JOURNAL OF COMPUTATIONAL PHYSICS 98, 1–10 (1992)

Vertex-Centered and Cell-Centered Multigrid for Interface Problems

M. KHALIL AND P. WESSELING

Faculty of Technical Mathematics and Informatics, Delft University of Technology, P.O. Box 356, 2600 AJ Delft, The Netherlands

Received December 6, 1988; revised April 30, 1990

Cell-centered and vertex-centered multigrid methods for solving interface problems are studied. These methods differ in the location of the nodes in the grids and in the transfer operators. It is shown how by means of stencil notation a compact and precise description can be given of the transfer and coarse grid operators. A structured FORTRAN description of the fundamental multigrid algorithm with only one goto statement is presented. Numerical results of several test problems with strong discontinuities in equation coefficients are presented. Storage and work requirements are discussed. © 1992 Academic Press, Inc.

1. INTRODUCTION

Multigrid (MG) methods to solve large systems that arise from discretizing elliptic partial differential equations have two main components: coarse grid approximation and smoothing. In problems with interfaces (discontinuous diffusion coefficients), vertex-centered coarsening together with standard bilinear interpolation leads to inaccurate coarse grid approximation, resulting in deterioration of the rate of convergence. In [1, 2, 4–9], vertex-centered MG methods with matrix-dependent transfer operators are presented. Such methods have a good rate of convergence but need more storage and preparation time per iteration (calculation of transfer operators) than standard MG. A new method has been developed in [10, 12, 13], based on cell-centered coarsening and simple interpolatory transfer operators, for which the cost per iteration is the same as for standard MG. In this paper we give an evaluation of the performance of the two kinds of methods for several difficult test problems.

2. CELL-CENTERED AND VERTEX-CENTERED COARSENING

There are basically two ways to construct coarse grids in multigrid methods. Suppose that we have a two-dimensional domain subdivided in cells of equal size. Suppose that we have n_α cells, $\alpha = 1, 2$ in each direction. The unknowns may be located at vertices of the cells (cf. Fig. 1a), or at the centers of the cells (cf. Fig. 1b).

In the vertex-centered case the computational grid G is defined by

$$G = \{(x_1, x_2): x_\alpha = i_\alpha h_\alpha,\ i_\alpha = 0, 1, ..., n_\alpha,$$
$$h_\alpha = L_\alpha / n_\alpha,\ \alpha = 1, 2\}.$$

In the cell-centered case G is defined by

$$G = \{(x_1, x_2): x_\alpha = (i_\alpha - \tfrac{1}{2})\, h_\alpha,\ i_\alpha = 1, ..., n_\alpha,$$
$$h_\alpha = L_\alpha / n_\alpha,\ \alpha = 1, 2\}.$$

To each case there corresponds a coarsening which characterizes the corresponding multigrid method:

— Vertex-centered multigrid (MGVC) associated with vertex-centered coarsening.

— Cell-centered multigrid (MGCC) associated with cell-centered coarsening.

In this section we need to consider only two grids, called G and $\bar{G}$; $\bar{\cdot}$ indicates a coarse grid quantity. Let n_α be even.

2.1. Vertex-Centered Coarsening

The coarse grid $\bar{G}$ is selected as follows:

$$\bar{G} = \{x = (x_1, x_2): x_\alpha = i_\alpha \bar{h}_\alpha,\ i_\alpha = 0, 1, ..., \bar{n}_\alpha,$$
$$\bar{h}_\alpha = 2h_\alpha,\ \bar{n}_\alpha = n_\alpha / 2\}.$$

This means that $\bar{G}$ is obtained from G by deleting every other vertex in each direction (cf. Fig. 2a).

0021-9991/92 $3.00

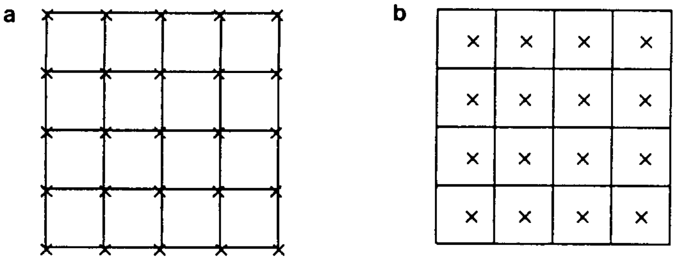

FIG. 1. a. Vertex-centered grid; b. cell-centered grid.

2.2. Cell-Centered Coarsening

The coarse grid $\bar{G}$ is selected as follows:

$$\bar{G} = \left\{ x = (x_1, x_2) : x_\alpha = \left(i_\alpha - \frac{1}{2} \right) \bar{h}_\alpha, i_\alpha = 1, ..., \bar{n}_\alpha, \right.$$
$$\left. \bar{n}_\alpha = \frac{n_\alpha}{2}, \bar{h}_\alpha = 2h_\alpha \right\}.$$

$\bar{G}$ is the set of coarse grid cell-centers where each coarse cell is the union of four fine cells (cf. Figs. 2b and 3).

3. EQUATION AND STENCIL NOTATION

3.1. Equation

We consider the following two-dimensional diffusion equation with discontinuous and/or anisotropic coefficients:

$$-\sum_{\alpha=1}^{2} \frac{\partial}{\partial x_\alpha} \left(D_\alpha \frac{\partial \phi}{\partial x_\alpha} \right) + \sigma\phi = f, \qquad x \in \Omega = \prod_{\alpha=1}^{2} (0, L_\alpha),$$
$$D_\alpha > 0, \qquad \sigma \geqslant 0, \quad (3.1a)$$

and boundary condition

$$b \frac{\partial \phi}{\partial n} + c\phi = g, \quad b \geqslant 0, \quad c \geqslant 0, \quad b + c \neq 0, \quad \text{on } \partial\Omega, \quad (3.1b)$$

where n is the outward normal to the boundary. D_α, f, g, σ, b, and c are given functions. D_α and σ may have strong discontinuities across internal interfaces. We use finite volume discretizations as described in [10, 11, 13]. The resulting system on the finest grid has a matrix A with five non-zero diagonals.

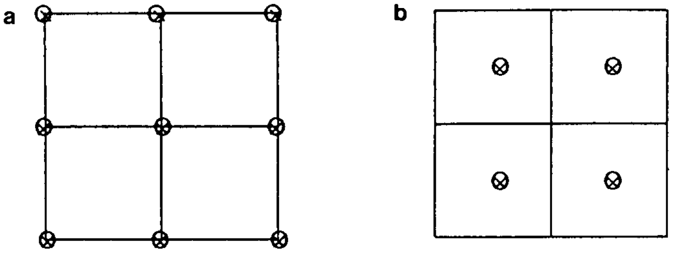

FIG. 2. a. Vertex-centered coarse grid; b. cell-centered coarse grid.

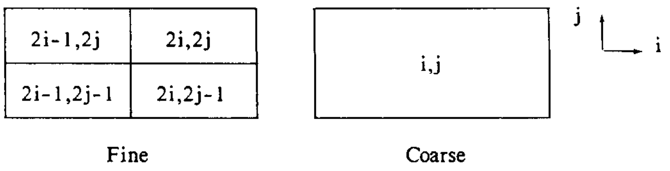

FIG. 3. Cell numbering in two dimensions.

3.2. Stencil Notation

We will make use of the following stencil notation. Let the grids be $G^1, G^2, ..., G^M$, with G^1 the finest grid and G^M the coarsest grid. Let the space of grid functions on G^k be denoted by $\Phi^k : G^k \to \mathbb{R}$. With $A^k : \Phi^k \to \Phi^k$, the stencil representation of $A^k\phi^k$ is defined by

$$(A^k\phi^k)_i = \sum_{j \in \mathbb{Z}^d} A^k(i, j)\, \phi^k_{i+j}, \qquad i \in G^k, \qquad (3.2)$$

where d is the dimension of G^k (here $d = 2$). The set of values

$$S_{A^k} = \{ j \in \mathbb{Z}^d : A^k(i, j) \neq 0, \text{ for some } i \in G^k \}, \qquad (3.3)$$

is called the structure of A^k. The stencil $[A^k]$ of A^k is defined in the usual way. For example, if A^k has the familiar five-point structure, we have

$$[A^k]_i = \begin{bmatrix} & A^k(i, e_2) & \\ A^k(i, -e_1) & A^k(i, 0) & A^k(i, e_1) \\ & A^k(i, -e_2) & \end{bmatrix},$$

where $e_1 = (1, 0)$ and $e_2 = (0, 1)$.

With $R^k : \Phi^{k-1} \to \Phi^k$ a restriction operator the stencil representation of $R^k\phi^{k-1}$ is defined as

$$(R^k\phi^{k-1})_i = \sum_{j \in \mathbb{Z}^d} R^k(i, j)\, \phi^{k-1}_{2i+j}. \qquad (3.4)$$

Define the inner product on Φ^k by

$$(\phi^k, \psi^k) = \sum_{i \in \mathbb{Z}^d} \phi^k_i \psi^k_i. \qquad (3.5)$$

If $i \notin G^k$ the corresponding grid function values are defined to be zero. With this inner product, one finds, for the adjoint $R^{k,*} : \Phi^k \to \Phi^{k-1}$:

$$(R^{k,*}\phi^k)_i = \sum_{j \in \mathbb{Z}^d} R^k(j, i - 2j)\, \phi^k_j. \qquad (3.6)$$

From the preceding equation follows the stencil notation for $P^k\phi^{k+1}$ for prolongation operators $P^k: \Phi^{k+1} \to \Phi^k$:

$$(P^k\phi^{k+1})_i = \sum_{j\in\mathbb{Z}^d} P^{k,*}(j, i-2j)\,\phi_j^{k+1}. \tag{3.7}$$

4. MULTIGRID ALGORITHM

Let the discrete fine grid problem to be solved be denoted by

$$A\phi = f. \tag{4.1}$$

We present a simple well-structured FORTRAN description, using only one **goto**, of the fundamental MG algorithm (correction variant, V-, F-, or W-cycle), see [12].

```
     Subroutine MG($\phi$, $f$)
     $k=1$
     ic(1) = maxit
     if ($V$-cycle) then $\gamma=1$ else $\gamma=2$
10   if (ic(1).gt.0) then
       if (ic($k$).eq.0.or.$k$.eq.$M$) then
         if ($k$.eq.$M$) then
           call $S(\phi^M, f^M)$
           ic($M$) = ic($M$) $-$ 1
           if ($F$-cycle) $\gamma=1$
         endif
         $k=k-1$
         $\phi^k=\phi^k+P^k\phi^{k+1}$
         call $S(\phi^k, f^k)$
         ic($k$) = ic($k$) $-$ 1
     else
         call $S(\phi^k, f^k)$
         $f^{k+1}=R^{k+1}(f^k-A^k\phi^k)$
         $k=k+1$
         $\phi^k=0$
         ic($k$) = $\gamma$
     endif
     if ($k$.eq.1.and.$F$-cycle) $\gamma=2$
     goto 10
   endif
end.
```

$k=1$	:finest grid
maxit	:number of MG iterations desired
ic(k)	: counter of MG cycles on grid k
$k=M$	:coarsest grid
S	:smoothing subroutine
f^k	:right-hand side on grid k
ϕ^k	:correction ($k \geqslant 2$) or solution ($k=1$).

5. TRANSFER OPERATORS

5.1. Transfer operators for MGVC

In vertex-centered multigrid for interface problems, matrix-dependent transfer operators have to be used. Several definitions of such operators have been given [1, 2, 4–9]. Here we restrict ourselves to the "collecting" matrix dependent operators proposed in [1, 2]. They can be conveniently described by the stencil notation introduced in Section 3. $P\phi$ is represented as (cf. (3.7))

$$(P\bar{\phi})_i = \sum_{j\in\mathbb{Z}^d} P^*(j, i-2j)\,\bar{\phi}_j, \qquad i\in G, \tag{5.1}$$

where we drop the superscript k for convenience and indicate coarse grid quantities by an overbar. P is defined by the stencil $[P^*]_i$.

Let $e_1=(1,0)$, $e_2=(0,1)$, and $e_3=(1,1)\;(=e_1+e_2)$. The vertex $i\in\bar{G}$ coincides with the vertex $2i\in G$ (Fig. 4). We define

$$(P\bar{\phi})_{2i}=\bar{\phi}_i$$

and

$$P^*(j, i-2j)=0 \qquad \text{for} \quad |i_\alpha - 2j_\alpha|>1, \qquad \alpha=1,2. \tag{5.2}$$

In other words, we have interpolation between nearest neighbours only. Using (5.1) we see immediately that

$$P^*(i,0)=1. \tag{5.3}$$

For the grid points $2i+e_\alpha$, $\alpha=1,2$, P is defined as follows: Let $[A]_i$ be given by

$$[A]_i=\begin{bmatrix} A(i,\,-e_1+e_2) & A(i,e_2) & A(i,e_3) \\ A(i,\,-e_1) & A(i,0) & A(i,e_1) \\ A(i,\,-e_3) & A(i,\,-e_2) & A(i,e_1-e_2) \end{bmatrix}. \tag{5.4}$$

Let C be a matrix which is A or a slight modification of A

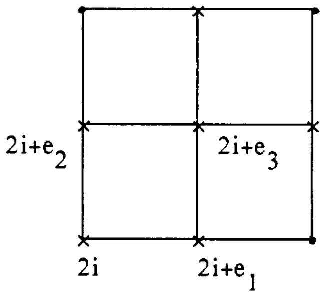

FIG. 4. Indices of grid points.

as defined below. By summing the columns of $[C]_i$, we obtain the operator H defined by

$$H(i, (\alpha, 0)) = \sum_{\beta=-1}^{1} C(i, (\alpha, \beta)), \qquad \alpha = -1, 0, 1. \tag{5.5}$$

By summing the rows of $[C]_i$ we obtain the operator V defined by

$$V(i, (0, \beta)) = \sum_{\alpha=-1}^{1} C(i, (\alpha, \beta)), \qquad \beta = -1, 0, 1. \tag{5.6}$$

For grid point $k = 2i + e_1$, we define $P\phi$ by

$$(H(P\bar{\phi}))_k = 0. \tag{5.7}$$

Using (5.7), (5.1), and (5.2) we have, after some manipulation (cf. [10]),

$$\begin{aligned} &P^*(l, k-2l) \\ &\quad = -\sum_{\alpha \neq 0} H(k, (\alpha, 0))\, P^*(l, k + (\alpha, 0) - 2l)/H(k, 0), \\ &\qquad k = 2i + e_1, \qquad l = i \qquad \text{or} \qquad i + e_1. \end{aligned} \tag{5.8}$$

We obtain

$$\begin{aligned} P^*(i, e_1) &= -\frac{H(2i+e_1, -e_1)}{H(2i+e_1, 0)}, \\ P^*(i+e_1, -e_1) &= -\frac{H(2i+e_1, e_1)}{H(2i+e_1, 0)}. \end{aligned} \tag{5.9}$$

We deduce

$$P^*(i, -e_1) = -\frac{H(2i-e_1, e_1)}{H(2i-e_1, 0)}. \tag{5.10}$$

Using the same method, we deduce for the grid point $2i + e_2$:

$$\begin{aligned} P^*(i, e_2) &= -\frac{V(2i+e_2, -e_2)}{V(2i+e_2, 0)}, \\ P^*(i, -e_2) &= -\frac{V(2i-e_2, e_2)}{V(2i-e_2, 0)}. \end{aligned} \tag{5.11}$$

For the points $k = 2i + e_3$, we define $P\phi$ such that

$$(C(P\bar{\phi}))_k = 0. \tag{5.12}$$

This gives

$$P^*(l, k-2l) = -\sum_{j \neq 0} C(k, j) P^*(l, k+j-2l)/C(k, 0). \tag{5.13}$$

For $k = 2i + e_3$, l can take the values i, $i + e_1$, $i + e_2$, and $i + e_3$ only. By substitution of these values of l in (5.13) we obtain the expressions for the remaining elements $P^*(i, e)$ with $e = (\alpha_1, \alpha_2)$ and $\|e\| = |\alpha_1| + |\alpha_2| = 2$:

$$P^*(i, e) = -\frac{\sum_{j:\, \|j+e\| \leqslant 1} C(2i+e, j)\, P^*(i, j+e)}{C(2i+e, 0)}. \tag{5.14}$$

The elements of P^* occurring in the right-hand side are defined by (5.2), (5.3), (5.9), (5.10), and (5.11). The restriction operator will be defined by

$$R = P^*. \tag{5.15}$$

We will choose C in two ways:

$$(1) \qquad C = A \tag{5.16}$$

as proposed in [1, 2, 4, 7–9]. The transfer operators following from this choice will be called type A.

$$(2) \qquad C(i, j) = A(i, j) \quad \text{for} \quad j \neq 0 \tag{5.17a}$$

and (cf. [5])

$$C(i, 0) = \begin{cases} -\sum_{j \neq 0} A(i, j), & \text{if } \left|\sum_j A(i, j)/\sum_{j \neq 0} A(i, j)\right| < 10^{-p} \\ A(i, 0), & \text{otherwise,} \end{cases} \tag{5.17b}$$

where p is an integer to be chosen. The transfer operators related to this choice of C will be called type B.

5.2. Transfer Operators for MGCC

5.2.1. *Restriction*

In the interior of the domain the restriction operator R is the (scaled) adjoint of linear interpolation in triangles. Interpolation takes place in triangles such as HKF and HEF (Fig. 5). The resulting stencil is

$$[R]_i = \frac{1}{16}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 0 & 2 & 3 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \tag{5.18}$$

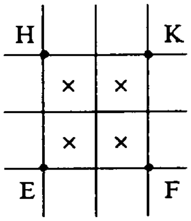

FIG. 5. Interpolation for adjoint of restriction; × : center of fine cells; •: center of coarse cells.

At boundaries this stencil has to be modified because some elements of this stencil refer to points outside the grid. For pure Neumann boundary conditions, when the solution is determined up to a constant, nice properties to have are (cf. [6]):

$$R^*\bar{e}=e, \qquad Re=\bar{e} \tag{5.19}$$

where e and $\bar{e}$ are grid functions which equal one everywhere on the fine and coarse grids, respectively. The resulting restriction operator will be used regardless of the type of boundary condition given.

It is found that a suitable boundary modification of the stencil of R satisfying (5.19) is obtained by the following rule: elements to be deleted because they refer to function values outside the domain are added to the nearest neighbour in the "north-west/south-east direction." This leads to the following stencils at the west boundary, north-west corner, and south-west corner, respectively:

$$\begin{aligned}
[R]_i &= \frac{1}{16}\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 4 & 2 & 0 \\ 0 & 3 & 3 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \\
[R]_i &= \frac{1}{16}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 4 & 3 & 0 \\ 0 & 3 & 3 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \\
[R]_i &= \frac{1}{16}\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 4 & 2 & 0 \\ 0 & 4 & 4 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.
\end{aligned} \tag{5.20}$$

The situation at other boundaries and corners follows from these examples. In general, R has the following stencil:

$$[R]_i = \begin{bmatrix} w_{2,i} & w_{3,i} & 0 & 0 \\ w_{1,i} & 4-w_{2,i} & 4-w_{3,i}-e_{3,i} & 0 \\ 0 & 4-w_{1,i}-e_{1,i} & 4-e_{2,i} & e_{3,i} \\ 0 & 0 & e_{1,i} & e_{2,i} \end{bmatrix}. \tag{5.21}$$

The values of e_α and w_α, $\alpha=1,2,3,4$, depend on the position of point i relative to the boundaries, in the way just discussed.

In multigrid methods P and R must satisfy the following requirement. Let m_P-1, m_R-1 be the maximum degree of polynomials that are interpolated exactly by sP or tR^* respectively for some real value of s or t. Then we must have [3, 6]

$$m_P+m_R>2m, \tag{5.22}$$

with $2m$ the order of the partial differential equation to be solved. The above restriction has $m_R=2$. It is therefore sufficient here to take an interpolation P with $m_P=1$. We define P by

$$[P^*]_i = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}. \tag{5.23}$$

Hence

$$(P\bar{\phi})_{2i+j}=\bar{\phi}_i \qquad \text{for} \quad i\in G, \qquad j\in S_{P^*}=\{-1,0\}^2.$$

When R is replaced by $1/4P^*$ and P by $4R^*$ the behaviour of the method remains about the same.

6. COARSE GRID OPERATORS

In this section we need to consider only two grids. Coarse grid quantities are denoted by an overbar. The coarse grid operator $\bar{A}$ is defined by

$$\bar{A}=RAP. \tag{6.1}$$

In the following we give a method for the computation of $\bar{A}$ in the case of MGVC and MGCC. Using (3.2) and (3.7) we obtain

$$\begin{aligned}
(AP\bar{\phi})_i &= \sum_k A(i,k)(P\bar{\phi})_{i+k} \\
&= \sum_k A(i,k)\sum_j P^*(j,i+k-2j)\,\bar{\phi}_j
\end{aligned} \tag{6.2}$$

and using (3.4) we derive, after some manipulations [10],

$$\begin{aligned}
\bar{A}(i,j) = \sum_m \sum_k & R(i,m)\,A(2i+m,k) \\
& \times P^*(i+j,m+k-2j).
\end{aligned} \tag{6.3}$$

In order to determine the storage required, we have to compute the set

$$S_{\bar{A}}=\{j\in \mathbb{Z}^d\colon \bar{A}(i,j)\neq 0 \text{ for at least one } i\in \bar{G}\}. \tag{6.4}$$

This is done by the following algorithm:

```
ALGORITHM structure.
$S_{\bar{A}}=\varnothing$
for $m\in S_R$ do
    for $k\in S_A$ do
        for $p\in S_{P^*}$ do
            begin $j=(m+k-p)/2$
                if $j\in \mathbb{Z}^d$ then $S_{\bar{A}}=S_{\bar{A}}\cup\{j\}$
            end.
```

Given $S_{\bar{A}}$, the following algorithm computes $\bar{A}$:

```
ALGORITHM RAP.
$\bar{A}(i, j) = 0$
for $j \in S_{\bar{A}}$ do
    for $m \in S_R$ do
        for $k \in S_A$ while $(m + k - 2j \in S_{P^*})$ do
            for $i \in \bar{G}$ while $(2i + m \in G$ and $i + j \in \bar{G})$ do
                $\bar{A}(i, j) = \bar{A}(i, j) + R(i, m)\, A(2i + m, k)$
                $P^*(i + j, m + k - 2j)$.
```

The inner loop does not allow vectorization because of the **while** clause. We therefore replace the last two lines by

$$G_1 = \{i \in \bar{G} : 2i + m \in G\} \tag{6.5}$$

$$G_2 = \{i \in \bar{G} : i + j \in \bar{G}\} \tag{6.6}$$

$$G_3 = G_1 \cap G_2 \tag{6.7}$$

```
for $i \in G_3$ do
    $\bar{A}(i, j) = \bar{A}(i, j) + R(i, m)\, A(2i + m, k)$
    $P^*(i + j, m + k - 2j)$.
```

The inner loop vectorizes along grid lines.

This algorithm is completely general. It covers cell-centered and vertex-centered multigrid and matrix-dependent P and R. Notice that in cell-centered multigrid, the algorithms can be simplified because $P^*(i, j) = 1, j \in S_{P^*}$.

To illustrate the computation of G_3 we give an example in two dimensions. Let $\bar{G}$ and G given by

$$\bar{G} = \{i = (i_1, i_2) : 0 \leqslant i_1 \leqslant nxc, 0 \leqslant i_2 \leqslant nyc\}$$

$$G = \{i = (i_1, i_2) : 0 \leqslant i_1 \leqslant 2nxc, 0 \leqslant i_2 \leqslant 2nyc\},$$

where $i \in G_3$ is equivalent to

$$m = (m_1, m_2) \in S_R, \qquad j = (j_1, j_2) \in S_{\bar{A}}$$

$$\max\left(-j_1, -\frac{m_1}{2}, 0\right) \leqslant i_1 \leqslant \min\left(nxc - \frac{m_1}{2}, nxc - j_1, nxc\right)$$

$$\max\left(-j_2, -\frac{m_2}{2}, 0\right) \leqslant i_2 \leqslant \min\left(nyc - \frac{m_2}{2}, nyc - j_2, nyc\right).$$

In the cell-centered case it is simple to give $\bar{A}$ explicitly. Let A be given by

$$[A]_{ij} = \begin{bmatrix} \eta_{ij} & \zeta_{ij} & 0 \\ \gamma_{ij} & \delta_{ij} & \varepsilon_{ij} \\ 0 & \alpha_{ij} & \beta_{ij} \end{bmatrix}, \tag{6.8}$$

where (temporarily) $i, j \in \mathbb{Z}$. It is found that if R is given by (5.21) and P by (5.23), then $\bar{A} = RAP$ is given by

$$[\bar{A}]_{ij} = \begin{bmatrix} \bar{\eta}_{ij} & \bar{\zeta}_{ij} & 0 \\ \bar{\gamma}_{ij} & \bar{\delta}_{ij} & \bar{\varepsilon}_{ij} \\ 0 & \bar{\alpha}_{ij} & \bar{\beta}_{ij} \end{bmatrix}, \tag{6.9}$$

where

$$\begin{aligned} \bar{\alpha}_{ij} = {} & (e_1(\alpha + \gamma + \delta))_{2i, 2j-2} + (e_2 \gamma)_{2i+1, 2j-2} \\ & + ((4 - e_1 - w_1)(\alpha + \beta))_{2i-1, 2j-1} + ((4 - e_2)\, \alpha)_{2i, 2j-1} \end{aligned} \tag{6.10.1}$$

$$\begin{aligned} \bar{\beta}_{ij} = {} & (e_1(\beta + \varepsilon))_{2i, 2j-2} + (e_2(\alpha + \beta + \delta + \varepsilon))_{2i+1, 2j-2} \\ & + ((4 - e_2)\, \beta)_{2i, 2j-1} + (e_3(\alpha + \beta))_{2i+1, 2j-1} \end{aligned} \tag{6.10.2}$$

$$\begin{aligned} \bar{\gamma}_{ij} = {} & ((4 - e_1 - w_1)(\gamma + \eta))_{2i-1, 2j-1} + (w_1(\alpha + \gamma + \delta))_{2i-2, 2j} \\ & + ((4 - w_2)\, \gamma)_{2i-1, 2j} + (w_2 \alpha)_{2i-2, 2j+1} \end{aligned} \tag{6.10.3}$$

$$\begin{aligned} \bar{\delta}_{ij} = {} & (e_1(\eta + \zeta))_{2i, 2j-2} + (e_2 \eta)_{2i+1, 2j-2} \\ & + ((4 - e_1 - w_1)(\delta + \varepsilon + \zeta))_{2i-1, 2j-1} \\ & + ((4 - e_2)(\gamma + \delta + \eta + \zeta))_{2i, 2j-1} + (e_3(\gamma + \eta))_{2i+1, 2j-1} \\ & + (w_1(\beta + \varepsilon))_{2i-2, 2j} + ((4 - w_2)(\alpha + \beta + \delta + \varepsilon))_{2i-1, 2j} \\ & + ((4 - e_3 - w_3)(\alpha + \gamma + \delta))_{2i, 2j} + (w_2 \beta)_{2i-2, 2j+1} \\ & + (w_3(\alpha + \beta))_{2i-1, 2j+1} \end{aligned} \tag{6.10.4}$$

$$\begin{aligned} \bar{\varepsilon}_{ij} = {} & (e_2 \zeta)_{2i+1, 2j-2} + ((4 - e_2)\, \varepsilon)_{2i, 2j-1} \\ & + (e_3(\delta + \varepsilon + \zeta))_{2i+1, 2j-1} + ((4 - e_3 - w_3)(\beta + \varepsilon))_{2i, 2j} \end{aligned} \tag{6.10.5}$$

$$\begin{aligned} \bar{\eta}_{ij} = {} & (w_1(\eta + \zeta))_{2i-2, 2j} + ((4 - w_2)\, \eta)_{2i-1, 2j} \\ & + (w_2(\gamma + \delta + \eta + \zeta))_{2i-2, 2j+1} + (w_3(\gamma + \eta))_{2i-1, 2j+1} \end{aligned} \tag{6.10.6}$$

$$\begin{aligned} \bar{\zeta}_{ij} = {} & ((4 - w_2)\, \zeta)_{2i-1, 2j} + ((4 - e_3 - w_3)(\eta + \zeta))_{2i, 2j} \\ & + (w_2 \varepsilon)_{2i-2, 2j+1} + (w_3(\delta + \varepsilon + \zeta))_{2i-1, 2j+1}. \end{aligned} \tag{6.10.7}$$

As $R \neq P^*$, symmetry is not conserved in general. Let us consider the special case,

$$[A]_{ij} = \begin{bmatrix} 0 & \zeta_{ij} & 0 \\ \gamma_{ij} & \delta_{ij} & \varepsilon_{ij} \\ 0 & \alpha_{ij} & 0 \end{bmatrix} \tag{6.11}$$

with $\alpha_{ij} = \zeta_{i, j-1}, \gamma_{ij} = \varepsilon_{i-1, j}$ (symmetry) and $(\alpha + \gamma + \delta + \varepsilon + \zeta)_{ij} = 0$ in the interior cells. Then we have

$$[\bar{A}]_{ij} = \begin{bmatrix} 0 & \bar{\zeta}_{ij} & 0 \\ \bar{\gamma}_{ij} & \bar{\delta}_{ij} & \bar{\varepsilon}_{ij} \\ 0 & \bar{\alpha}_{ij} & 0 \end{bmatrix} \tag{6.12}$$

with

$$\bar{\alpha}_{ij} = (\alpha_{2i-1,2j-1} + \alpha_{2i,2j-1})/8 \qquad (6.13.1)$$

$$\bar{\gamma}_{ij} = (\gamma_{2i-1,2j-1} + \gamma_{2i-1,2j})/8 \qquad (6.13.2)$$

$$\bar{\varepsilon}_{ij} = (\varepsilon_{2i,2j} + \varepsilon_{2i,2j-1})/8 \qquad (6.13.3)$$

$$\bar{\zeta}_{ij} = (\zeta_{2i-1,2j} + \zeta_{2i,2j})/8. \qquad (6.13.4)$$

At boundaries and corners, $\bar{\alpha}_{ij}$, $\bar{\gamma}_{ij}$, $\bar{\varepsilon}_{ij}$, and $\bar{\zeta}_{ij}$ are obtained in the same way, by substitution of the corresponding values of e_α and w_α in (6.10.1), (6.10.3), (6.10.5), and (6.10.7), respectively.

It is found that if A is given by (6.11) and if

$$(\alpha + \gamma + \delta + \varepsilon + \zeta)_{ij} = s \qquad (6.14)$$

with s constant in all (interior, boundary) cells, then $\bar{A}$ is symmetric and $(\bar{\alpha} + \bar{\gamma} + \bar{\delta} + \bar{\varepsilon} + \bar{\zeta})_{ij} = s$.

In order to make $\bar{A}$ as sparse as A, and in order to have $\bar{A}$ symmetric even when (6.14) is not satisfied, the coarse grid matrix definition is modified as follows. We split A as

$$A = B + S, \qquad (6.15)$$

where C is defined by

$$\begin{aligned} B(i, j) &= A(i, j) \quad \text{for} \quad j \neq 0 \\ B(i, 0) &= -\sum_{j \neq 0} A(i, j). \end{aligned} \qquad (6.16)$$

The coarse grid matrix is defined by

$$\bar{A} = RBP + \bar{S} \qquad (6.17)$$

with

$$\begin{aligned} \bar{S}(i, 0) = \tfrac{1}{4}\{&S(2i, 0) + S(2i - e_1, 0) \\ &+ S(2i - e_2, 0) + S(2i - e_3, 0)\}, \end{aligned} \qquad (6.18)$$

where e_1, e_2, and e_3 are defined as in Section 5.1.

7. NUMERICAL EXPERIMENTS

In this section we use the vertex-centered multigrid method with matrix-dependent transfer operators and the cell-centered multigrid method to solve three special cases of (3.1). In all tests W-cycles are used. For test problem 1 the V-cycle is also used. We use one presmoothing and one postsmoothing by the symmetric (forward–backward) point Gauss–Seidel method, or the alternating zebra method (line-Gauss–Seidel, ordering the lines in a zebra pattern, using lines in both directions) when anisotropy is involved. The coarsest grid consists of 1×1 or 3×3 cells in the cell-centered case and 2×2 or 3×3 vertices in the vertex-centered case. The solution method on the coarsest grid is a direct method using QR decomposition. When on the coarsest grid the system is singular, we take the solution $\phi = 0$ on a 1×1 grid; on other grids, we replace one equation i of the system by $\phi_i = 0$. The results of the tests are collected in the tables given later. The quantities listed in the tables are the reduction factors κ, defined by

$$\kappa = \frac{\|r^{(\nu)}\|_2}{\|r^{(\nu-1)}\|_2}, \qquad \nu \geq 1, \qquad (7.1)$$

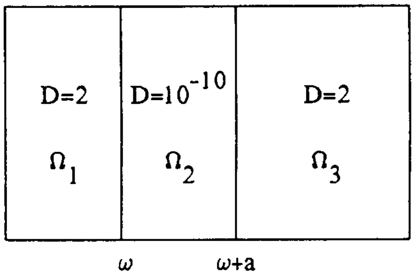

FIG. 6. Subdivision of Ω and diffusion coefficients for problem 1.

TABLE Ia

Results for Test Problem 1,
Cell-Centered Multigrid

			MGCC, W-cycle			MGCC, V-cycle		
h	ω	a	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/8	0.6250	1/8	5	0.075	0.061	6	0.128	0.079
1/12	0.5833	1/6	5	0.071	0.055	5	0.082	0.061
1/16	0.5625	1/8	6	0.135	0.081	9	0.320	0.204
1/24	0.5417	1/8	5	0.70	0.048	6	0.148	0.092
1/32	0.5313	1/8	5	0.065	0.046	9	0.338	0.196
1/48	0.5208	1/8	5	0.064	0.045	12	0.477	0.306
1/12	0.5833	h	6	0.094	0.072	6	0.142	0.095
1/16	0.5625	h	7	0.182	0.109	10	0.364	0.243
1/24	0.5417	h	9	0.322	0.196	15	0.535	0.394
1/32	0.5313	h	11	0.432	0.276	20	0.638	0.497
1/48	0.5208	h	16	0.579	0.410	29	0.750	0.620

TABLE Ib

Results for Test Problem 1,
Vertex-Centered Multigrid

			MGVC, W-cycle			MGVC, V-cycle		
h	ω	a	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/10	0.6000	1/10	6	0.099	0.091	6	0.099	0.091
1/14	0.5714	1/7	6	0.086	0.076	6	0.114	0.092
1/18	0.5556	1/9	6	0.091	0.086	6	0.108	0.097
1/26	0.5385	3/26	6	0.104	0.094	7	0.148	0.123
1/34	0.5294	2/17	6	0.098	0.089	7	0.172	0.121
1/50	0.5200	3/25	6	0.099	0.090	7	0.171	0.123

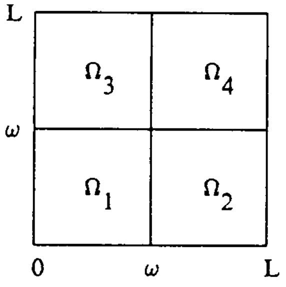

FIG. 7. Subdivision of Ω.

where $r^{(v)}$ is the residual at the iteration v ($r^{(v)} = f - A\phi^{(v)}$) and the average reduction factor $\bar{\kappa}$ is defined by

$$\bar{\kappa} = \left\{ \frac{\|r^{(v)}\|_2}{\|r^{(0)}\|_2} \right\}^{1/v}. \tag{7.2}$$

Of course, κ and $\bar{\kappa}$ depend not only on the multigrid method used, but also on v, f, and $\phi^{(0)}$. However, it was found that in many cases κ tends rapidly to the spectral radius. We take v such that $\bar{\kappa}^v < 10^{-6}$. In cases where the methods converge well, the value of κ thus obtained is not far from the spectral radius.

Two positions of the interfaces will be considered:

— Case 1. Interfaces are midway between two mesh lines;

— Case 2. Interfaces are mesh lines.

Note that in Case 1 (resp. Case 2) diffusivities across interfaces are the harmonic (resp. arithmetic) average of the diffusion coefficients in the adjacent cells in the vertex-centered case. In the cell-centered case, the arithmetic average is used in Case 1 and the harmonic average in Case 2.

Test Problem 1 [11, 13]. The domain is the unit square discretized into cells of size h, $h = 1/n$. The diffusion coefficients D_α in (3.1a) are given in the subdomains (Fig. 6)

$$\Omega_1 = \{(x_1, x_2): 0 \leqslant x_1 < \omega, 0 \leqslant x_2 \leqslant 1\},$$
$$\Omega_2 = \{(x_1, x_2): \omega \leqslant x_1 < \omega + a, 0 \leqslant x_2 \leqslant 1\},$$
$$\Omega_3 = \{(x_1, x_2): \omega + a \leqslant x_1 \leqslant 1, 0 \leqslant x_2 \leqslant 1\};$$

$x_1 = \omega$ and $x_1 = \omega + a$ are interfaces (lines of discontinuities) with a the width of the band Ω_2. D_α is defined by

$$D_\alpha = D = \begin{cases} 2 & \text{for } (x_1, x_2) \in \Omega_\beta, \quad \beta = 1 \text{ and } 3, \\ 10^{-10} & \text{for } (x_1, x_2) \in \Omega_2, \quad \alpha = 1, 2. \end{cases}$$

TABLE II

D and f for Test Problem 2

	Ω_1	Ω_2	Ω_3	Ω_4
D	1	10^3	10^3	1
f	0	1	1	0

TABLE III

Results for Problem 2, Case 1

		MGVC			MGCC		
L	ω	v	κ	$\bar{\kappa}$	v	κ	$\bar{\kappa}$
8	5.5	7	0.113	0.114	13	0.337	0.340
12	7.5	7	0.111	0.112	12	0.304	0.314
16	9.5	7	0.106	0.107	14	0.343	0.360
24	13.5	7	0.104	0.104	14	0.330	0.356
32	17.5	7	0.103	0.103	15	0.343	0.374
64	33.5	7	0.103	0.103	15	0.340	0.386

The right-hand side is $f(x_1, x_2) = x_1 x_2$, and $\sigma = 0$. We take a Dirichlet boundary condition ($b = 0$, $c = 1$ in (3.1b)) with $g(x_1, x_2) = x_1^2 + x_2^2$. The results of the computations in Case 2 are given in Tables I. Results for Case 1 are nearly the same as for Case 2 and are not reported here.

The values of h in the vertex-centered case are slightly different from those in the cell-centered case because in the vertex-centered case the Dirichlet boundaries have been eliminated, resulting in a number of meshes in each direction which is a small integer times a power of 2, plus 2.

The results are found to depend very little on the value of ω. MGVC with the W-cycle needs about the same number of iterations as MGCC with the W-cycle, for $a \simeq \frac{1}{8}$. For $a = h$ the rate of convergence of MGCC deteriorates with $1/h$. This is thought to be due to the fact that, as suggested by A. Brandt (private communication), according to (6.13) the isolation between the regions separated by the vertical strip may disappear after two coarsenings, depending on the value of ω. Nevertheless, convergence of MGCC is still rapid; see [11, 13]. In a physical application a is expected

TABLE IV

Results for Problem 2, Case 2

		MGVC			MGCC		
L	ω	v	κ	$\bar{\kappa}$	v	κ	$\bar{\kappa}$
8	5	8	0.121	0.139	9	0.182	0.196
12	7	8	0.121	0.146	9	0.197	0.212
16	9	8	0.123	0.150	9	0.196	0.213
24	13	8	0.124	0.157	10	0.207	0.226
32	17	8	0.123	0.161	10	0.210	0.233
64	33	8	0.121	0.169	11	0.228	0.256

TABLE V

Diffusion Coefficients for Problem 3

	Ω_1	Ω_2	Ω_3	Ω_4
D_1	1	10^{-2}	1	10^{-2}
D_2	10^2	10^2	10^{-2}	10^{-2}

not to depend on h, of course. The rate of convergence of MGVC is found to be insensitive to the value of a (results not shown here); the subregions remain isolated on coarse grids.

With the V-cycle, MGVC needs hardly more iterations than with the W-cycle, but the rate of convergence of MGCC becomes h-dependent. This is thought to be due to the fact that the cell-centered transfer operators satisfy $m_P + m_R = 3$ (cf. (5.22)), whereas in the vertex-centered case $m_P + m_R = 4$. Hence, the coarse grid operators are less accurate in the former case. Therefore we use only the W-cycle from now on.

For test problems 2 and 3 we define the following subdivision of the domain:

Let $\Omega = [0, L] \times [0, L]$, L given. Let Ω be subdivided in four subdomains Ω_i, $i = 1, 2, 3, 4$ (Fig. 7). Here ω designates the location of the interfaces in the x_1 or x_2 direction.

Test Problem 2 [4]. For this problem D_1 and D_2 in (3.1a) are taken equal to D; f and D are discontinuous across the internal boundaries of Ω_i, $i = 1, 2, 3, 4$, and are defined in Table II. Furthermore, $\sigma = 1/3D$. The boundary conditions are defined as

$$\frac{\partial \phi}{\partial n} = 0 \qquad \text{for} \quad x_1 = 0 \qquad \text{or} \qquad x_2 = 0$$

$$\frac{\partial \phi}{\partial n} + \frac{1}{2D}\phi = 0 \qquad \text{for} \quad x_1 = L \qquad \text{or} \qquad x_2 = L.$$

Table III gives results for Problem 2 in Case 1 for MGVC and MGCC for various values of L, taking $h = 1$. The smoother used is the symmetric point Gauss–Seidel method. We see that MGVC is two times faster in number of iterations than MGCC.

TABLE VI

Results for Problem 3a, Case 1

		MGVC			MGCC		
h	ω	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/8	0.6875	4	0.011	0.015	6	0.098	0.097
1/12	0.6250	4	0.010	0.011	6	0.111	0.073
1/16	0.5938	4	0.023	0.018	6	0.112	0.093
1/24	0.5625	4	0.029	0.020	7	0.136	0.117
1/32	0.5469	4	0.033	0.021	7	0.147	0.118
1/64	0.5234	4	0.044	0.028	7	0.145	0.115

TABLE VII

Results for Problem 3a, Case 2

		MGVC			MGCC		
h	ω	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/8	0.6250	4	0.011	0.016	5	0.050	0.042
1/12	0.5833	4	0.014	0.013	6	0.041	0.071
1/16	0.5625	4	0.025	0.023	6	0.142	0.095
1/24	0.5417	4	0.029	0.024	7	0.123	0.112
1/32	0.5313	4	0.033	0.025	7	0.137	0.111
1/64	0.5156	4	0.046	0.037	6	0.154	0.100

Table IV gives results for Problem 2 in Case 2. In this case MGCC is slightly slower than MGVC. For both methods, the reduction factors are independent of h.

Test Problem 3 [5]. This is a two-dimensional version of the three-dimensional problem (4.4) reported in [5]. For this problem $L = 1$ and D_i, $i = 1, 2$, are given in Table V. Problem 3a will refer to the case where unit sources of opposite sign are present at the "south-west" and "north-east" corner cells, and $\sigma = 0$. Problem 3b will refer to the case where a unit source is present at the "south-west" corner cell only, and $\sigma = 10^{-4}$. Problems 3a and 3b have homogeneous Neumann boundary conditions.

Tables VI and VII give results for Problem 3a for Case 1 and Case 2. The rate of convergence is independent of h. MGVC is about two times faster than MGCC. Many different values of ω were tried, with similar results.

Tables VIII and IX give results for Problem 3b in Case 1 and Case 2. We have the same conclusions as for Problem 3a, except that when h goes to zero we need to use transfer operators of type B instead of type A for MGVC, since type A gave divergence.

TABLE VIII

Results for Problem 3b, Case 1

		MGVC			MGCC		
h	ω	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/8	0.6875	3	0.003	0.005	6	0.097	0.085
1/12	0.6250	3	0.009	0.007	6	0.106	0.071
1/16	0.7188	3	0.011	0.005	6	0.121	0.091
1/24	0.6458	3	0.015	0.007	6	0.139	0.089
1/32	0.6719	4^a	0.025	0.017	6	0.153	0.100
1/64	0.6484	4^a	0.048	0.031	7	0.134	0.107

[a] Operator type B used with $p = 3$.

TABLE IX

Results for Problem 3b, Case 2

		MGVC			MGCC		
h	ω	ν	κ	$\bar{\kappa}$	ν	κ	$\bar{\kappa}$
1/8	0.6250	3	0.007	0.009	4	0.085	0.029
1/12	0.5833	3	0.009	0.009	6	0.055	0.074
1/16	0.6875	3	0.008	0.006	7	0.096	0.104
1/24	0.6250	4	0.019	0.012	7	0.130	0.112
1/32	0.6563	4[a]	0.026	0.023	6	0.131	0.099
1/64	0.6406	4[a]	0.049	0.040	6	0.148	0.095

[a] Operator type B used with $p = 3$.

TABLE XI

Preliminary Work

	Vertex-centered			Cell-centered		
	(a)	(b)	(c)	(a)	(d)	(e)
Coarse matrices	169+ 338 *	96+ 111 *	64+ 67 *	70+ 70 *	63+ 14 *	8+ 5 *
Transfer operators	24*	16+			—	

Note. (a) Using Algorithm RAP of Section 6; (b) using explicit expressions in [2]; (c) as (b), for the symmetric case; (d) using explicit expressions (6.10); (e) using explicit expressions (6.13), (6.17), (6.18) for the symmetric case.

8. STORAGE AND PRELIMINARY WORK

In this section estimates are presented of the storage and preliminary work requirements of the two methods. The preliminary work consists of the computation of the coarse grid matrices and, for the vertex-centered method, of the transfer operators. We consider only the coarse grids, as on the fine grid the storage requirement is nearly the same for the two methods.

Table X gives the number of reals to be stored divided by the number of grid points of the finest grid. The total number of coarse grid points is assumed to be $\frac{1}{4} + \frac{1}{16} + \cdots \simeq \frac{1}{3}$ times the number of grid points on the finest grid. Pointer calculations are neglected. In the symmetric case, the coarse grid matrices have five-point stencils, using (6.17) in the cell-centered case.

Table XI gives the preliminary work, counting operations (+ and $*$) per coarse grid point. In the cell-centered case the fact that $P^*(i, j) = 1$ is exploited, and in column (d) the fact that $e_i = w_i = 1$ in the interior has been used.

9. CONCLUDING REMARKS

The numerical tests show that MGVC is robust for interface, anisotropic, singular, and nearly singular problems. For certain nearly singular problems poor convergence and even divergence may occur when transfer operators type A are used. This is cured by using the transfer operators type B. MGCC also handles these problems well, but may require more iterations than MGVC.

We note the simplicity of the implementation of the restriction, prolongation, and Galerkin coarse grid approximation for MGCC. The amount of preliminary work and storage required is smaller in the cell-centered case than in the vertex-centered case.

TABLE X

Reals to Be Stored Divided by Number of Grid Points on the Finest Grid

	Vertex-centered		Cell-centered	
	General	Symmetric	General	Symmetric
Coarse matrices	3	5/3	7/3	1
Transfer operators	8/3	8/3	0	0

REFERENCES

1. R. E. Alcouffe, A. Brandt, J. E. Dendy, and J. W. Painter, *SIAM J. Sci. Stat. Comput.* **2**, 430 (1981).
2. A. Behie and P. A. Forsyth, *IMA J. Numer. Anal.* **3**, 41 (1983).
3. A. Brandt, *Math. Comput.* **31**, 333 (1977).
4. J. E. Dendy, *J. Comput. Phys.* **48**, 366 (1982).
5. J. E. Dendy, *SIAM J. Sci. Stat. Comput.* **8**, 673 (1987).
6. W. Hackbusch, *Multi-grid Methods and Applications* (Springer-Verlag, Berlin, 1985).
7. R. Kettler and J. A. Meijerink, Publication 604, KSEPL, Rijswijk, 1981 (unpublished).
8. R. Kettler, Ph.D. thesis, University of Technology, Department of Technical Math. and Informatics, Delft, 1987 (unpublished).
9. R. Kettler, in *Proceedings of the Conference on Multigrid Methods, Köln-Porz, West Germany, 1981*, edited by W. Hackbusch and U. Trottenberg, Lecture Notes in Mathematics, Vol. 960 (Springer-Verlag, Berlin, 1982), p. 502.
10. M. Khalil, Ph.D. thesis, University of Technology, Department of Technical Math. and Informatics, Delft, 1989 (unpublished).
11. P. Wesseling, in *Lecture Notes in Pure and Applied Mathematics*, Vol. 110, edited by S. F. McCormick (Dekker, New York, 1988), p. 631.
12. P. Wesseling, in *Proceedings, Fourth GAMM Seminar, Kiel, West Germany, January 1988*, edited by W. Hackbusch (Vieweg, Braunschweig, 1988).
13. P. Wesseling, *J. Comput. Phys.* **79**, 85 (1988).