SIAM J. MATRIX ANAL. APPL.
Vol. 21, No. 1, pp. 29–44

REDUCED SYSTEMS FOR THREE-DIMENSIONAL ELLIPTIC EQUATIONS WITH VARIABLE COEFFICIENTS*

CHEN GREIF†

Abstract. We consider large sparse nonsymmetric linear systems arising from finite difference discretization of three-dimensional (3D) convection-diffusion equations with variable coefficients. We show that performing one step of cyclic reduction yields a system of equations which is well conditioned and for which fast convergence can be obtained. A certain block ordering strategy is applied, and analytical results concerning symmetrizability conditions and bounds on convergence rates are given. The analysis is accompanied by numerical examples.

Key words. cyclic reduction, 3D elliptic problems, variable coefficients

AMS subject classifications. 65F10, 65N22

PII. S0895479897329849

1. Introduction.

Consider the following three-dimensional (3D) convection-diffusion equation

$$-[(pu_x)_x + (qu_y)_y + (ru_z)_z] + su_x + tu_y + vu_z = w \ , \tag{1.1}$$

on a domain $\Omega \subset \mathbb{R}^3$, subject to Dirichlet, Neumann, or mixed boundary conditions, where all the functions in (1.1) are trivariate, and $p,\ q,\ r > 0$ on Ω. Several discretization schemes are possible. See Morton [11] for a comprehensive survey on numerical solution of the convection-diffusion problem. In this work we use a seven-point discretization technique as a starting point and extend the analysis of Elman and Golub [4], done for the two-dimensional (2D) variable coefficient case, and the analysis of Greif and Varah [8], [9] for the 3D problem with constant coefficients to the 3D problem with variable coefficients.

Let h denote the width of a uniform mesh. In the description that follows we use the notation $G_{i,j,k} \equiv G(ih, jh, kh)$, where G is a trivariate function. The seven-point discretization is done as follows (see, e.g., [4] for the analogous 2D case). For the first term in (1.1) we have

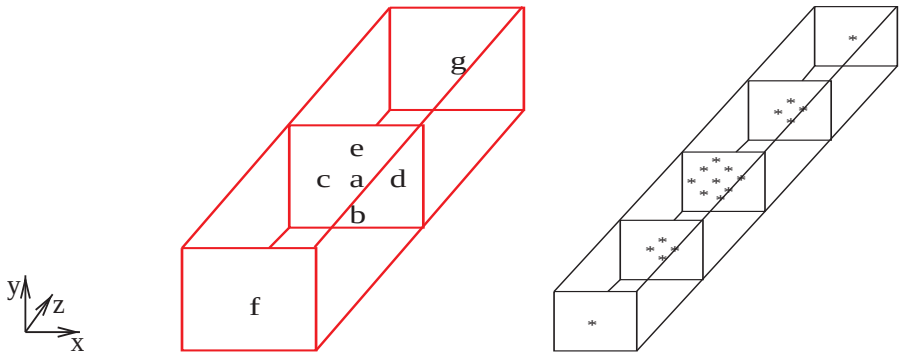
$$(pu_x)_x \approx \frac{p_{i+\frac{1}{2},j,k}u_{i+1,j,k} - (p_{i+\frac{1}{2},j,k} + p_{i-\frac{1}{2},j,k})u_{i,j,k} + p_{i-\frac{1}{2},j,k}u_{i-1,j,k}}{h^2} \ ,$$

and an analogous discretization is performed for $(qu_y)_y$ and $(ru_z)_z$. For the convective terms $su_x,\ tu_y$, and vu_z we use either upwind or centered difference schemes.

Let F denote the corresponding difference operator, scaled by h^2, and denote the values of the associated computational molecule by $a_{i,j,k},\ b_{i,j,k},\ c_{i,j,k},\ d_{i,j,k},\ e_{i,j,k},\ f_{i,j,k}$, and $g_{i,j,k}$, in the following manner: if (i,j,k) is a gridpoint not next to the boundary, then

$$\begin{aligned} F\,u_{i,j,k} = {} & a_{i,j,k}\,u_{i,j,k} + b_{i,j,k}\,u_{i,j-1,k} + c_{i,j,k}\,u_{i-1,j,k} + d_{i,j,k}\,u_{i+1,j,k} \\ & + e_{i,j,k}\,u_{i,j+1,k} + f_{i,j,k}\,u_{i,j,k-1} + g_{i,j,k}\,u_{i,j,k+1} \ . \end{aligned} \tag{1.2}$$

*Received by the editors November 10, 1997; accepted for publication (in revised form) by Z. Strakoš August 20, 1998; published electronically August 3, 1999.
http://www.siam.org/journals/simax /21-1/32984.html
†Department of Computer Science, The University of British Columbia, Vancouver, BC, V6T 1Z4 Canada (greif@cs.ubc.ca).



(**a**) seven-point operator (**b**) reduced operator

FIG. 1.1. *Computational molecules of the unreduced and the reduced operators.*

The computational molecule is graphically illustrated in Figure 1.1(a) (in the figure the subscripts are dropped).

If centered differences are used to discretize the convective terms, the values of the computational molecule are given by

$$
\begin{aligned}
(1.3)\qquad a_{i,j,k} &= p_{i+\frac{1}{2},j,k} + p_{i-\frac{1}{2},j,k} + q_{i,j+\frac{1}{2},k} + q_{i,j-\frac{1}{2},k} + r_{i,j,k+\frac{1}{2}} + r_{i,j,k-\frac{1}{2}};\\
b_{i,j,k} &= -q_{i,j-\frac{1}{2},k} - \frac{t_{i,j,k}h}{2}; \quad e_{i,j,k} = -q_{i,j+\frac{1}{2},k} + \frac{t_{i,j,k}h}{2};\\
c_{i,j,k} &= -p_{i-\frac{1}{2},j,k} - \frac{s_{i,j,k}h}{2}; \quad d_{i,j,k} = -p_{i+\frac{1}{2},j,k} + \frac{s_{i,j,k}h}{2};\\
f_{i,j,k} &= -r_{i,j,k-\frac{1}{2}} - \frac{v_{i,j,k}h}{2}; \quad g_{i,j,k} = -r_{i,j,k+\frac{1}{2}} + \frac{v_{i,j,k}h}{2}.
\end{aligned}
$$

If one uses upwind schemes, then the type of scheme depends on the sign of the convective terms. Assuming that s, t, and v do not change sign in the domain, if they are positive one can use the backward scheme, and if they are negative one can use the forward scheme. Discretizing using backward differences yields

$$
\begin{aligned}
(1.4)\qquad a_{i,j,k} &= p_{i+\frac{1}{2},j,k} + p_{i-\frac{1}{2},j,k} + q_{i,j+\frac{1}{2},k} + q_{i,j-\frac{1}{2},k} + r_{i,j,k+\frac{1}{2}} + r_{i,j,k-\frac{1}{2}} + s_{i,j,k}h\\
&\quad + t_{i,j,k}h + v_{i,j,k}h;\\
b_{i,j,k} &= -q_{i,j-\frac{1}{2},k} - t_{i,j,k}h; \quad e_{i,j,k} = -q_{i,j+\frac{1}{2},k};\\
c_{i,j,k} &= -p_{i-\frac{1}{2},j,k} - s_{i,j,k}h; \quad d_{i,j,k} = -p_{i+\frac{1}{2},j,k};\\
f_{i,j,k} &= -r_{i,j,k-\frac{1}{2}} - w_{i,j,k}h; \quad g_{i,j,k} = -r_{i,j,k+\frac{1}{2}},
\end{aligned}
$$

and for forward differences (1.4) needs to be modified in an obvious manner.

The sparsity structure of the matrix representing the system of equations depends on the ordering of the unknowns. A common strategy is the red/black ordering, which is depicted in Figure 1.2: the gridpoints are colored using two colors in a checkerboard fashion, and the points that correspond to one of the colors (say, red) are numbered first. In this case the corresponding matrix can be written as

$$
(1.5)\qquad \begin{pmatrix} B & C \\ D & E \end{pmatrix} \begin{pmatrix} u^{(r)} \\ u^{(b)} \end{pmatrix} = \begin{pmatrix} w^{(r)} \\ w^{(b)} \end{pmatrix},
$$

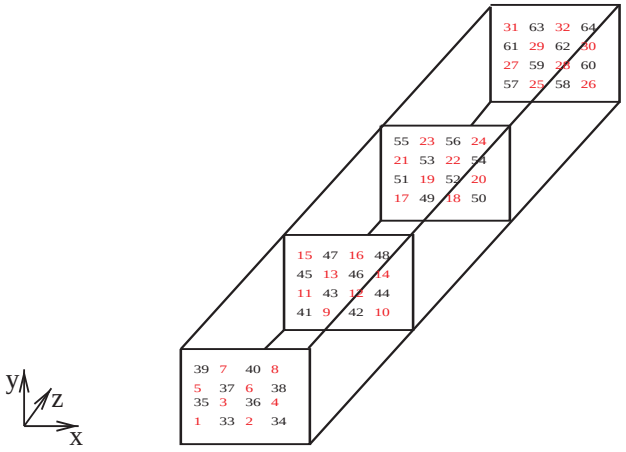


FIG. 1.2. *Red/black ordering of the 3D grid.*

where both B and E are diagonal. In (1.5) superscripts (r) and (b) are attached to denote the associated colors. A simple process of block Gaussian elimination leads to a smaller system, for the black points only, which is called a *reduced system* [2]:

$$[E - DB^{-1}C]u^{(b)} = w^{(b)} - DB^{-1}w^{(r)} \ . \tag{1.6}$$

Since B is diagonal, the matrix of (1.6) is sparse. In the 3D case the corresponding difference operator has a computational molecule which consists of 19 points, as illustrated in Figure 1.1(b). Once the solution for the black points is computed, the solution for the red points corresponds to solving a diagonal system and thus is readily obtained. Moving from system (1.5) to system (1.6) amounts to performing *one step of cyclic reduction* [2]. This procedure can be repeated until a small system of equations is obtained, which can be solved directly. An overview of the idea of cyclic reduction and several references are given in [5, pp. 177–180].

The elimination of half of the unknowns is accompanied by permutation of the matrix (equivalently, reordering of the unknowns). Once the permuted reduced system is formed, an iterative method can be used to find the solution. The procedure of performing one step of cyclic reduction for a non-self-adjoint problem and solving the resulting system using an iterative solver was extensively investigated by Elman and Golub for 2D problems [2], [3], [4]. They showed that one step of cyclic reduction leads to systems with valuable properties, such as symmetrizability by a real diagonal nonsingular matrix for a large set of the underlying PDE coefficients (which is effectively used to derive bounds on the convergence rates of iterative solvers), and fast convergence. In [2] the univariate case, which is naturally more transparent, is used to illustrate the advantages of this technique. Many of the highly effective techniques presented and used by Elman and Golub can be generalized to the 3D case and will be mentioned throughout this paper.

An outline of the rest of this paper follows. In section 2 we introduce the cyclically reduced operator. In section 3 we discuss block orderings and present a family of orderings for the reduced grid. In section 4 we present symmetrization results. In section 5 we use the results of section 4 to derive bounds on convergence rates for block stationary methods. In section 6 we present numerical examples, which include solving the systems using Krylov subspace solvers. Finally, in section 7 we draw some conclusions.

2. The reduced operator. One step of cyclic reduction for a 3D model problem with constant coefficients has been described in [9], where full details on the construction of the matrix are given. Convergence analysis and techniques for solving the resulting system of equations using block stationary methods are described in [8], [9]. In the case of constant coefficients, the values of the computational molecule associated with a given gridpoint do not depend on the point's coordinates, as opposed to the variable coefficient case. Nevertheless, the sparsity structure of the reduced matrix is identical in both cases.

The explicit difference equation associated with the reduced operator for the 3D variable coefficient case can be found in [7] and should be used for constructing the reduced matrix. The alternative of performing the matrix products in (1.6) might be significantly more costly, especially in the 3D case, and in particular, in programming environments where vectorization is crucial (e.g., Matlab).

Consider the following constant coefficient model problem: $p(x,y,z) = q(x,y,z) = r(x,y,z) \equiv 1,\ s(x,y,z) = \sigma,\ t(x,y,z) = \tau,\ v(x,y,z) = \mu$. After scaling by ah^2, the difference equation has the form

$$\begin{aligned}(2.1)\qquad \tilde{R}\,u_{i,j,k} = {} & (a^2 - 2be - 2cd - 2fg)\,u_{i,j,k} - f^2\,u_{i,j,k-2} - 2ef\,u_{i,j+1,k-1} \\ & -2cf\,u_{i-1,j,k-1} - 2df\,u_{i+1,j,k-1} - 2bf\,u_{i,j-1,k-1} - e^2\,u_{i,j+2,k} \\ & -2de\,u_{i+1,j+1,k} - c^2\,u_{i-2,j,k} - d^2\,u_{i+2,j,k} - 2bc\,u_{i-1,j-1,k} \\ & -b^2\,u_{i,j-2,k} - 2eg\,u_{i,j+1,k+1} - 2cg\,u_{i-1,j,k+1} - 2ce\,u_{i-1,j+1,k} \\ & -2bd\,u_{i+1,j-1,k} - 2dg\,u_{i+1,j,k+1} - 2bg\,u_{i,j-1,k+1} - g^2\,u_{i,j,k+2}\ .\end{aligned}$$

Denote the continuous operator corresponding to this model problem

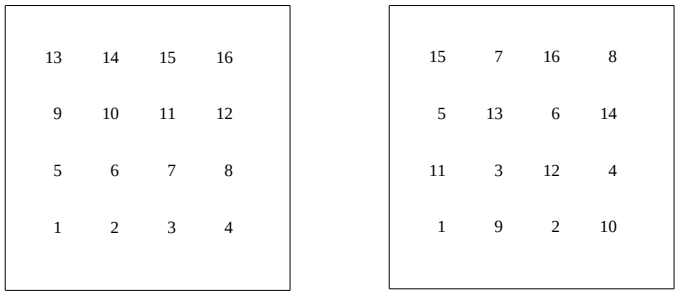
$$(2.2)\qquad L = -\Delta + (\sigma,\tau,\mu)^T\nabla\ .$$

The reduced operator can be derived directly as a discretization scheme of the original PDE, with $O(h^2)$ correction terms in the case of centered difference discretization and $O(h)$ correction terms if upwind discretization is used. This can be done by means analogous to the techniques used for the 2D case (see Elman and Golub [3] and Golub and Tuminaro [6]). Consider the centered difference discretization. Expanding (2.2) in a multivariate Taylor expansion about the gridpoint (ih, jh, kh) yields, after scaling by $2ah^2$,

$$\begin{aligned}(2.3)\qquad 2\tilde{R}\,u = {} & L\,u - \frac{1}{6}\,h^2\,u_{yyyy} - \frac{1}{6}\,h^2\,u_{yyzz} - \frac{1}{6}\,h^2\,\tau\,\mu\,u_{yz} + \frac{1}{6}\,h^2\,\mu\,u_{yyz} + \frac{1}{6}\,h^2\,\sigma u_{xzz} \\ & + \frac{1}{6}\,h^2\,\tau\,u_{xxy} + \frac{1}{6}\,h^2\,\sigma\,u_{xyy} + \frac{1}{3}\,h^2\,\sigma\,u_{xxx} - \frac{1}{6}\,h^2\,\sigma\,\mu\,u_{xz} - \frac{1}{6}\,h^2\,\sigma\,\tau\,u_{xy} + \frac{1}{3}\,h^2\,\mu\,u_{zzz} \\ & - \frac{1}{12}\,h^2\,\mu^2\,u_{zz} + \frac{1}{3}\,h^2\,\tau\,u_{yyy} - \frac{1}{12}\,h^2\,\tau^2\,u_{yy} + \frac{1}{6}\,h^2\,\tau\,u_{yzz} - \frac{1}{6}\,h^2\,u_{xxyy} - \frac{1}{6}\,h^2\,u_{xxzz} \\ & + \frac{1}{6}\,h^2\,\mu\,u_{xxz} - \frac{1}{12}\,h^2\,\sigma^2\,u_{xx} - \frac{1}{6}\,h^2\,u_{xxxx} - \frac{1}{6}\,h^2\,u_{zzzz} + O(h^3)\ .\end{aligned}$$

The above computation was carried out using Maple V. The $O(h^2)$ terms contain, among other terms, the expression $-\frac{h^2}{12}(\sigma^2 u_{xx} + \tau^2 u_{yy} + \mu^2 u_{zz})$, which can be thought of as addition of artificial viscosity to the original equation. The reduced right-hand side is equal to $w_{i,j,k}$ with an $O(h^2)$ error. Gaussian elimination yields the following right-hand side,

$$w_{i,j,k} - \frac{b}{a}\,w_{i,j-1,k} - \frac{c}{a}\,w_{i-1,j,k} - \frac{d}{a}\,w_{i+1,j,k} - \frac{e}{a}\,w_{i,j+1,k} - \frac{f}{a}\,w_{i,j,k-1} - \frac{g}{a}\,w_{i,j,k+1}\ ,$$



(**a**) Natural (**b**) Red/black

FIG. 3.1. *Two possible orderings of the block grid. Each point in these grids corresponds to a one-dimensional (1D) block of gridpoints in the underlying 3D grid.*

whose Taylor expansion about the gridpoint (ih, jh, kh), after scaling by $2ah^2$, is $w - \frac{h^2}{12}[\,-\Delta w + (\sigma w_x + \tau w_y + \mu w_z)\,]$, evaluated at the point (ih, jh, kh). This is another similarity with the 2D case [3].

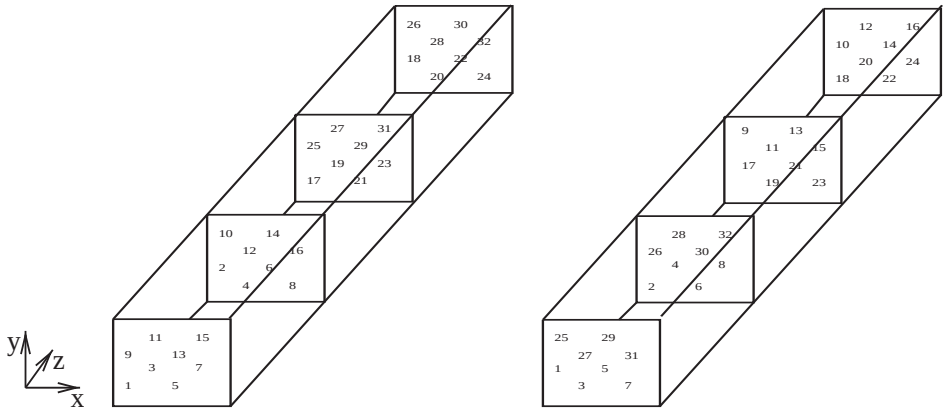
3. Block ordering strategies for the reduced grid. The question of ordering is of major importance, as a good ordering strategy can lead to fast convergence. An excellent overview of the literature that deals with ordering strategies is found in a recent report by Benzi, Szyld, and van Duin [1]. For 3D problems it seems useful to consider the ordering of *blocks* of unknowns, rather than "pointwise" ordering. Such a strategy could be particularly useful for the cyclically reduced problem, as the reduced grid is somewhat irregular. Instead of ordering the unknowns directly in the 3D reduced grid, the problem of ordering is divided into two parts. First, define blocks of gridpoints and order them in a tensor-product 2D "block grid." Once the ordering of the blocks is determined, the task of the "inner" ordering in each of the blocks is relatively simple.

We can define an x-oriented "1D block of gridpoints" by referring to a set of gridpoints whose collection of all x-coordinate values include all the possible values $\{jh\}$ on the grid. A simple example is a single horizontal line of gridpoints in a tensor-product 3D grid. Similarly, y-oriented and z-oriented 1D blocks can be defined. Once the 1D blocks of gridpoints are defined, a *block computational molecule* can be defined as follows.

DEFINITION 3.1. *For a certain given 3D grid and a 1D block of gridpoints in it, the associated* block computational molecule *is defined as the computational molecule in the corresponding block grid. That is, its components are the 1D blocks in the block grid, each of which contains at least one gridpoint which belongs to the (point) computational molecule associated with the 3D problem.*

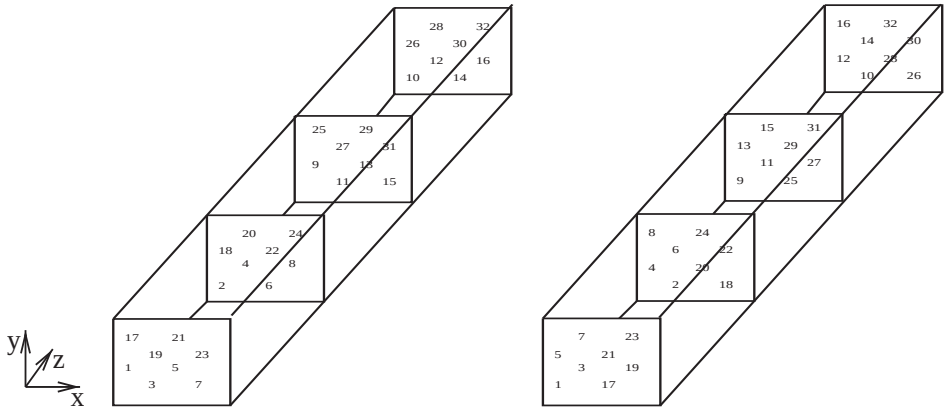
Using the above, we can now easily define different families and types of orderings. For example, a certain ordering strategy is a *natural* block ordering strategy relative to the 1D blocks of gridpoints if these blocks are ordered in the block grid using natural lexicographic ordering. Similarly, one can define a red/black block ordering strategy, and so on (see Figure 3.1).

Below we focus on a particular family of orderings for the reduced grid, which we call the *two-plane* ordering. This ordering corresponds to defining each of the 1D blocks of gridpoints as a collection of $2n$ gridpoints from two horizontal lines and two adjacent planes (here n is the number of gridpoints in a single line in the original



(**a**) 2PNxy (**b**) 2PRBxy

FIG. 3.2. *Two types of two-plane ordering.*

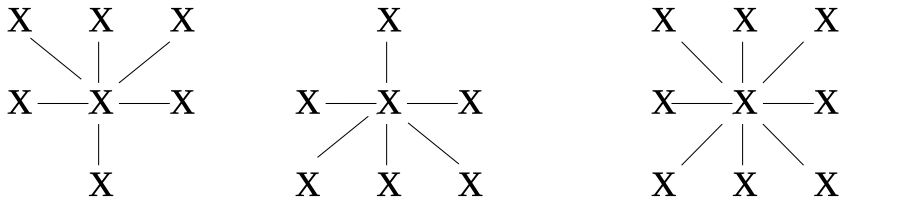


(**a**) 2PNxz (**b**) 2PNyz

FIG. 3.3. *Possible orientations of the* 1*D blocks of gridpoints in the set of natural two-plane orderings.*

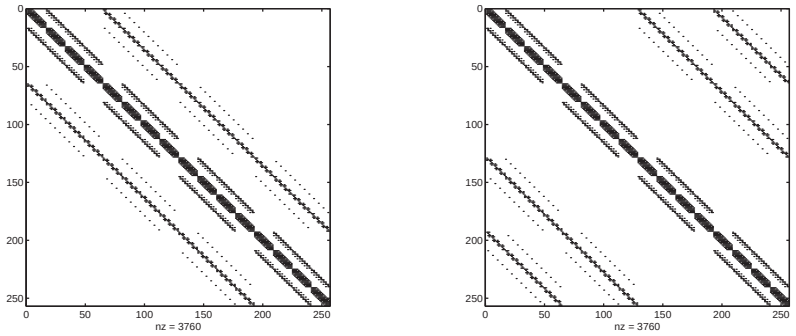
unreduced grid). A single member of this family was introduced in [9]. In Figure 3.2 two members of the family are depicted: natural two-plane ordering and red/black two-plane ordering. For notational convenience, we label them "2PN" and "2PRB," respectively. Two additional letters are added in order to distinguish between different orientations of the $2n$-item 1D blocks of gridpoints. Let us illustrate this for the specific case depicted in Figure 3.2(a). Here $n = 4$. Indices 1–8, 9–16, 17–24, and 25–32 are each an x-oriented 1D block. The block grid is of size 2×2 and its components are ordered in natural lexicographic fashion. Each of the sets of indices 1–16 and 17–32 forms an x-y–oriented pair of planes. Hence the name 2PNxy. In Figure 3.3 other orientations of the blocks in the natural two-plane ordering are depicted.

Figures 3.4(a) and (b) illustrate what blocks are associated with a single gridpoint. In these figures, each "X" corresponds to a 1D block which contains at least one gridpoint in the "point" computational molecule associated with the selected gridpoint (the one that is at the center of the computational molecule). As is evident, the



(**a**) even $k = \frac{z}{h}$ (**b**) odd $k = \frac{z}{h}$ (**c**) computational molecule

FIG. 3.4. *Block computational molecule associated with the two-plane ordering.*



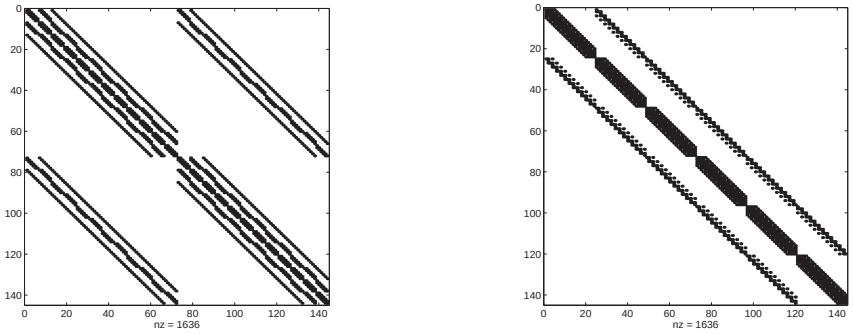
(**a**) natural (**b**) red/black based on 2D blocks

FIG. 3.5. *Sparsity structures of two matrices which belong to the family of two-plane orderings.*

structure depends on the parity of this gridpoint's index. The block computational molecule (Figure 3.4(c)) is obtained by taking the union of all the 1D blocks associated with each of the gridpoints in the block, and thus it is identical to the computational molecule of the classical nine-point operator. This allows one to conclude, e.g., that the reduced matrix does not have block property A [14] relative to partitioning into $2n \times 2n$ blocks. On the other hand, applying a 4-color scheme to the blocks of gridpoints can be effective for parallelization.

The same ideas as above can be applied to $2D$ *blocks* of gridpoints. The block grid in this case is univariate. The reduced matrix is consistently ordered relative to partitioning associated with 2D blocks. In Figure 3.5 the sparsity structures of 256×256 matrices corresponding to natural two-plane ordering and red/black block ordering relative to 2D blocks are depicted.

In order to illustrate the effectiveness of the two-plane ordering, we present in Figure 3.6 a single 2D block of a natural two-plane matrix vs. one that corresponds to ("point") natural lexicographic ordering. The matrices in the figure are associated with a $12 \times 12 \times 12$ grid. As is evident, the main diagonal block of the two-plane matrix is more dense. Compared to the natural lexicographic ordering, there are significantly more nonzero entries in the block diagonal submatrix whose bandwidth does not depend on n. As a result, direct preconditioner solves will be more efficient due to less fill-in. If a stationary scheme such as block Jacobi is used, by [13, Thm. 3.15] it is guaranteed that if the reduced matrix is an M-matrix, then the convergence of a scheme associated with the two-plane matrix is faster than that of a scheme with



(**a**) lexicographic (**b**) natural two-plane

FIG. 3.6. *A zoom on 2D blocks of the matrices corresponding to two ordering strategies of the reduced grid.*

the lexicographic ordering. (The circumstances in which the reduced matrix is an M-matrix in the constant coefficient case are discussed in [9].)

4. Symmetrization of the reduced matrix. In order to obtain bounds on convergence rates for block stationary methods, we consider the following technique, suggested and used in [2], [3], [4] and also effectively applied in [8], [9]: if there exists a real diagonal matrix Q, such that $\hat{S} = Q^{-1}SQ$ is symmetric, then for a splitting $S = D - C$ and an analogous splitting of $\hat{S}$, namely $\hat{S} = Q^{-1}DQ - Q^{-1}CQ \equiv \hat{D} - \hat{C}$, we have

$$\rho(D^{-1}C) = \rho(\hat{D}^{-1}\hat{C}) \leq ||\hat{D}^{-1}||_2||\hat{C}||_2 = \frac{\rho(\hat{C})}{\lambda_{min}(\hat{D})} \ . \tag{4.1}$$

A bound for the spectral radius of the iteration matrix can thus be obtained by evaluating the expression on the right-hand side of (4.1). Denote the entries of S by $\{s_{i,j}\}_{i,j=1}^{n^3/2}$. Since S is sparse and has a block structure, a small amount of work is needed in order to find Q—by requiring for the entries of $\hat{S}$, which we denote by $\{\hat{s}_{i,j}\}_{i,j=1}^{n^3/2}$, that $\hat{s}_{i,j} = \hat{s}_{j,i}$. Since matrices that correspond to different orderings are merely symmetric permutations of one another, we can pick a matrix that corresponds to a specific ordering strategy and do all the work for it. This will result in obtaining general symmetrization conditions for the reduced matrices (regardless of the ordering used). Thus we pick the specific ordering strategy 2PNxz.

Let q_ℓ denote the ℓth diagonal entry in Q. Then

$$\hat{s}_{j,\ell} = \frac{s_{j,\ell}q_\ell}{q_j} \ , \qquad 1 \leq j, \ell \leq \frac{n^3}{2} \ , \tag{4.2}$$

and the symmetry conditions can be expressed as

$$\frac{q_\ell}{q_j} = \sqrt{\frac{s_{\ell,j}}{s_{j,\ell}}} \ , \qquad 1 \leq \ell \leq \frac{n^3}{2}, \quad 1 \leq j \leq \ell \ . \tag{4.3}$$

It is sufficient to look at $2n \times 2n$ blocks. We start with the main diagonal blocks. We have to examine all the entries of the main block that appear in the ℓth row of

the matrix, namely $s_{\ell,\ell-4}, s_{\ell,\ell-3}, \ldots, s_{\ell,\ell}, \ldots, s_{\ell,\ell+4}$. For $s_{\ell,\ell-4}$, if $\ell \bmod 2n \geq 5$ or is equal to 0, then $\ell - 4$ corresponds to the $(i-2, j, k)$ mesh point. Thus

$$s_{\ell-4,\ell} = - \left. \frac{d_{\tilde{\alpha}+1,\tilde{\beta},\tilde{\gamma}} d_{\tilde{\alpha},\tilde{\beta},\tilde{\gamma}}}{a_{\tilde{\alpha}+1,\tilde{\beta},\tilde{\gamma}}} \right|_{\tilde{\alpha}=i-2,\tilde{\beta}=j,\tilde{\gamma}=k}, \tag{4.4}$$

and from this it follows that

$$\left(\frac{q_\ell}{q_{\ell-4}}\right)^2 = \frac{-\left(\frac{c_{i-1,j,k}c_{i,j,k}}{a_{i-1,j,k}}\right)}{-\left(\frac{d_{i-2,j,k}d_{i-1,j,k}}{a_{i-1,j,k}}\right)} = \frac{c_{i-1,j,k}c_{i,j,k}}{d_{i-2,j,k}d_{i-1,j,k}} \,. \tag{4.5}$$

In this case the values associated with the center of the computational molecule (namely, $a_{i,j,k}$) are canceled, but this happens only for rows that involve the $(i \pm 2, j, k)$, $(i, j \pm 2, k)$, and $(i, j, k \pm 2)$ gridpoints. Applying the same procedure to the rest of the entries of the main diagonal block, we obtain the following:

$$\left(\frac{q_\ell}{q_{\ell-3}}\right)^2 = \frac{\frac{c_{i,j,k-1}f_{i,j,k}}{a_{i,j,k-1}} + \frac{c_{i,j,k}f_{i-1,j,k}}{a_{i-1,j,k}}}{\frac{g_{i,j,k-1}d_{i-1,j,k-1}}{a_{i,j,k-1}} + \frac{g_{i-1,j,k-1}d_{i-1,j,k}}{a_{i-1,j,k}}} \quad \ell \bmod 2n = 0, 4, 6, \ldots, 2n-2; \tag{4.6}$$

$$\left(\frac{q_\ell}{q_{\ell-2}}\right)^2 = \frac{\frac{c_{i,j+1,k}e_{i,j,k}}{a_{i,j+1,k}} + \frac{c_{i,j,k}e_{i-1,j,k}}{a_{i-1,j,k}}}{\frac{b_{i,j+1,k}d_{i-1,j+1,k}}{a_{i,j+1,k}} + \frac{d_{i-1,j,k}b_{i-1,j+1,k}}{a_{i-1,j,k}}} \qquad \ell \bmod 4 = 2 \text{ or } 3; \tag{4.7}$$

$$\left(\frac{q_\ell}{q_{\ell-2}}\right)^2 = \frac{\frac{b_{i-1,j,k}c_{i,j,k}}{a_{i-1,j,k}} + \frac{c_{i,j-1,k}b_{i,j,k}}{a_{i,j-1,k}}}{\frac{d_{i-1,j,k}e_{i-1,j-1,k}}{a_{i-1,j,k}} + \frac{e_{i,j-1,k}d_{i-1,j-1,k}}{a_{i,j-1,k}}} \qquad \ell \bmod 4 = 0 \text{ or } 1; \tag{4.8}$$

$$\left(\frac{q_\ell}{q_{\ell-1}}\right)^2 = \frac{\frac{g_{i-1,j,k}c_{i,j,k}}{a_{i-1,j,k}} + \frac{c_{i,j,k+1}g_{i,j,k}}{a_{i,j,k+1}}}{\frac{d_{i-1,j,k}f_{i-1,j,k+1}}{a_{i-1,j,k}} + \frac{d_{i-1,j,k+1}f_{i,j,k+1}}{a_{i,j,k+1}}} \quad \ell \bmod 2n = 3, \ldots, 2n-1; \tag{4.9}$$

$$\left(\frac{q_\ell}{q_{\ell-1}}\right)^2 = \frac{\frac{e_{i,j,k-1}f_{i,j,k}}{a_{i,j,k-1}} + \frac{f_{i,j+1,k}e_{i,j,k}}{a_{i,j+1,k}}}{\frac{g_{i,j,k-1}b_{i,j+1,k-1}}{a_{i,j,k-1}} + \frac{b_{i,j+1,k}f_{i,j+1,k-1}}{a_{i,j+1,k}}} \qquad \ell \bmod 4 = 2; \tag{4.10}$$

$$\left(\frac{q_\ell}{q_{\ell-1}}\right)^2 = \frac{\frac{b_{i,j,k-1}f_{i,j,k}}{a_{i,j,k-1}} + \frac{f_{i,j-1,k}b_{i,j,k}}{a_{i,j-1,k}}}{\frac{g_{i,j,k-1}e_{i,j-1,k-1}}{a_{i,j,k-1}} + \frac{e_{i,j-1,k}g_{i,j-1,k-1}}{a_{i,j-1,k}}} \qquad \ell \bmod 4 = 0. \tag{4.11}$$

As is evident, (4.5)–(4.11) overdetermine the nonzero values of the matrix Q. Indeed, (4.9)–(4.11) are sufficient to determine all the diagonal entries, except the first entry in each $2n \times 2n$ block, which at this stage can be arbitrarily chosen. We have to make sure, therefore, that (4.5)–(4.8) are consistent with these three equations, and this requirement imposes some additional conditions. In the constant coefficient case there is unconditional consistency. The problematic nature of the variable coefficient case can be demonstrated simply by looking at one of the consistency conditions.

Consider a gridpoint (i,j,k) whose associated index, ℓ, satisfies $\ell \bmod 4 = 0$. Applying (4.9) to $\ell - 1$ means looking at the row corresponding to the $(i, j-1, k-1)$ gridpoint, and multiplying (4.9), applied to $\ell - 1$, by (4.11) results in an equation for $\frac{q_\ell}{q_{\ell-2}}$, which should be consistent with (4.8). There are three additional consistency conditions for the main block and then eight additional conditions for the rest of the blocks of the reduced matrix. In the consistency condition, if we equate variables that belong to the same location in the computational molecule, we find that sufficient conditions for the above-mentioned consistency conditions to hold are $b_{i-1,j,k} = b_{i,j,k-1},\ c_{i,j-1,k-1} = c_{i,j,k},\ d_{i-1,j-1,k-1} = d_{i-1,j,k},\ e_{i,j-1,k-1} = e_{i-1,j-1,k},\ f_{i,j,k} = f_{i-1,j-1,k},\ g_{i,j,k-1} = g_{i-1,j-1,k-1}$. If $b_{i,j,k} = b_j$ for all $i,\ k$, and similarly $c_{i,j,k} = c_i,\ d_{i,j,k} = d_i,\ e_{i,j,k} = e_j,\ f_{i,j,k} = f_k,\ g_{i,j,k} = g_k$, the consistency condition becomes

$$\frac{c_i g_{k-1}}{f_k d_{i-1}} \cdot \frac{b_j f_k}{e_{j-1} g_{k-1}} = \frac{b_j c_i}{d_{i-1} e_{j-1}} \ , \tag{4.12}$$

which is obviously satisfied. The actual meaning of these conditions is that the continuous problem is separable.

The analysis for off-diagonal blocks is identical, and the following additional conditions are obtained:

$$\left(\frac{q_\ell}{q_{\ell-2n}}\right)^2 = \frac{f_{i,j,k} f_{i,j,k-1}}{g_{i,j,k-1} g_{i,j,k-2}} \quad , \qquad 2n \le \ell \le \frac{n^3}{2} \ ; \tag{4.13}$$

$$\left(\frac{q_\ell}{q_{\ell-n^2}}\right)^2 = \frac{b_{i,j-1,k} b_{i,j,k}}{e_{i,j-1,k} e_{i,j-2,k}} \quad , \qquad n^2 \le \ell \le \frac{n^3}{2} \ . \tag{4.14}$$

The two equations above determine the rest of the entries of the matrix, and only the first entry in the symmetrizer can be determined arbitrarily.

Last, in order for the symmetrizer to be real, we must require that the products $c_i d_{i-1},\ b_j e_{j-1}$, and $f_k g_{k-1}$ have the same sign.

All that has been said can be summarized in the following theorem, which demonstrates another point of similarity between the 2D [4] and the 3D problems with variable coefficients.

THEOREM 4.1. *Suppose the operator of* (1.1) *is separable. If* $c_i d_{i-1},\ b_j e_{j-1}$, *and* $f_k g_{k-1}$ *are all nonzero and have the same sign for all* $i,\ j$, *and* k, *then there exists a real nonsingular diagonal matrix* Q *such that* $Q^{-1}SQ$ *is symmetric.*

The symmetrized computational molecule can be derived *without* actually performing the similarity transformation. For example, the symmetrized value corresponding to $-\frac{c_{i,j,k} c_{i-1,j,k}}{a_{i-1,j,k}}$ is $-\sqrt{\frac{c_{i-1,j,k} c_{i,j,k} d_{i-2,j,k} d_{i-1,j,k}}{a_{i-1,j,k}}}$, and so on. The symmetrization operation should not actually be performed in order to solve the linear system, as the symmetrizing matrix has entries that are unbounded as h goes to zero. The symmetrization should be done for the mere purpose of convergence analysis.

5. Bounds on convergence rates for block stationary methods. The reduced matrix which corresponds to the family of natural two-plane orderings is of size $(n^3/2) \times (n^3/2)$ and can be thought of as a block tridiagonal matrix, relative to $n^2 \times n^2$ blocks:

$$S = tri[S_{j,j-1}, S_{j,j}, S_{j,j+1}] \ . \tag{5.1}$$

$S_{i,j}$ are block tridiagonal matrices with respect to $2n \times 2n$ blocks.

In [8] two partitionings are considered: a partitioning into 1D blocks ($2n \times 2n$ blocks) and a partitioning into 2D blocks (of size $n^2 \times n^2$). In order to find a bound, if symmetrization is possible, then the strategy in [4] can be applied. In Theorem 5.1, we refer to the ordering strategy 2PNxz. Since other orderings are symmetric permutations of 2PNxz, finding the bounds for other ordering strategies discussed in this paper is straightforward.

THEOREM 5.1. *Suppose the continuous problem is separable and $c_{i+1}d_i, b_{j+1}e_j$, and $f_{k+1}g_k$ are all positive and bounded by β_x, β_y, and β_z, respectively. Suppose also that $a_{i,j,k} \geq \alpha$ for all i, j, and k. Denote $\tilde{h} = \frac{1}{\frac{n}{2}+1}$. Then the spectral radii of the iteration matrices associated with the block Jacobi scheme which correspond to 1D splitting and 2D splitting are bounded by $\frac{\phi+\xi}{\eta}$ and $\frac{\phi}{\eta-\xi}$, respectively, where η, ξ, and ϕ are defined as follows:*

$$\eta = \alpha^2 - 2\beta_y - 2\beta_z - 2\sqrt{\beta_y\beta_z} - 4(\sqrt{\beta_x\beta_y} + \sqrt{\beta_x\beta_z})\cos(\pi h) - 4\beta_x \cos^2(\pi h); \tag{5.2a}$$

$$\xi = 2\beta_z \cos(\pi\tilde{h}) + \sqrt{4\beta_y\beta_z + 16\beta_x\beta_z\cos^2(\pi h) + 16\beta_z\sqrt{\beta_x\beta_y}\cos(\pi h)}\,; \tag{5.2b}$$

$$\phi = 4\sqrt{\beta_y\beta_z} + 4\sqrt{\beta_x\beta_y}\cdot\cos(\pi h) + 2\beta_y\cos(\pi\tilde{h})\,. \tag{5.2c}$$

Proof. The proof follows by using the technique of [4, pp. 346–347]. The conditions stated in the theorem guarantee that the matrix is symmetrizable. Denote the reduced matrix by S and the symmetrized matrix by $\hat{S}$. Suppose S^* is obtained by modifying $\hat{S}$ in the following manner: replace each occurrence of c_i and d_i by $\sqrt{\beta_x}$, replace each occurrence of b_j and e_j by $\sqrt{\beta_y}$, replace each occurrence of f_k and g_k by $\sqrt{\beta_z}$, and replace each occurrence of $a_{i,j,k}$ by α. Denote by $S^* = D^* - C^*$ the splitting, which is analogous, as far as sparsity structure is concerned, to the splitting $S = D - C$. For the 1D splitting, the matrix D^* is block diagonal with semibandwidth 4, its sparsity structure is identical to that of $\hat{D}$, and moreover, it is componentwise smaller than or equal to the entries of $\hat{D}$. By [9, Lem. 3.3], D^* is an irreducible diagonally dominant M-matrix.

The matrix C^* is nonnegative and satisfies $C^* \geq \hat{C}$. Thus the Perron–Frobenius theorem [13, p. 30] can be used to obtain an upper bound on the convergence rate for this splitting. Since the matrix S^* can now be referred to as a symmetrized version of a matrix that is associated with a constant coefficient case, the bound on the convergence rate is readily obtained from [9, Thm. 3.15]. For the 2D splitting the procedure is completely analogous and the bound is obtained from [8, Thm. 3.6]. ∎

We remark that estimates for the convergence rates of block Gauss–Seidel and block SOR schemes can be obtained by using the "near property A" analysis presented in [7], [8].

6. Numerical experiments. In the examples that follow, we begin with some results which validate the convergence analysis of section 5 for stationary methods. We then compare the performance of Krylov subspace solvers for the reduced and the unreduced systems. The experiments were performed on an SGI Origin 2000 machine. The program is written in Matlab 5.

6.1. Test problem 1. Consider the separable problem

$$-\Delta u + p_1\, x\, u_x + p_2\, y\, u_y + p_3\, z\, u_z = w(x, y, z) \tag{6.1}$$

TABLE 6.1
Comparison between the computed spectral radii of the block Jacobi iteration matrices and the bounds, using centered differences for the two splittings, with $p_1 = p_2 = p_3 = 1$.

Splitting		1D			2D		
n	n^3	ρ	Bound	Ratio	ρ	Bound	Ratio
8	512	0.793	0.894	1.13	0.682	0.826	1.21
12	1728	0.895	0.946	1.06	0.825	0.908	1.10
16	4096	0.937	0.968	1.03	0.892	0.944	1.06
20	8000	0.958	0.979	1.02	0.927	0.962	1.04
24	13824	0.970	0.985	1.02	0.948	0.973	1.03

on $\Omega = (0,1) \times (0,1) \times (0,1)$, with Dirichlet boundary conditions, where $w(x,y,z)$ is constructed so that the solution is

$$u(x,y,z) = x\ y\ z\ (1-x)\ (1-y)\ (1-z)\ \exp(x+y+z)\ . \tag{6.2}$$

For notational convenience, let $\gamma = \frac{p_1 h}{2}$, $\delta = \frac{p_2 h}{2}$, and $\mu = \frac{p_3 h}{2}$. Suppose h is sufficiently small and centered difference discretization is performed. Then

$$c_{i+1} d_i = (1+\gamma x_{i+1})(1-\gamma x_i) = 1 + \gamma h - \gamma^2 h^2 (i^2 + i)\ , \tag{6.3}$$

hence

$$1 + \gamma h - \gamma^2 (1-h) \le c_{i+1} d_i \le 1 + \gamma h - 2\gamma^2 h^2\ . \tag{6.4}$$

If $-1 < \gamma < \frac{1}{1-h}$, then $c_{i+1} d_i > 0$. For $b_{j+1} e_j$ and $f_{k+1} g_k$ the bounds are obtained in an identical manner. The center of the computational molecule is $a = 6$. In terms of the PDE coefficients, the condition on γ means that the convergence analysis of section 5 is applicable if the PDE coefficients are $O(n)$.

If the above conditions hold, the matrix is symmetrizable. Using the notation of the previous sections, let $\hat{S}$ be the symmetrized matrix, let $\beta_x = 1 + \gamma h - 2\gamma^2 h^2, \beta_y = 1 + \delta h - 2\delta^2 h^2,\ \beta_z = 1 + \mu h - 2\mu^2 h^2$, and let S^* be a modified version of $\hat{S}$, such that each occurrence of $c_{i+1} d_i,\ b_{j+1} e_j$, and $f_{k+1} g_k$ in $\hat{S}$ is replaced by the upper bounds, namely $\beta_x,\ \beta_y$, and β_z, respectively. Since $S^* \ge \hat{S}$, S^* is a symmetrized version of a matrix corresponding to the constant coefficient case, and by [9, Lem. 3.3], it is a diagonally dominant M-matrix. Using Theorem 5.1, the bounds on the convergence rate of the block Jacobi method are given in Table 6.1. As is evident, the bounds are tight even for small n. It should be noted, however, that as the PDE coefficients grow larger, the bounds are not expected to be as tight, as the inequalities in (6.4) become less effective.

Next, the spectral radii of the block Jacobi and block Gauss–Seidel iteration matrices and an approximation to the optimal SOR parameter have been computed for both the upwind scheme and the centered scheme. The relaxation parameter was computed according to the formula $\frac{2}{1+\sqrt{1-\rho_J^2}}$. It is only an approximation to the optimal SOR parameter since the matrix is only "close" to being consistently ordered [8]. The experiments were done on an $8 \times 8 \times 8$ grid (512 gridpoints). In Tables 6.2 and 6.3 the superscripts c and u stand for centered or upwind, respectively, R and U stand for reduced system and unreduced system, respectively, and the subscripts J and GS stand for Jacobi and Gauss–Seidel, respectively. We present the results for

TABLE 6.2
Spectral radius of the block Jacobi, block Gauss–Seidel, and approximate optimal relaxation parameter for the reduced system, using both upwind and centered differences and 1*D splitting.*

$p_1 = p_2 = p_3$	$\rho_J^u(R)$	$\rho_{GS}^u(R)$	$\tilde{\omega}^u(R)$	$\rho_J^c(R)$	$\rho_{GS}^c(R)$	$\tilde{\omega}^c(R)$
10	0.77	0.60	1.23	0.77	0.59	1.22
100	0.36	0.14	1.04	> 1	0.35	-

TABLE 6.3
Spectral radii of the block Jacobi, block Gauss–Seidel, and optimal relaxation parameter for the unreduced system, using both upwind and centered differences with 1*D splitting.*

$p_1 = p_2 = p_3$	$\rho_J^u(U)$	$\rho_{GS}^u(U)$	$\tilde{\omega}^u(U)$	$\rho_J^c(U)$	$\rho_{GS}^c(U)$	$\tilde{\omega}^c(U)$
10	0.90	0.81	1.39	0.91	0.82	1.40
100	0.66	0.44	1.14	> 1	> 1	-

two different cases, one of which has convection of moderate size, and the other with large convection, for which the upwind scheme is more effective than the centered scheme.

Note that for the unreduced system, in most cases the matrix satisfies all the conditions required for Young's SOR analysis [14]; thus, the spectral radius of the Gauss–Seidel matrix and the optimal relaxation parameter can be computed from the spectral radius of the block Jacobi matrix. By comparing Tables 6.2 and 6.3 it is evident that stationary solvers for the reduced system converge faster than for the unreduced system. In one case there is convergence for the reduced system and divergence for the unreduced system.

Moving to consider Krylov subspace solvers, in Table 6.4 we make a comparison between the performance of solvers for the two systems. The stopping criterion was relative residual smaller than 10^{-10}. The method that is used is nonpreconditioned Bi-CGSTAB. The table presents information on the complete process, namely, construction of the systems and the iterative solves. The increase in iteration counts as the grid is refined agrees with theory, at least if one assumes that for this well conditioned and mildly nonsymmetric system, the condition number is of magnitude $O(h^{-2})$ and the convergence rate is similar to that of the conjugate gradient method for symmetric positive definite systems. When one step of cyclic reduction is applied, the savings become more dramatic as the systems grow larger. An explanation for this is that the construction of the reduced system, which requires significantly more floating point operations compared with the construction of the unreduced system, becomes a less significant factor in the overall computation as the grid becomes finer. In general, since the iterative solve is the costly component of the computation, it is significant that the number of iterations until convergence of the unreduced solver is larger by a factor of approximately 2 compared with the reduced solver. Figure 6.1 illustrates the saving and the convergence behavior for this problem.

In Table 6.5 we provide some numerical evidence which suggests that the good performance of reduced solvers is due to effective preconditioning of the original matrix. In the table, estimates of the condition numbers for $p_1 = 500$, $p_2 = 200$, $p_3 = 100$ with upwind difference discretization are presented. The estimates were obtained using Matlab's command "condest." The factor of approximately 2 has been obtained for several additional cases that have been tested. More observations on the condition

TABLE 6.4
Comparison between the performance of the unreduced and the reduced solvers for increasing mesh size, using nonpreconditioned Bi-CGSTAB, for $p_1 = 50,\ p_2 = 20,\ p_3 = 10$.

n	n^3	Iterations		Mflops		Time (sec.)	
—	—	Unreduced	Reduced	Unreduced	Reduced	Unreduced	Reduced
64	262,144	153	79	3,556.8	1,956.1	958.1	502.7
80	512,000	191	90	8,637.4	4,346.7	2,325.9	1,125.2
96	884,736	224	113	17,497.2	9,376.7	4,689.8	2,454.9

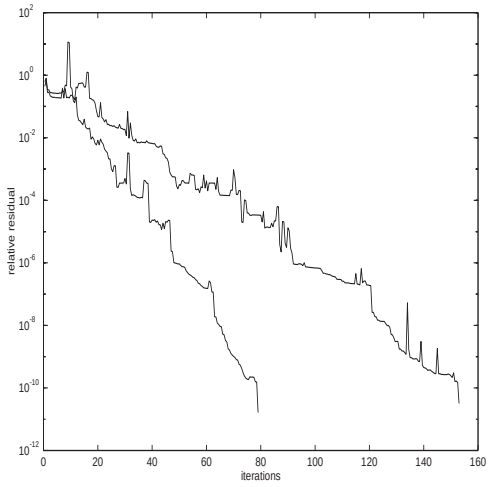


FIG. 6.1. *Relative residuals for nonpreconditioned Bi-CGSTAB applied to linear systems arising from discretization of the test problem, with* $p_1 = 50,\ p_2 = 20,\ p_3 = 10$, *on a* $64 \times 64 \times 64$ *grid. The residual associated with the reduced solver is the lower curve.*

number of the reduced matrix can be found in [7].

6.2. Test problem 2. Consider the nonseparable problem

$$-\Delta u + p_1 x\ \exp(x+y+z)u_x + p_2 y\ \exp(x+y+z)u_y + p_3\ z\ \exp(x+y+z)u_z = w(x,y,z)$$

on $\Omega = (0,1) \times (0,1) \times (0,1)$, with Dirichlet boundary conditions, where $w(x,y,z)$ is constructed so that the solution is (6.2). For this problem the convergence analysis of section 5 does not apply. Results are given in Table 6.6. The experiments were done for $n = 24$, so that the tensor-product grid has $13,824$ gridpoints. GMRES(5) [12] was used, preconditioned by ILU with drop tolerance of 10^{-3}. The stopping criterion was $||r_i||/||r_0|| < 10^{-7}$. In all cases that have been tested, setting up and solving the reduced system is faster compared with setting up and solving the unreduced system. It should be noted that the CPU times are affected by a long preconditioner setup time. When convection dominates, the centered scheme performs poorly. (In general, this scheme suffers numerical instability when the Reynolds numbers are large [10].) Additional numerical experiments indicate that both solvers have the property that for centered difference discretization the solver of mildly nonsymmetric systems converges faster than the solver of a close-to-symmetric system. This phenomenon was proved analytically for stationary methods for the constant coefficient case in [2] (2D) and in [9] (3D).

TABLE 6.5
Comparison between estimates of condition numbers of the unreduced matrix (denoted by U) vs. the reduced matrix (R), for $p_1 = 500,\ p_2 = 200,\ p_3 = 100$.

n	$\kappa_2(U)$	$\kappa_2(R)$
8	420.9	195.0
12	1,049.0	489.8
16	1,859.4	866.5
20	2,693.7	1,258.6
24	3,578.4	1,657.1

TABLE 6.6
Comparison of CPU times (seconds) for setting up and solving the unreduced system and the reduced system using ILU+GMRES. R and U stand for reduced and unreduced, respectively, and the subscripts c and u stand for centered and upwind, respectively.

$p_1 = p_2 = p_3$	U_c	U_u	R_c	R_u
10	58.0	66.2	39.9	47.5
50	91.2	43.5	48.7	32.7
100	148.2	37.3	62.1	23.8

6.3. Test problem 3. Consider the nonseparable problem

$$-0.1\Delta u + yzu_x + xzu_y + xyu_z = w,$$

with Neumann boundary conditions $u_z = 0$ on $z = 0$ and zero Dirichlet conditions for $x = 0$ and $y = 0$ on the unit cube. w was constructed so that the exact solution is $u(x, y, z) = \sin(\pi x)\sin(\pi y)\cos(\pi z)$. Here we compare the performance of a few Krylov subspace solvers; thus, the focus is on the actual iterative solve time, once the systems and the preconditioners were set up.

The results in Table 6.7 are for a $20 \times 20 \times 20$ grid. ILU(0) was used as a preconditioner. The stopping criterion was $||r_i||/||r_0|| < 10^{-7}$. In all cases the reduced solver converges faster than the unreduced solver. The factor of approximately 2 can be a good indication for the gain in performing one step of cyclic reduction in much finer grids. Bi-CGSTAB is slightly faster than CGS. These two schemes are faster than BiCG. The differences in performance between the solvers are qualitatively similar for the reduced and the unreduced systems.

TABLE 6.7
Iteration counts and solving times for three Krylov solvers.

Method	Iterations		Time (sec.)	
—-	Unreduced	Reduced	Unreduced	Reduced
BiCG	32	19	13.9	6.9
CGS	23	14	8.7	4.1
Bi-CGSTAB	19	11	8.2	3.8

7. Concluding remarks. A cyclically reduced operator for a 3D convection-diffusion equation with variable coefficients has been derived. Block orderings have been discussed, some solving techniques for the reduced system have been examined,

and numerical experiments illustrate the fact that the reduced system is easier to solve than the unreduced system.

The results presented in this work show that one step of cyclic reduction can be effectively used as a preconditioning technique for solving the convection-diffusion equation with variable coefficients. The questions of parallelism and applications are topics for further investigation.

Acknowledgments. I am very grateful to Jim Varah for carefully reading this manuscript and for his valuable advice. I would also like to thank Gene Golub, who has held helpful discussions with me and has pointed out some useful references. Many thanks to the two referees for their helpful comments and suggestions.

REFERENCES

[1] M. Benzi, D. B. Szyld, and A. van Duin, *Orderings for incomplete factorization preconditioning of nonsymmetric problems.* SIAM J. Sci. Comput., 20 (1999), pp. 1652–1670.

[2] H. C. Elman and G. H. Golub, *Iterative methods for cyclically reduced non-self-adjoint linear systems,* Math. Comp., 54 (1990), pp. 671–700.

[3] H. C. Elman and G. H. Golub, *Iterative methods for cyclically reduced non-self-adjoint linear systems* II, Math. Comp., 56 (1991), pp. 215–242.

[4] H. C. Elman and G. H. Golub, *Line iterative methods for cyclically reduced discrete convection-diffusion problems,* SIAM J. Sci. Comput., 13 (1992), pp. 339–363.

[5] G. H. Golub and C. F. Van Loan, *Matrix Computations,* 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[6] G. H. Golub and R. S. Tuminaro, *Cyclic Reduction/Multigrid,* Tech. report, Stanford University, Palo Alto, CA, 1993.

[7] C. Greif, *Analysis of Cyclic Reduction for the Numerical Solution of Three-Dimensional Convection-Diffusion Equations*, Ph.D. thesis, The University of British Columbia, Vancouver, BC, Canada, 1998.

[8] C. Greif and J. M. Varah, *Block stationary methods for nonsymmetric cyclically reduced systems arising from three-dimensional elliptic equations* SIAM J. Matrix Anal. Appl., 20 (1999), pp. 1038–1059.

[9] C. Greif and J. M. Varah, *Iterative solution of cyclically reduced systems arising from discretization of the three-dimensional convection diffusion equation,* SIAM J. Sci. Comput., 19 (1998), pp. 1918–1940.

[10] P. M. Gresho and R. L. Lee, *Don't suppress the wiggles—They're telling you something!,* Comput. & Fluids, 9 (1981), pp. 223–253.

[11] K. W. Morton, *Numerical Solution of Convection-Diffusion Problems*, Chapman and Hall, London, 1996.

[12] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, MA, 1996.

[13] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1962.

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