR method [26].

COROLLARY 1. *If $A^{(b)}$ is the reduced matrix derived from a separable operator, and $c_i d_{i-1}$ and $b_j e_{j-1}$ have the same* sign *for all* i *and j, then $\rho(D^{-1}C) = \rho(\hat{D}^{-1}\hat{C})$. If $\rho(D^{-1}C) < 1$, then $\rho(\mathcal{L}_{\omega^*}) = \omega^* - 1$, where $\omega^* = 2/\left(1 + \sqrt{1 + [\rho(D^{-1}C)]^2}\right)$* minimizes $\rho(\mathcal{L}_\omega)$.

REMARK 2. It may be possible to establish the requirements of Corollary 1 a priori. Sufficient conditions to guarantee that $\rho(D^{-1}C) < 1$ are that the original matrix A be a diagonally dominant M-matrix, which is always the case for upwind differences, and

is also true for centered differences for small enough h.[2] Moreover, even if Corollary I cannot be invoked from an a priori examination of matrix entries, it may still be useful as a guideline **for** practical computation. For example, for constant coefficient problems, empirical evidence and Fourier analysis suggest that $\rho(D^{-1}C) < 1$ in cases where $c_i d_{i-1}$ and $b_j e_{j-1}$ *axe* both negative but A is not a diagonally dominant M-matrix. **A good** value for the SOR parameter could be computed from a dynamic estimation of $\rho(D^{-1}C)$, *e.g.* using the methods of [12], §9. In addition, note that it is not necessary to compute Q or $\hat{A}^{(b)}$ in order to apply this result, see [7].

COROLLARY *2. Let $A^{(b)}$ come from a separable operator discretized on a uniform square grid of mesh width h, and assume that*

$$a_i^{(x)} \geq \alpha^{(x)}, \quad a_j^{(y)} \geq \alpha^{(y)}, \quad 0 < c_{i+1}d_i \leq \xi, \quad 0 < b_{j+1}d_j \leq \eta, \tag{3.8}$$

for all i, j. If $A^{(b)} = D - C$ is a **one-line** *Jacobi splitting and*

$$\alpha^{(x)} + \alpha^{(y)} \geq \sqrt{2}(\sqrt{\xi} + \sqrt{\eta}), \tag{3.9}$$

then

$$\rho(D^{-1}C) \leq \frac{2(\sqrt{\xi} + \sqrt{\eta})^2}{(\alpha^{(x)} + \alpha^{(y)})^2 - 2(\sqrt{\xi} + \sqrt{\eta})^2 + 4\sqrt{\xi\eta}\,(1 - \cos \pi h)}. \tag{3.10}$$

If $A^{(b)} = D - C$ is a two-line Jacobi **splitting** *and*

$$(\alpha^{(x)} + \alpha^{(y)})^2 \geq 2(\sqrt{\xi} + \sqrt{\eta})^2 + 2\xi, \tag{3.11}$$

then

(3.12)

$$\rho(D^{-1}C) \leq \frac{2\eta \cos 2\pi h + 4\sqrt{\xi\eta}\cos \pi h}{(\alpha^{(x)} + \alpha^{(y)})^2 - 2(\sqrt{\xi} + \sqrt{\eta})^2 - 2\xi + 4\sqrt{\xi\eta}\,(1 - \cos \pi h) + 4\xi\,(1 - \cos^2 \pi h)} + o(h^2).$$

Proof. **Using Corollary** 1, we have (for any ordering)

$$\rho(D^{-1}C) = \rho(\hat{D}^{-1}\hat{C}) \leq \|\hat{D}^{-1}\|_2 \|\hat{C}\|_2 = \rho(\hat{D})\rho(\hat{C}).$$

Consider the **one-line orderings. By (3.8), all nonzero** off-diagonal entries of $\hat{D}$ are bounded below by $-2\sqrt{\xi\eta}/(\alpha^{(x)} + \alpha^{(y)})$, **and all diagonal entries of D are bounded below by**

$$\alpha^{(x)} + \alpha^{(y)} - 2\xi/(\alpha^{(x)} + \alpha^{(y)}) - 2\eta/(\alpha^{(x)} + \alpha^{(y)}).$$

[2] A **nonsingular matrix X is an M-matrix if $X_{ij} \leq 0$ for $i \neq j$ and $X^{-1} \geq 0$.**

Thus, $\hat{D} \geq \tilde{D}$, where each block of $\tilde{D}$ is a constant coefficient tridiagonal matrix

$$(3.13) \qquad tri\ [\ -\frac{2\sqrt{\xi\eta}}{\alpha^{(x)}+\alpha^{(y)}},\ \alpha^{(x)}+\alpha^{(y)}-\frac{2\xi}{\alpha^{(x)}+\alpha^{(y)}}-\frac{2\eta}{\alpha^{(x)}+\alpha^{(y)}},\ -\frac{2\sqrt{\xi\eta}}{\alpha^{(x)}+\alpha^{(y)}}\ ].$$

The size of this block depends on the line from which it is derived. Assumption (3.9) implies that each block (3.13) and, therefore, each corresponding block of $\hat{D}$, is an irreducibly diagonally dominant M-matrix. Hence, the Perron-Frobenius theory implies $\rho(\hat{D}^{-1}) \leq \rho(\tilde{D}^{-1})$. Similarly, by (3.8), $0 \leq \hat{C} \leq \tilde{C}$, where $\tilde{C}$ is a matrix with the same nonzero structure as that of $\hat{C}$ in which all occurences of $c_i d_{i-1}$, $b_j e_{j-1}$, and a_{ij} are replaced by ξ, η, and $\alpha^{(x)} + \alpha^{(y)}$, respectively. Consequently, $\rho(\hat{C}) \leq \rho(\tilde{C})$, and we have

$$(3.14) \qquad \rho(\hat{D}^{-1})\rho(\hat{C}) \leq \rho(\tilde{D}^{-1})\rho(\tilde{C}),$$

where. the right side of the inequality contains constant coefficient matrices. The bound (3.10) is determined from the maximum eigenvalue of $\tilde{D}^{-1}$ and use of Gerschgorin's theorem for $\tilde{C}$. (See [7], Theorem 4.)

For the two-line ordering, the blocks of D and $\hat{D}$ are pentadiagonal matrices, and $\hat{D} \geq \tilde{D}$ where each block of D is a constant coefficient pentadiagonal matrix,

$$penta\ [\ -\frac{\xi}{\alpha^{(x)}+\alpha^{(y)}},\ -\frac{2\sqrt{\xi\eta}}{\alpha^{(x)}+\alpha^{(y)}},\ \alpha^{(x)}+\alpha^{(y)}-\frac{2\xi}{\alpha^{(x)}+\alpha^{(y)}}-\frac{2\eta}{\alpha^{(x)}+\alpha^{(y)}},\ -\frac{2\sqrt{\xi\eta}}{\alpha^{(x)}+\alpha^{(y)}},\ -\frac{\xi}{\alpha^{(x)}+\alpha^{(y)}}\ ],$$

which is assumed in (3.11) to be diagonally dominant. In addition, exactly as above, $0 \leq \hat{C} \leq \tilde{C}$ where $\tilde{C}$ has the same nonzero structure as $\hat{C}$. The bound (3.12) then follows from Theorem 5 of [8]. □

We will examine the use of this result in §5.

REMARK 3. In the interest of brevity, we have limited our attention to the natural and red-black variants of the one-line orderings. Other variants, called "torus" one-line orderings, collect some individual lines together into sets of equal sizes; this is useful for parallel computations. (See [8],[14].) All of the analysis of this section also applies to the torus orderings.

We now turn our attention to incomplete (IC) factorizations. Let B be an M-matrix of order N, and let $\mathcal{N} \subseteq \{(i, j) \mid 1 \leq i, j \leq N\}$ be an index set containing all diagonal indices (i, i). It is shown in [15] that there is a unique IC factorization LU such that L is unit lower triangular, U is upper triangular, $l_{ij} = 0$ and $u_{ij} = 0$ for (i, j) $\notin \mathcal{N}$, and $[LU - B]_{ij} = 0$ for (i, j) $\in \mathcal{N}$. The IC(0) factorization of (2.4) is a particular example. The following result of Beauwens ([2], Theorem 4.4) can be used to compare the IC(0) splitting to the block Jacobi splitting.

THEOREM 2. *Let B be a nonsingular M-matrix, and let*

$$(3.15) \qquad B = M_1 - R_1 = M_2 - R_2,$$

where $M_1 = L_1U_1$ and $M_2 = L_2U_2$ are incomplete factorizations of B such that the set of matrix indices *for which $L_1 + U_1$ is permitted to be nonzero is contained in the set of* indices *for which $L_2 + U_2$ is permitted to be nonzero. Then*

$$(3.16) \qquad \rho(M_2^{-1}R_2) \leq \rho(M_1^{-1}R_1).$$

The analysis in [2] actually applies to a more general class of factorizations than the **standard** IC factorization. Theorem 2 can be proved using the result of Woinicki [25], that if (3.15) represents two regular splittings of a matrix B for which $B^{-1} \geq 0$, then

$$M_2^{-1} \geq M_1^{-1} \tag{3.17}$$

implies the conclusion (3.16). It is straightforward to establish (3.17) for IC factorizations.

COROLLARY *3. Suppose $A^{(b)}$ is an M-matrix, ordered* **wing** *any of the orderings under consideration. Let $A^{(b)} = M - R$ where M is the IC(0) factorization of $A^{(b)}$, and let $A^{(b)} = D - C$ denote the block JQCb's splitting. Then $\rho(M^{-1}R) \leq \rho(D^{-1}C)$.*

Proof. The index set of nonzeros of the block diagonal D is a proper subset of the **nonzero** index set for the **IC(0)** factorization. The result then follows from Theorem 2, where (the factorization of) D is viewed as an incomplete factorization of $A^{(b)}$. □

Thus, we expect convergence of a stationary method based on the **IC(0)** splitting to be at least as fast as that for the block Jacobi method, for any ordering. (The work per step for the Jacobi method will be smaller, though.) In particular, as observed in [7], [8], convergence should be faster for mildly nonsymmetric problems than for symmetric ones. Combining the **IC(0)** factorization with an acceleration scheme such as GMRES (i.e. using M as a preconditioner) should further improve convergence. Numerical experiments with the **IC(0)** preconditioner that support this statement are presented in the following sections.

4. Experimental Results: Constant Coefficient Problems.

In this section, we examine the numerical performance of the block Gauss-Seidel and SOR stationary methods, and GMRES(5) with the **IC(0)** preconditioner, for solving the constant coefficient model problem

$$-\Delta u + \sigma u_x + \tau u_y = 0 \tag{4.1}$$

on $\Omega = (0,1) \times (0,1)$. Dirichlet boundary conditions on $\partial\Omega$ are determined from the exact solution

$$u(x,y) = \frac{e^{\sigma x} - 1}{e^{\sigma} - 1} + \frac{e^{\tau y} - 1}{e^{\tau} - 1} \tag{4.2}$$

on $\bar{\Omega}$. The vector (σ, τ) represents a velocity field with the signs of σ or τ determining the direction of flow. We consider eight types of velocity fields, corresponding to eight flow directions in the (z, y)-plane:

East (E): $\sigma > 0$, $\tau = 0$,	Northeast (NE): $\sigma = \tau > 0$,
West (W): $\sigma < 0$, $\tau = 0$,	Southeast (SE): $\sigma = -\tau > 0$,
North (N): $\sigma = 0$, $\tau > 0$,	Northwest (NW): $\sigma = -\tau < 0$,.
South (S): $\sigma = 0$, $\tau < 0$,	Southwest (SW): $\sigma = \tau < 0$.

(For $\sigma = 0$ or $\tau = 0$, (4.2) is defined using the limit, i.e. $\lim_{\sigma \to 0} \frac{e^{\sigma x} - 1}{e^{\sigma} - 1} = x$.) In addition, the solution (4.2) has a boundary layer at any outflow boundary, i.e. near x = 1 for positive σ and x = 0 for negative σ, and similarly for y and τ. Plots of the solution for four such (σ, τ) combinations, corresponding to flows in the east, north, northeast and southeast directions, are shown in Fig. 4.1. Our concern is to determine the effects of direction and magnitude of flow, ordering of unknowns, discretization scheme, and use of local mesh refinement, on the performance of reduced system iterative methods.

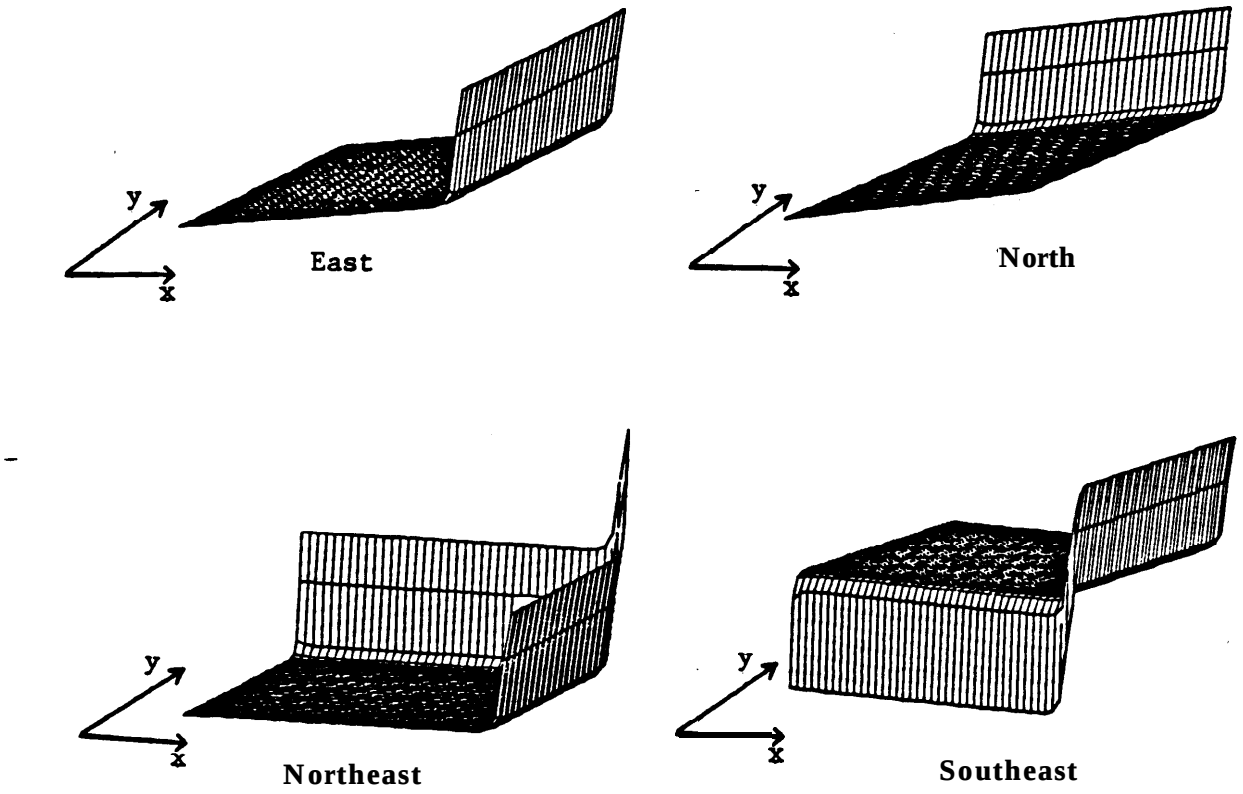


Fig. 4.1: Plots of the constant coefficient solution for four different directions of flow.

Details of the numerical experiments are as follows. The experiments were performed on a VAX-8600 in double precision Fortran. Reported iteration counts are averages over three initial guesses consisting of vectors of random numbers in $[-1, 1]$. The stopping criterion for all methods was $\|r_i\|_2 / \|r_0\|_2 \leq 10^{-6}$. A maximum of 150 iterations was permitted; an asterisk "*" in any table entry below indicates that for at least one initial guess, the stopping criterion was not met after 150 steps. (We remark that when the block stationary methods failed to meet the stopping criterion, they never "stagnated," i.e. they appeared to be converging.) For red-black SOR, the first iteration was performed with w = 1, as in [22]. Preconditioned GMRES was performed with right-oriented preconditioning, i.e. GMRES was applied to the preconditioned problem $A^{(b)} M^{-1} \hat{u}^{(b)} = g^{(b)}$, where M is the preconditioning matrix and $u^{(b)} = M^{-1} \hat{u}^{(b)}$. The construction of the reduced matrices and the experiments with GMRES were performed with PCGPAK [19].

	mrx $\lvert\sigma\rvert$, $\lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	SE $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	SW $\sigma=\tau<0$	Avg.
Gauss-Seidel	10	124	148	124	149	63	101	101	117	116
	50	17	35	17	35	5	19	19	35	23
	100	7	26	7	26	8	14	14	40	18
	200	12	31	12	31	32	28	28	71	31
	500	53	75	53	75	124	123	122	150*	97*
	1000	150*	150*	150*	150*	150*	150*	150*	150*	150*
SOR	10	34	47	34	47	22	33	33	44	37
	50	13	30	13	30	4	17	17	32	19
	100					5	15	15	33	17
	200					11	24	23	36	23
	500					27	37	37	42	36
	1000					54	61	60	65	60
GMRES / IC	10	15	16	14	15	11	16	17	14	15
	50	12	12	8	8	4	16	16	5	10
	100	11	11	6	6	5	15	14	6	9
	200	10	10	4	4	7	14	13	7	9
	500	10	10	4	4	11	17	17	12	11
	1000	9	9	4	4	18	22	21	20	13

Table 4.1: Average iteration counts for the natural one-line ordering, for eight flow directions.

	max $\lvert\sigma\rvert$, $\lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	SE $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	SW $\sigma=\tau<0$	Avg.
Gauss-Seidel	10	132	144	133	144	82	103	103	108	119
	50	23	24	23	24	19	18	18	21	23
	100	13	14	13	14	22	11	11	26	15
	200	20	21	20	21	49	27	27	57	30
	500	63	69	63	69	140*	128	128	150*	102*
	1000	150*	150*	150*	150*	150*	150*	150*	150*	150*
SOR	10	33	34	33	34	27	29	30	28	31
	50	23	24	23	24	19	18	18	21	21
	100					18	14	14	19	16
	200					21	23	22	22	22
	500					31	35	34	33	33
	1000					57	58	57	57	57
GMRES / IC	10	24	28	25	30	27	29	27	32	28
	50	29	35	26	35	37	22	20	51	32
	100	28	33	27	35	38	16	16	53	31
	200	28	34	28	34	37	14	14	53	30
	500	31	34	31	33	35	27	26	49	33
	1000	39	42	39	43	46	52	52	53	46

Table 4.2: Average iteration counts for the red-black one-line ordering, for eight flow directions.

	mrx $\lvert\sigma\rvert, \lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	S E $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	s w $\sigma=\tau<0$	Avg.
	10	101	109	92	115	50	84	72	87	89
	50	22	23	9	25	7	22	8	23	18
Gauss	100	13	13	8	23	6	21	7	21	14
Seidel	200	9	9	15	31	13	28	14	28	19
	500	6	6	52	64	47	63	53	64	44
	1000	5	5	150*	150*	143	150*	148*	150*	117*
	10	30	31	22	33	25	37	26	38	30
	50	19	20	6	20	6	21	8	22	15
SOR	100					9	25	11	25	17
	200					16	29	17	29	23
	500					31	41	31	41	36
	1000					56	64	56	65	60
	10	17	16	17	1 7	12	19	18	18	17
	50	12	13	12	13	5	27	25	5	10
;MRES	100	10	10	10	11	5	30	30	5	14
/ IC	200	8	8	8	9	10	33	30	10	14
	500	7	7	8	8	22	43	41	22	20
	1000	6	6	8	8	45	49	49	48	28

Table 4.3: Average iteration counts for the natural two-line ordering, for eight flow directions.

	mrx $\lvert\sigma\rvert, \lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	SE $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	SW $\sigma=\tau<0$	Avg.
	10	100	110	100	109	60	78	78	82	90
	50	19	20	17	18	14	15	15	16	17
Gauss-	100	10	11	15	16	13	14	13	14	13
Seidel	200	8	8	22	24	20	21	21	21	18
	500	6	6	56	58	54	56	59	57	44
	1000	5	5	150*	150*	146	150*	150*	149*	113*
	10	24	26	24	25	28	29	29	29	26
	50	15	16	13	14	13	14	14	15	14
SOR	100					17	17	17	17	17
	200					21	23	22	23	22
	500					34	35	35	35	35
	1000					58	58	58	58	58
	10	20	21	20	23	16	23	23	25	21
	50	12	13	25	31	15	23	24	25	21
3MRES	100	8	9	26	30	16	22	24	25	20
/ IC	200	6	7	26	30	17	23	23	28	20
	500	8	9	34	29	24	30	28	31	24
	1000	7	8	40	43	36	42	41	45	33

Table 4.4: Average iteration counts for the red-black two-line ordering, for eight flow directions.

The orientation of line orderings was as in §2. That is, for the one-line orderings, lines were oriented in the NW-SE direction, and the natural ordering arranged the lines starting from the SW corner; and for the two-line orderings, line pairs were grouped by horizontal lines and the natural listing is from bottom (south) to top (north). Note that the lines associated with ordering strategies have a relationship with the direction of flow (see also [4]). For example, for the natural one-line ordering, when the flow direction is NE, the lines are perpendicular to the direction of flow, and the Gauss-Seidel and SOR sweeps follow the flow. When the flow direction is SW, the lines are perpendicular to flow, but the sweeps are in the opposite direction of the flow. On the other hand, the sweeps for the red-black orderings do not have a clear relationship to the direction of flow (although the line orientations still do). The IC(0) preconditioning entails lower and upper triangular solves, so that, for the natural line orderings, the preconditioning operation can be thought of as a pair of bidirectional sweeps.

Tables 4.1 – 4.4 contain results for centered difference discretizations on a uniform mesh of width $h = 1/32$. For this class of problems, the analysis of §3 is applicable when $|\sigma h/2|$ and $|\tau h/2|$ are both less than one, i.e. when σ or τ are 10 or 50 in the problems considered. In these cases, Corollary 1 is used to choose the SOR parameter w, where $\rho(D^{-1}C)$ is approximated using the bounds (3.10) and (3.12); here

$$(4.3) \qquad a_i^{(x)} = \alpha^{(x)} = a_j^{(y)} = \alpha^{(y)} = 2, \qquad \xi = 1 - (\sigma h/2)^2, \qquad \eta = 1 - (\tau h/2)^2.$$

For the one-line orderings, when both $|\sigma h/2|$ and $|\tau h/2|$ are greater than one, the Fourier analysis of [7] can be used to estimate $\rho(D^{-1}C)$, from which good values of w are also obtained. (i.e. using the formula for ω^* in Corollary 1). These values were also used for the two-line orderings when $|\sigma h/2| > 1$ and $|\tau h/2| > 1$, although there is no theoretical justification for this. We did not examine SOR when one of $|\sigma h/2|, |\tau h/2|$ is greater than one and the other is less than one. Table 4.5 shows the choices of w used for Tables 4.1 – 4.4. Note that the analysis of §3 and [7],[8], does not distinguish between natural and red-black orderings, or between problems where the magnitudes of σ (or τ) are the same but the signs differ.

	One-line Orderings		Two-lineOrderings		
max $\lvert\sigma\rvert, \lvert\tau\rvert$	E/W/N/S	NE/SE/NW/SW	E/W	N/S	NE/SE/NW/SW
10	1.63	1.52	1.52	1.52	1.44
50	1.07	1.02	1.06	1.04	1.01
100		1.05			1.05
200		1.27			1.27
500		1.60			1.60
1000		1.77			1.77

Table 4.5: Values of SOR parameters used for Tables 4.1 – 4.4.

We make the following observations on the data of Tables 4.1 – 4.4:

1. For the stationary methods (Gauss-Seidel and SOR), performance depends on the relationship between flow direction and sweep direction, but the effects vary depending on

the magnitudes of the velocity vectors. For example, for the natural one-line orderings, when the convection terms are small or moderate in size, the best performance of the Gauss-Seidel and SOR methods occurs when the sweeps follow the flow (i.e. when the flow direction is NE). When the convection terms dominate, the stationary methods perform better when the flow direction forms a nonzero acute angle with the sweep direction (flow is N or E), than when the sweeps follow the flow. For the natural two-line ordering, performance for moderate sized convection terms is best when the flow direction forms an acute angle with the sweep direction (i.e. when flow is N, NE or NW); for convection-dominated systems, performance is best when the sweep is perpendicular to the flow. It is always the case that sweeping in the opposite direction of the flow is a bad choice.

2. Performance of stationary methods for the red-black orderings is much less sensitive to flow directions. In particular, the average iteration counts (over the eight flow directions) are essentially the same for the natural and red-black orderings. This is significant on parallel architectures, where the red-black orderings can be implemented more efficiently [8]. The minimum iteration counts are typically lower for the natural orderings than for the red-black orderings.

3. Somewhat different conclusions apply for GMRES/IC. There is no clear correlation between direction of flow and performance, except that for convection dominated problems, performance for both natural orderings degrades when the directions of flow are not parallel to one of the grid coordinates. We have no simple explanation for this. The average iteration counts for GMRES/IC are typically higher for the red-black orderings than for the natural orderings. Similar results have been obtained for symmetric problems, with point red-black and natural orderings, e.g. in [1].

4. One step of the block SOR method is approximately as expensive as one matrix vector-product and one scalar-vector product [8]. Thus, its cost per step is approximately $10N_b$ multiply-adds, where N_b is the order of $A^{(b)}$. One step of GMRES(5) with IC(0) preconditioning entails a preconditioning solve, a matrix-vector product, and approximately $8N_b$ vector operations [21], for a total cost of $26N_b$ multiply-adds. That is, one GMRES/IC step is about 2.5 times as expensive as one SOR step. Consequently, the performances of the stationary methods and GMRES/IC are comparable for problems with small and moderate-sized convection terms (where for problems with small convection terms, it is necessary to use a good SOR parameter to achieve good performance). GMRES/IC is somewhat more effective for convection-dominated systems, especially when there is no simple way of choosing a relaxation parameter. GMRES(5) requires $7N_b$ storage locations [21], plus approximately $9N_b$ for the factors of M. SOR requires essentially one vector of storage for the solution iterates $\{u_b^{(k)}\}$, plus storage for the factors of the block diagonal D. If no pivoting is required, these factors could overwrite the analogous locations of $A^{(b)}$.

Table 4.6 shows the performance of the block Gauss-Seidel method for solving the same set of problems using the upwind difference scheme for the first derivative terms. The main difference from the results for centered differences is that performance improves as σ or τ increases. This is because $A^{(b)}$ *(as well as A)* becomes more diagonally dominant in these cases. In addition, for the natural one-line ordering, performance is consistently best when the flow is in the same direction as the sweep (NE), and good performance is achieved when the sweep and flow directions make an acute angle. Similar observations

	max $\lvert\sigma\rvert, \lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	SE $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	SW $\sigma=\tau<0$	Avg.
	10	134	150*	135	150*	77	116	116	133	126*
	50	30	48	30	48	16	34	34	49	36
Natural	100	16	33	16	33	9	24	24	40	24
One-line	200	9	26	9	26	5	19	19	35	19
	500	5	22	5	22	3	17	17	33	15
	1000	4	20	4	20	2	16	16	32	14
	10	143	150*	144	150*	93	118	118	124	130*
	50	37	39	37	39	31	33	33	36	36
Red-black	100	23	24	23	24	23	23	23	26	24
One-line	200	16	17	16	17	20	18	18	21	18
	500	13	13	13	13	17	15	15	19	15
	1000	11	12	11	12	16	14	14	18	14
	10	104	113	105	129	54	95	84	99	98
	50	27	28	24	41	17	34	20	35	28
Natural	100	16	17	13	29	11	27	13	27	19
Two-line	200	11	11	8	23	7	23	9	23	14
	500	7	7	5	20	5	20	6	21	11
	1000	5	6	3	19	4	19	5	20	10
	10	103	113	113	124	65	90	90	94	99
	50	24	26	32	34	24	27	27	28	28
Red-black	100	14	15	21	22	18	20	13	20	18
Two-line	200	9	10	5	16	14	6	6	16	10
	500	6	6	12	13	12	13	13	14	11
	1000	5	5	11	12	11	12	12	13	10

Table 4.6: Average iteration counts for the block Gauss-Seidel method, upwind differences,

apply for the natural two-line ordering, except that sweeping in the direction of flow (N) is not best when the convection terms are small. As above, the red-black orderings tend to be less sensitive than the natural orderings to flow directions.

The results above do not address the issue of accuracy of the discrete solution. If $|\sigma h/2|$ or $|\tau h/2|$ is greater than one and boundary layers are present in the continuous solution, then the discrete solution tends to be inaccurate near the boundary layers, and it is oscillatory when centered differences are used [20]. If the boundary layer can be located, then one possible remedy is to use local mesh refinement. For the solution (4.2), for nonzero σ or τ, there are boundary layers of width $O(1/\sigma)$ (or $O(1/\tau)$) near the outflow boundary. We consider one local refinement strategy, which we describe in terms of the "horizontal" parameters x and σ. In the interval of width $2/\sqrt{\sigma}$ containing the boundary layer (at either $x = 0$ or $x = 1$), we use a mesh of size $\tilde{h}$ such that $|\sigma\tilde{h}/2| = .75$; away from that interval, we use $h = 1/32$.[3] It was shown in [6] that this strategy does a good job of resolving the

[3] Grid points are distributed from left to right within each of these subintervals, so that the rightmost mesh width of either interval may differ from h and $\tilde{h}$.

boundary layer with the addition of a relatively small number of additional mesh points. For example, in the present set of experiments, when σ = 100 there are 25 coarse grid points and 14 fine grid points in the horizontal direction; when σ = 1000, there are 29 coarse and 43 fine grid points. (The unrefined mesh contains 31 points in each direction.) Table 4.7 shows the performance of the Gauss-Seidel and GMRES/IC methods for four problems where mesh refinement is used, for the natural one-line ordering. Comparison with Table 4.1 shows that the behavior of the two iterative methods is essentially the same as that for uniform meshes. Similar conclusions apply for the three other ordering strategies. Thus, we conclude that the behavior on uniform meshes is indicative of behavior where mesh refinement is used to resolve boundary layers. (Experiments with the Gauss-Seidel method for $\max(|\sigma|,|\tau|)$ = 1000 were not performed because of storage constraints in our implementation.)

	max $\lvert\sigma\rvert, \lvert\tau\rvert$	E $\sigma>0,\tau=0$	W $\sigma<0,\tau=0$	N $\sigma=0,\tau>0$	S $\sigma=0,\tau<0$	NE $\sigma=\tau>0$	SE $\sigma=-\tau>0$	NW $\sigma=-\tau<0$	SW $\sigma=\tau<0$	Avg.
Gauss-Seidel	100	7	31	7	31	8	17	17	47	21
	200	12	37	12	37	32	28	28	80	33
	500	46	73	46	73	1 2 4	111	109	150*	91*
	1000	134	150*	132	150*					
GMRES / IC	100	12	12	6	6	6	17	17	6	10
	200	10	10	4	4	8	18	17	8	10
	500	10	10	4	3	12	18	21	11	11
	1000	9	9	4	2	18	23	24	14	13

Table 4.7: Average iteration counts for the natural one-line ordering, centered differences and local mesh refinement.

5. Experimental Results: Separable Variable Coefficient Problems.

In this section, we examine the use of Corollary 2 to derive bounds on $\rho(D^{-1}C)$ when $A^{(b)}$ comes from a separable operator. We consider three model problems taken from [3]. Other experiments with these problems are described in [7].

PROBLEM 5.1: $-Au + \frac{\sigma}{2}(1+x^2)u_x + \tau u_y = 0$ on $\Omega = (0,1) \times (0,1)$

$u = o$ on $\partial\Omega$.

Discretization by centered differences gives, after scaling by h^2,

$$a_i^{(x)} = \alpha^{(x)} = a_j^{(y)} = \alpha^{(y)} = 2,$$

$$c_{i+1}d_i = \left(1+\tfrac{\sigma h}{4}(1+x_{i+1}^2)\right)\left(1-\tfrac{\sigma h}{4}(1+x_i^2)\right) \le 1 - \tfrac{1}{4}\left(\tfrac{\sigma h}{2}\right)^2 + \sigma h^2 = \xi, \tag{5.1}$$

$$b_{j+1}e_j = 1 - \left(\tfrac{\tau h}{2}\right)^2 = \eta.$$

	Centered Differences				Upwind Differences			
	One-line		Two-line		One-line		Two-line	
$\sigma = \tau$	Computed	Bound	Computed	Bound	Computed	Bound	Computed	Bound
20	.741	.809	.674	.731	.817	1.298	.772	1.379
40	.323	.385	.236	.275	.611	1.182	.544	1.212
60	.047	.062	.015	.018	.455	.961	.386	.985

Table 5.1: Comparison of computed spectral radii and bounds for the block Gauss-Seidel iteration matrices, for Problem 5.1 with h=1/32.

For $\sigma \geq 0$ and $\tau \geq 0$, upwind discretization gives

$$a_i = 2 + \tfrac{\sigma h}{2}(1 + x_i^2) \geq 2 + \tfrac{\sigma h}{2} = \alpha^{(x)},$$
$$a_j = 2 + \tau h = \alpha^{(y)},$$
$$c_{i+1} d_i = 1 + \tfrac{\sigma h}{2}(1 + x_{i+1}^2) \leq 1 + ah = \xi,$$
$$b_{j+1} e_j = 1 + \tau h = \eta.$$

Table 5.1 compares the bounds for $\rho(\mathcal{L}_1) = \rho(D^{-1}C)^2$ obtained from Corollary 2 with the corresponding computed values of $\rho(\mathcal{L}_1)$, for $h = 1/32$. For this problem, as well as the others considered below, we examine several choices of σ and τ where for the largest such choice, $\max_{x_i} |r(x_i)h/2|$ and $\max_{y_j} |s(y_j)h/2|$ are both close to one.

PROBLEM 5.2: $-Au + \sigma x^2 u_x = 0$ on $\Omega = (0,1) \times (0,1)$
$u = o$ on $\partial\Omega$.

Centered difference discretization gives

$$a_i^{(x)} = \alpha^{(x)} = d_j^{(y)} = \alpha^{(y)} = 2,$$
$$c_{i+1} d_i = \left(1 + \tfrac{\sigma h}{2} x_{i+1}^2\right)\left(1 - \tfrac{\sigma h}{2} x_i^2\right) \leq 1 + \tfrac{\sigma h^2}{2} - \tfrac{\sigma^2 h^4}{2} = \xi,$$
$$b_{j+1} e_j = 1 = \eta.$$

Upwind difference discretization gives

$$a_i = 2 + \sigma x_i^2 h \geq 2 + \sigma h^3 = \alpha^{(x)},$$
$$a_j = 2 = \alpha^{(y)},$$
$$c_{i+1} d_i = 1 + \sigma x_{i+1}^2 h \leq 1 + \sigma h = \xi,$$
$$b_{j+1} e_j = 1 = \eta.$$

Table 5.2 compares bounds for $\rho(\mathcal{L}_1)$ with corresponding computed values for Problem 5.2. **An entry "–" means that the analysis is not applicable because** (3.11) **is not satisfied.**

PROBLEM 5.3: $-Au + \sigma(1 - 2x)u_x + \tau(1 - 2y)u_y = 0$ on $\Omega = (0,1) \times (0,1)$
$u = o$ on $\partial\Omega$.

	Centered Differences				Upwind Differences			
	One-line		Two-line		One-line		Two-line	
σ	Computed	Bound	Computed	Bound	Computed	Bound	Computed	Bound
20	.963	1.014	.951	.987	.964	3.077	.951	6.630
40	.953	1.033	.939	1.011	.955	10.37	.939	–
60	.945	1.051	.928	1.035	.947	56.22	.928	–

Table 5.2: Comparison of computed spectral radii and bounds for the block Gauss-Seidel iteration matrices, for Problem 5.2 with h=1/32.

Centered difference discretization gives

$$
\begin{aligned}
&a_i^{(x)} = \alpha^{(x)} = a_j^{(y)} = \alpha^{(y)} = 2,\\
&c_{i+1}d_i = \left(1 + \tfrac{\sigma h}{2}(1 - 2x_{i+1})\right)\left(1 - \tfrac{\sigma h}{2}(1 - 2x_i)\right)\\
&\qquad = 1 - 2h\left(\tfrac{\sigma h}{2}\right) - \left(\tfrac{\sigma h}{2}\right)^2(1 - 2x_i)(1 - 2x_{i+1})\\
&\qquad \le 1 - \sigma h^2 + \left(\tfrac{\sigma h}{2}\right)^2(2h^3 - h^4) = \xi,\\
&b_{j+1}e_j = \left(1 + \tfrac{\tau h}{2}(1 - 2y_{j+1})\right)\left(1 - \tfrac{\tau h}{2}(1 - 2y_j)\right)\\
&\qquad \le 1 - \tau h^2 + \left(\tfrac{\tau h}{2}\right)^2(2h^3 - h^4) = \eta,
\end{aligned}
$$

For $\sigma \ge 0$ and $\tau \ge 0$, upwind discretization gives

$$
\begin{aligned}
&a_i = 2 + \sigma|1 - 2x_i|h \ge 2 = \alpha^{(x)},\\
&a_j = 2 + \tau|1 - 2y_j|h \ge 2 = \alpha^{(y)},\\
&c_{i+1}d_i = 1 + \sigma|1 - 2x_{i+1}|h \le 1 + \sigma h = \xi,\\
&b_{j+1}e_j = 1 + \tau|1 - 2y_{j+1}|h \le 1 + \tau h = \eta.
\end{aligned}
$$

Table 5.3 compares bounds and computed values of $\rho(\mathcal{L}_1)$ for Problem 5.3; the entry "–" indicates that either (3.9) or (3.11) is not satisfied.

	Centered Differences				Upwind Differences			
	One-line		Two-line		One-line		Two-line	
$\sigma = \tau$	Computed	Bound	Computed	Bound	Computed	Bound	Computed	Bound
20	.854	.921	.813	.869	.871	3.611	.833	6.986
40	.733	.852	.669	.785	.780	–	.723	–
60	.629	.788	.553	.710	.703	–	.634	–

Table 5.3: Comparison of computed spectral radii and bounds for the block Gauss-Seidel iteration matrices, for Problem 5.3 with h=1/32.

To understand these results, it is useful to recall the constant coefficient problem (4.1). For that problem, the parameters associated with centered differences are given by ***(4.3).***

As shown in [7], [8], if both $\sigma h/2 < 1$ and $\tau h/2 < 1$, then the bounds from Corollary 2 essentially have the form $1 - O(\sigma^2 h^2) - O(\tau^2 h^2)$. In particular, if either $\sigma h/2$ or $\tau h/2$ are near 1, then ξ or η are close to 0, and the bounds from Corollary 2 are very small. For Problem 5.1, r(x) (the coefficient of u_x) is bounded below away from 0, so that for large σ, the contribution $hr(x_i)/2$ cannot be small for any x_i. Consequently, the bounding value ξ is qualitatively like its constant coefficient counterpart (compare (5.1) and (4.3)). Moreover, $\alpha^{(x)}$, $\alpha^{(y)}$ and η have the same values as in the constant coefficient case. (This is true for $\alpha^{(x)}$ and $\alpha^{(y)}$ with all three problems considered here.) Thus, the bounds from Corollary 2 behave like their constant coefficient analogues. For Problem 5.2, the upper bound ξ corresponds to a value for $x_i (= h)$ for which the differential operator is locally nearly self-adjoint; the resulting bounds typically do not even guarantee convergence, and they are larger than what would be obtained in the self-adjoint case. For Problem 5.3, $\xi = 1 - O(\sigma h^2)$ and $\eta = 1 - O(\sigma h^2)$, which lead to asymptotic bounds of the form $1 - O(\sigma h^2) - O(\tau h^2)$; these are larger than those occurring for Problem 5.1 but smaller than for Problem 5.2. Note that for all three problems, the bounding values are qualitively similar to the behavior of $\mathcal{L}_1$.

The parameters for upwind differences applied to the constant coefficient problem are

$$a_i^{(x)} = \alpha^{(x)} = 2 + uh, \quad a_j^{(y)} = \alpha^{(y)} = 2 + \tau h, \quad \xi = 1 + uh, \quad \eta = 1 + \tau h.$$

-Although ξ and η do not approach zero, the bounds on $\rho(D^{-1}$ C) from Corollary 2 are less than one, and they decrease with increasing σ or τ (see [7], [8]). However, the extra inequalities required to define $\alpha^{(x)}$ and $\alpha^{(y)}$ decrease the size of the denominators in (3.10) and (3.12) and limit the usefulness of the corollary. For Problem 5.1, ah is replaced by $\sigma h/2$ in $\alpha^{(x)}$, and the bounds on $\rho(D^{-1}C)$ are less than one only when ah is large. The bounds for Problems 5.2 and 5.3, where they are defined, do not provide any useful information.

6. Experimental Results: Nonseparable Variable Coefficient Problems.

We now examine the performance of the iterative methods for solving some nonseparable problems. Our goals are to examine the effectiveness of the block Gauss-Seidel and SOR methods, and IC-preconditioned GMRES, for solving such problems; and to determine whether the analytic results of [7] [8] and §3 are of use in predicting behavior.

We consider two model equations that differ only in their boundary conditions. Both equations model a circular flow of a fluid around a point. The velocity vectors have turning points in the vertical component, and their magnitudes vary throughout the domain of definition.

PROBLEM 6.1: $-\epsilon\Delta u + 2y(1-x^2)u_x - 2x(1-y^2)u_y = 0$ on $\Omega = (-1,1) \times (0,1)$

u = o	on $0 \le y \le 1$, $x = -1$,
u = 100	on $0 \le y \le 1$, x = 1,
u = o	on $-1 \le x < 0$, $y = 0$,
$u_n = 0$	on $0 \le x \le 1$, $y = 0$,
u = o	on $-1 \le x \le 1$, y = 1.

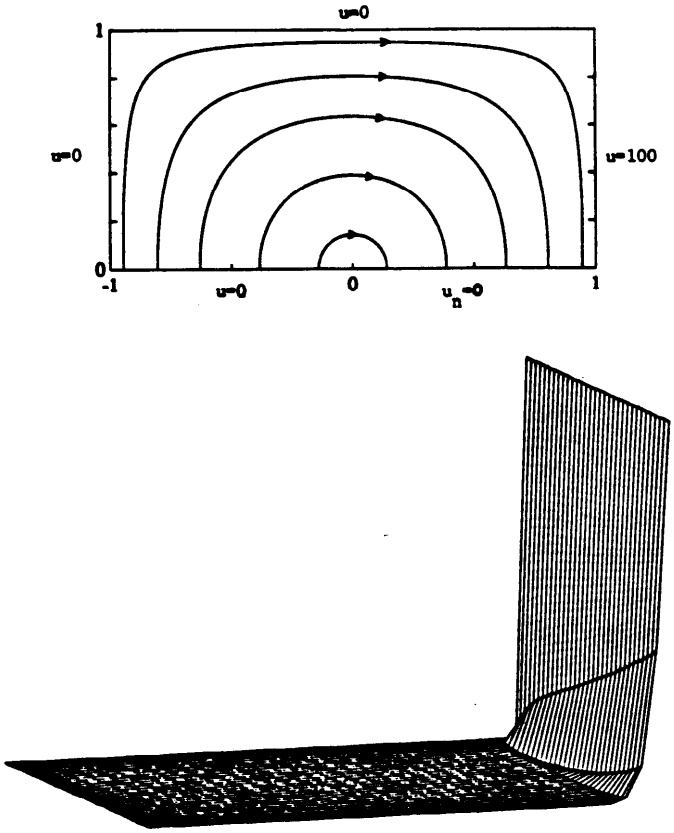


Fig. 6.1: Boundary conditions and solution for Problem 6.1.

This problem is taken from [17]. It models the flow of a cold fluid with a hot wall at the right boundary. The solution contains a boundary layer at x = 1. Fig. 6.1 shows the boundary conditions and streamlines, and the general shape of the solution, for $\epsilon = 1/100$.[4]

PROBLEM 6.2: $-\epsilon\Delta u + 2y(1-x^2)u_x - 2x(1-y^2)u_y = 0$ on $\Omega = (-1,1)\times(0,1)$

$u = 0$	on $0 \le y \le 1$, $x = \pm 1$,
$u = 1 + \tanh(10(1 + 2x))$	on $-1 \le 3 < 0$, $y = 0$,
$u_n = 0$	on $0 \le x \le 1$, $y = 0$,
$u = 0$	on $-1 \le x \le 1$, $y = 1$.

This problem is taken from [13]. The differential operator is the same as that of Problem 6.1. The solution contains a boundary layer near the point x = 0, y = 0. Fig. 6.2 shows the boundary conditions and a representative solution.

As above, we consider centered differences and upwind differences to discretize these problems. We discretize the outflow boundary condition at y = 0 by first order upwind

[4] The discrete solutions depicted in Figs. 6.1 and 6.2 were computed using centered differences with 31 interior grid points in each direction; the figures include the exact solution values at $x=\pm 1$ and $y=1$, but not at $y=0$.

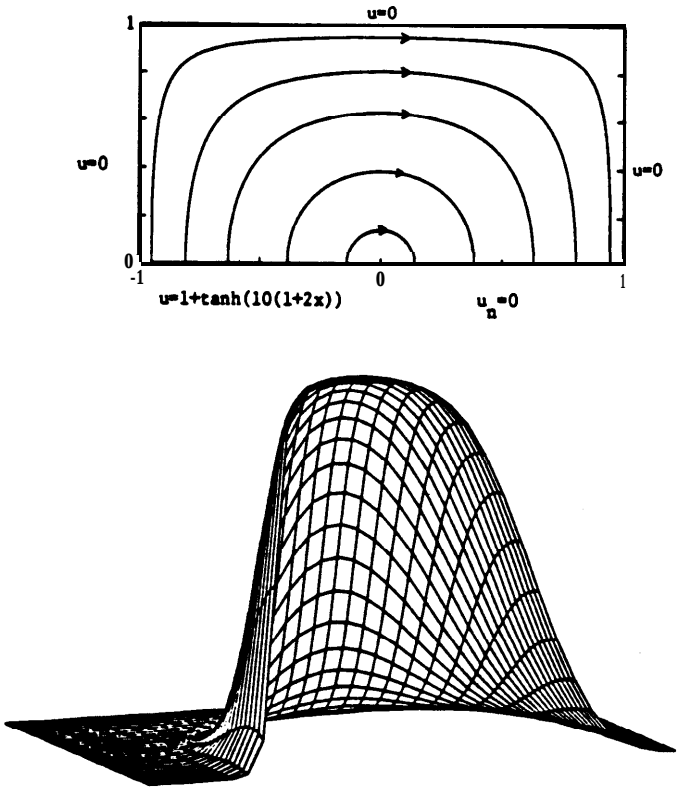


Fig. 6.2: Boundary conditions and solution for Problem 6.2.

differences,

$$0 = u_n(x_i, 0) = u_y(x_i, y_0) \approx \frac{u(x_i, y_1) - u(x_i, y_0)}{h},$$

i.e. $u(x_i, 0) = u(x_i, y_1)$. For the centered difference scheme, we consider both a square 31 x 31 mesh, and a uniform mesh of width $h = 1/32$. The first choice produces matrices with the same algebraic structure as those considered in §§4 – 5, but the horizontal mesh width is twice that of the vertical width; the-second choice leads to lines of different length in the grid. We also consider a strategy for improving the accuracy of the solution, based on defect correction methods. For all experiments, the initial guesses and stopping criteria are as in 94.

Tables 6.1 and 6.2 show average iteration counts for solving the reduced system derived when centered differences are applied on a square 31 x 31 grid. Here, the grid sizes for the full system are uniform in each of the x and y coordinates, with $h_x = 1/16$ and $h_y = 1/32$. As in the constant coefficient case (§4), block relaxation is most effective for intermediate values of ϵ^{-1}, where it is competitive with GMRES/IC. The latter method is more effective when ϵ^{-1} is either small or large. The performance of the stationary methods is fairly insensitive to the choice of ordering. This is consistent with the fact that, because of variable directions of flow, there is no clear correspondence between lines and flow direction. On the other hand, as in §4, the performance of GMRES/IC is typically

		1/ε					
	Ordering	10	50	100	200	500	1000
Gauss-Seidel	Natural One-line	122	22	27	57	150 (4)	150 (1)
	Red-black One-line	119	26	29	63	150 (4)	150 (1)
	Natural Two-line	114	24	26	54	150 (4)	150 (1)
	Red-black Two-line	111	25	26	54	150 (4)	150 (1)
GMRES / IC	Natural One-line	10	7	7	8	15	33
	Red-black Onoline	27	27	34	37	46	74
	Natural Two-line	14	10	10	10	19	87
	Red-black Two-line	24	21	26	26	42	71

Table 6.1: Average iteration counts for Problem 6.1 on a 31 x 31 grid ($h_x = 1/16, h_y = 1/32$), with centered differences. Numbers in parentheses are approximate number of digits of accuracy when methods did not meet the stopping criterion.

		1/ε					
	Ordering	10	50	100	200	500	1000
Gauss-Seidel	Natural Onoline	146	33	23	49	150 (5)	150 (4)
	Red-black Onoline	150 (> 5)	36	30	64	150 (5)	150 (4)
	Natural Two-line	143	33	26	50	148 (5)	150 (4)
	Red-black Two-line	145	33	27	55	150 (5)	150 (4)
GMRES / IC	Natural One-line	13	8	8	9	13	25
	Red-black Onoline	34	33	34	40	48	60
	Natural Two-line	18	11	10	11	17	65
	Red-black Two-line	33	23	24	29	37	56

Table 6.2: Average iteration counts for Problem 6.2 on a 31 x 31 grid ($h_x = 1/16, h_y = 1/32$), with centered differences. Numbers in parentheses are approximate number of digits of accuracy when methods did not meet the stopping criterion.

better with the natural orderings than with the red-black orderings. We also remark that in a few experiments with Orthomin [5], we found Orthomin(5) to be somewhat less robust than GMRES(5).

Table 6.3 shows iteration counts for solving the reduced system derived from an underlying uniform mesh of width $h = 1/32$, for block Gauss-Seidel and GMRES/IC, with the two natural line orderings. The lines are oriented as in Fig. 2.2. These results are similar to those of Tables 6.1 and 6.2, except that GMRES/IC has trouble with one problem class ($\epsilon = 1/1000$ with the natural one-line ordering). In this case (for both orderings), the iteration "stagnates," in the sense that the residual norm $\|g^{(b)} - A^{(b)}u_i^{(b)}\|_2$ remains constant over many iterations.[5] In contrast, whenever the block relaxation methods fail to meet the stopping criterion, they appear to be converging.

[5] Stagnation of this type also occurs for GMRES(lO) and GMRES(15).

		1/ε					
	Method	10	50	100	200	500	1000
Problem 6.1	G.S. Natural One-line	150 (5)	28	22	35	122	150 (3)
	G.S. Natural Two-line	129	27	22	34	101	150 (3)
	GMRES/IC Natural One-line	17	11	10	10	16	150 (3)
	GMRES/IC Natural Two-line	20	14	12	12	17	64
Problem 6.2	G.S. Natural One-line	150 (5)	49	24	35	122	150 (4)
	G.S. Natural Two-line	150 (5)	39	23	32	82	150 (5)
	GMRES/IC Natural One-line	22	16	11	10	16	150 (3)
	GMRES/IC Natural Two-line	28	22	15	12	17	26

Table 6.3: Average iteration counts for the natural one-line and two-line orderings, on a uniform grid with mesh size $h = 1/32$, with centered differences. Numbers in parentheses are approximate number of digits of accuracy when methods did not meet the stopping criterion.

		1/ε					
	Ordering	10	50	100	200	500	1000
Gauss-Seidel	Natural One-line	142	31	24	21	18	17
	Red-black One-line	139	37	29	26	24	23
	Natural Two-line	132	32	25	23	20	19
	Red-black Two-line	131	35	27	22	20	19
GMRES / IC	Natural One-line	10	8	8	7	7	6
	Red-black One-line	29	25	28	32	36	37
	Natural Two-line	15	10	10	9	8	7
	Red-black Two-line	28	20	20	21	26	25

Table 6.4: Average iteration counts for Problem 6.1 on a 31 x 31 grid ($h_x = 1/16, h_y = 1/32$), with upwind differences.

Table 6.4 shows average iteration counts for solving the reduced system derived when upwind differences are applied to Problem 6.1. Results for upwinding and Problem 6.2 were similar. Note that the mesh points used for discretization depend on the direction of flow (see §2), and the reduced matrices $A^{(b)}$ are always diagonally dominant. The results of Table 6.4 (for the stationary methods) are consistent with those for constant coefficient problems.

A methodology for improving accuracy that does not require a priori knowledge about the solution is the class of defect correction methods. A description of this approach can be found in [10], which contains several other references. For the operator $L_\epsilon u \equiv -\epsilon\Delta u + ru_x + su_y$, let $A_{\epsilon,h}$ denote the matrix associated with the (second order) centered difference discretization on a uniform mesh of width h. For $2 > \epsilon$, let $A_{\hat{\epsilon},h}$ denote the analogous matrix derived from $L_{\hat{\epsilon}}$. In its simplest form, the defect correction iteration consists of the following steps, where f is the discrete right hand side.

Solve $A_{\hat{\epsilon},h}u^{(m)} = f$.
For $m = 0, 1, \ldots,$ Do
$r^{(m)} = f - A_{\epsilon,h}u^{(m)}$
Solve $A_{\hat{\epsilon},h}d^{(m)} = r^{(m)}$
$u^{(m+1)} = u^{(m)} + d^{(m)}$
End

The idea is to compensate for instabilities associated with high order operators using lower order operators. For the choice $2 = \epsilon + ch$ where c > 0 is a fixed constant, $A_{\hat{\epsilon},h}$ is a first order discretization. At every step of the iteration, $A_{\epsilon,h}$ is used only to calculate the residual, and a linear system with coefficient matrix $A_{\hat{\epsilon},h}$ must be solved. Thus, the cost of this method is highly dependent on the cost of solving the linear system.

Any c > 0 prevents the convection terms from dominating the discrete problem, for arbitrarily small ϵ. For $c \geq \max\{|r(x,y)|/2, |s(x,y)|/2\}$, $A_{\hat{\epsilon},h}$ and the resulting reduced matrix $A^{(b)}_{\hat{\epsilon},h}$ are diagonally dominant M-matrices. For Problems 6.1 and 6.2, this gives = 1. However, Hemker [13] has observed that (using a variant of the algorithm above) better accuracy is obtained with smaller c. Following [13], we use c = 1/2. The differential operator $L_{\hat{\epsilon}}$ for Problems 6.1 and 6.2 is then equivalent to

$$-\Delta u + \frac{2y(1-x^2)}{\epsilon+h/2}u_x - \frac{2x(1-y^2)}{\epsilon+h/2}u_y.$$

We refer to the discretization of this operator by centered difference as the "defect correction discretization." Table 6.5 shows the performance of the various iterative methods for solving the resulting reduced linear systems. (See [13] for a discussion of the overall iteration.) These results are qualitatively similar to performance for upwind differences.

		$1/\epsilon$					
	Method	**10**	**50**	**100**	**200**	**500**	**1000**
Problem 6.1	G.S. Natural One-line	150 (4)	42	32	28	26	25
	G.S. Natural Two-line	150 (5)	38	30	27	25	25
	GMRES/IC Natural One-line	17	12	12	11	11	11
	GMRES/IC Natural Two-line	21	16	14	14	13	13
Problem 6.2	G.S. Natural One-line	150 (3)	85	62	50	43	41
	G.S. Natural Two-line	150 (4)	67	50	41	35	33
	GMRES/IC Natural One-line	23	21	18	16	13	13
	GMRES/IC Natural Two-line	23	26	24	22	20	20

Table 6.5: Average iteration counts to solve the linear systems arising from the defect correction method, for the natural one-line and two-line orderings on a uniform grid with mesh size $h = 1/32$. Numbers in parentheses are approximate number of digits of accuracy when methods did not meet the stopping criterion.

Finally, in contrast to the separable case, the spectra of the block Jacobi matrices arising from nonseparable operators are typically not real even when ϵ^{-1} is small. Consequently, Corollary 1 does not apply. For example, Fig. 6.3 shows the eigenvalue distributions in the complex plane of the block Jacobi matrices associated with the defect

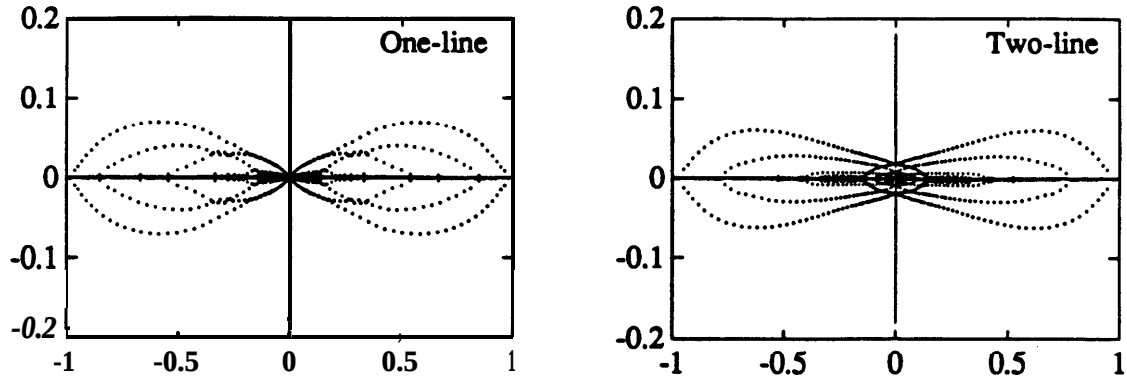


Fig. 6.3: Eigenvalues of the line Jacobi matrices for the reduced system, with ϵ=1/10, h=1/32, defect correction discretization.

correction discretizations for the problems used to produce Table 6.5, for $\epsilon = 1/10$. (The matrices are the same for Problems 6.1 and 6.2.) For these problems, the real parts of all the eigenvalues are less then one in absolute value, so that Young's analysis for complex eigenvalues could be used to choose a relaxation parameter to guarantee convergence ([26], 56.4). In such cases, an ellipse containing the spectrum could be found using the methods of [4]. An alternative adaptive strategy is based on the fact that when ϵ^{-1} is small, the -coefficient matrices are in some sense close to being symmetric. Thus, one could estimate p(B) and choose ω^* as in Corollary 1. Fig. 6.4 graphs average iteration counts required for convergence of line SOR, as a function of the SOR parameter w, for Problem 6.2 with the defect correction discretization and $h = 1/32$. The computation with ω^* is identified with an asterisk. These results suggest that this heuristic strategy gives a reasonable choice of ω when ϵ^{-1} is small.

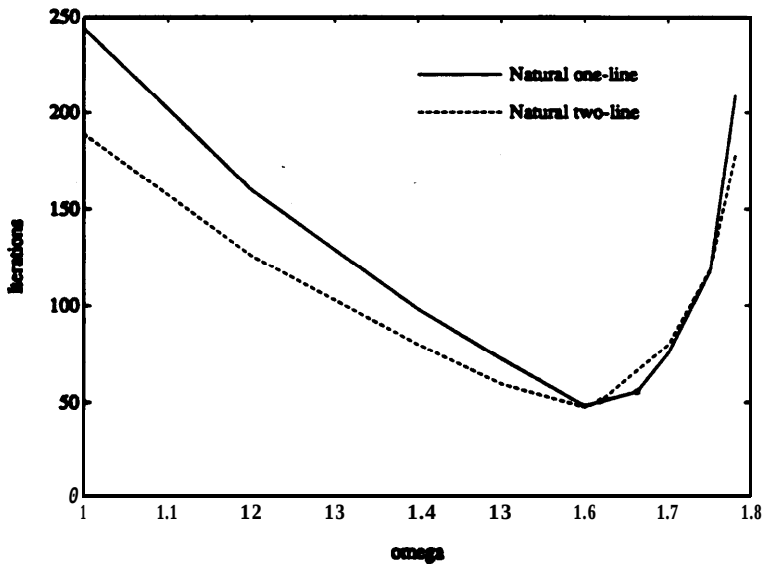


Fig. 6.4: Average line SOR iteration counts, for Problem 6.2 with ϵ=1/10, h=1/32, defect correction discretization.

7. Concluding Remarks.

In this paper, we have continued the study of line iterative methods for solving reduced systems begun in [7],[8]. We have extended the analysis in two ways. First, for matrices that arise from variable coefficient separable differential operators, we derived conditions under which the reduced matrices can be symmetrized via diagonal similarity transformations; previous results applied only to constant coefficient problems. Symmetrization *is* the key to the analysis of convergence behavior for the constant coefficient case. In the present analysis, it determines conditions under which the classical analysis of SOR applies, from which the optimal SOR parameter can be expressed as a simple function of the maximum eigenvalue of the line Jacobi iteration matrix, and it leads to some analytic bounds on performance for separable problems. In addition, we used regular splitting results to show that the analysis of line Jacobi splittings can be extended to splittings based on incomplete LU factorizations, for various line orderings of the reduced grid. The results help explain the good performance of IC preconditioners applied to the nonsymmetric matrix problems arising from the convection-diffusion equation.

We have also performed an extensive set of numerical experiments that examine the effects of direction of flow, discretization and grid ordering on performance of the line iterative methods. For constant coefficient problems, the results reveal correlations between relaxation sweep direction and direction of flow that are not displayed by any analytic results. They also show that for block relaxation methods, red-black orderings are less sensitive to flow directions than natural orderings, whereas for IC-preconditioned GMRES, convergence is faster for natural orderings than for red-black orderings. In addition, both block relaxation and IC preconditioned GMRES are effective for many problems where the analysis does not apply. In general, IC preconditioned GMRES is more robust than block relaxation. Finally, experimental results for problems with variable coefficients or locally refined grids are largely consistent with analysis and experiments for constant coefficients and uniform grids.

References

[1] C. C. Ashcraft and R. G. Grimes, On vectorizing incomplete factorization and SSOR preconditioners, *SIAM J. Sci. Std. Comput.* 9:122-151, *1988.*

[2] R. Beauwens, Factorization iterative methods, M-operators and H-operators, *Numer. Math.* 31:335-357, 1979.

[3] E. F. F. Botta and A. E. P. Veldman, On local relaxation methods and their application to convection-diffusion equations, *J. Comput. Phys.* 48:127-149, 1981.

[4] R. C. Y. Chin and T. A. Manteuffel, An analysis of block successive overrelaxation for a class of matrices with complex spectra, *SIAM J. Numer. Anal.* 25:564-585, 1988.

[5] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, Variational iterative methods for nonsymmetric systems of linear equations, *SIAM J. Numct. Anal.* 20:345-357, 1983.

[6] H. C. Elman, Relaxed and Stabilized Incomplete Factorizations for Non-Self-Adjoint Linear Systems, Report UMIACS-TR-89-1, University of Maryland, Jan. 1989. To appear in *BIT.*

[7] Iterative Methods for Cyclically Reduced Non-Self-Adjoint Linear Systems, Report UMIACS-TR-88-87, University of Maryland, Nov. 1988. To appear in ***Math. Comp.***

[8] Iterative Methods for Cyclically Reduced Non-Self-Adjoint Linear Systems II, Report UMIACS-TR-89-45, University of Maryland, Jun. 1989. To appear in ***Math. Comp.***

[9] G. E. Forsythe and W. R. Wasow, ***Finite Difference Methods for Partial Differential Equations,*** John Wiley and Sons, New York, 1960.

[10] W. Hackbusch, ***Multi- Grid Methods and Applications,*** Springer-Verlag, Berlin, 1985.

[11] L. A. Hageman and R. S. Varga, Block iterative methods for cyclically reduced matrix equations, ***Numer. Math.*** 6:106-119, 1964.

[12] L. A. Hageman and D. M. Young, ***Applied Iterative Methods,*** Academic Press, New York, 1981.

[13] P. W. Hemker, Mixed defect correction iteration for the accurate solution of the convection diffusion equation, in W. Hackbusch and U. Trottenberg, Eds., ***Multi-grid Methods,*** Lecture Notes in Mathematics 960, Springer-Verlag, Berlin, 1982.

[14] L. Lamport, The parallel execution of DO loops, ***Comm. ACM.*** 17:83-93, 1974.

[15] J. A. Meijerink and H. A. van der Vorst, An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix, ***Math. Comp.*** 31:148-162, ***1977.***

[16] J. P. Milaszewicz, Improving Jacobi and Gauss-Seidel iterations, ***Lin. Alg. Appl.*** 93:161-170, 1987.

[17] K. W. Morton, Generalised Galerkin methods for steady and unsteady problems, in K. W. Morton and M. J. Baines, Eds., ***Numerical Methods for Fluid Dynamics,*** Academic Press, Orlando, 1982.

[18] S. V. Parter and M. Steuerwalt, Block iterative methods for elliptic and parabolic difference equations, ***SIAM J. Numer. Anal.*** 19:1173-1195, 1982.

[19] PCGPAK User's Guide, Version 1.04, Scientific Computing Associates, New Haven, CT, 1987.

[20] P. J. Roache, ***Computational Fluid Dynamics,*** Hermosa Publishers, Albuquerque, 1982.

[21] Y. Saad and M. H. Schultz, GMRES: A generalized minimual residual algorithm for solving nonsymmetric linear systems, ***SIAM J. Sci. Stat. Comput*** 7:856-869, ***1986.***

[22] J. Sheldon, On the spectral norms of several iterative processes, ***J. Assoc. Comput. Mach.*** 6:494-505, 1959.

[23] R. S. Varga, ***Matrix Iterative Analysis,*** Prentice-Hall, New Jersey, 1962.

[24] H. F. Weinberger, ***A First Course in Partial Differential Equations with Complex Variables and Transform Methods,*** Blaisdell, New York, 1965.

[25] Z. Woinicki, Two-sweep iterative methods for solving large linear systems and their application to the numerical solution of multi-group multi-dimensional neutron diffusion equation, Doctoral Dissertation, Report N^0 1447/CYFRONET/PM/A, Institute of Nuclear Research, Swierk, Poland, 1973.

[26] D. M. ***Young, Iterative Solution of Large Linear Systems,*** Academic Press, New York, 1971.