

# Multigrid preconditioned conjugate-gradient solver for mixed finite-element method

John David Wilson · Richard L. Naff

Received: 30 January 2006 / Accepted: 13 July 2009 / Published online: 1 August 2009  
© The Author(s) 2009. This article is published with open access at Springerlink.com

**Abstract** The mixed finite-element approximation to a second-order elliptic PDE results in a saddle-point problem and leads to an indefinite linear system of equations. The mixed system of equations can be transformed into coupled symmetric positive-definite matrix equations, or a Schur complement problem, using block Gauss elimination. A preconditioned conjugate-gradient algorithm is used for solving the Schur complement problem. The mixed finite-element method is closely related to the cell-centered finite difference scheme for solving second-order elliptic problems with variable coefficients. For the cell-centered finite difference scheme, a simple multigrid algorithm can be defined and used as a preconditioner. For distorted grids, an additional iteration is needed. Nested iteration with a multigrid preconditioned conjugate gradient inner iteration results in an effective numerical solution technique for the mixed system of linear equations arising from a discretization on distorted grids. Numerical results show that the preconditioned conjugate-gradient inner iteration is robust with respect to grid size and variability in the hydraulic conductivity tensor.

**Keywords** Mixed finite element method · Lowest-order Raviart–Thomas · Subsurface flow · Multigrid · Distorted grids · Conjugate gradient · Nested iteration

## 1 Introduction

Lowest-order Raviart–Thomas mixed finite-element approximations to second-order elliptic PDEs with variable hydraulic conductivity coefficients are recognized as important methods for modeling subsurface flow in heterogeneous media [12, 14, 16]. In this paper, we consider the three-dimensional second-order elliptic problem given by

$$\begin{aligned} -\nabla \cdot \mathbf{K}\nabla p &= f \text{ in } \Omega, \\ p &= q \text{ on } \partial\Omega_D, \\ -\mathbf{K}\nabla p \cdot \mathbf{n} &= g \text{ on } \partial\Omega_N, \end{aligned} \quad (1)$$

where  $\Omega$  is a bounded domain in  $\mathfrak{R}^3$ ,  $p$  is a pressure or hydraulic head potential,  $\mathbf{n}$  denotes the unit outward normal vector,  $f$  is a sink/source term, and  $q$  and  $g$  are given functions.  $\mathbf{K}$  is a bounded, symmetric, and uniformly positive-definite matrix-valued function related to the intrinsic permeability of the porous media. The flux  $\mathbf{v}$  in subsurface flow models, given by the Darcy relation

$$\mathbf{v} = -\mathbf{K}\nabla p, \quad (2)$$

is often of primary interest. The use of a mixed method, which approximates both the hydraulic head and flux simultaneously, is desirable in the presence of rough coefficients since traditional methods, using

J. D. Wilson (✉) · R. L. Naff  
U.S. Geological Survey, Denver Federal Center,  
Box 25046, MS 413, Denver, CO 80225, USA  
e-mail: jwilson@math.cudenver.edu

R. L. Naff  
e-mail: rlnaff@usgs.gov

post-processing of primal hydraulic head solutions, generally lead to loss of accuracy when estimating the flux vector. Mixed methods also have the advantage of conserving mass locally, which can be critical in modeling subsurface flow.

Using the Darcy relation given by Eq. 2, problem 1 can be rewritten as a system of first-order equations. When discretized using the Raviart–Thomas finite-element spaces [25] on triangular or hexahedral elements, an indefinite system of equations results with the form given by

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{g} \\ \mathbf{f} \end{bmatrix}. \quad (3)$$

The matrix  $\mathbf{M}$  is real symmetric positive definite and, using the Raviart–Thomas finite element space, the discretization satisfies the inf-sup condition [10] so that  $\mathbf{B}^T$  has full column rank. In this paper, a lowest-order Raviart–Thomas finite element discretization on hexahedral grids is assumed.

Problem 3 can be transformed symbolically into coupled symmetric positive-definite matrix equations, or a Schur complement problem, using block Gauss elimination:

$$\begin{cases} \mathbf{S}_M \mathbf{p} = \mathbf{f} - \mathbf{B} \mathbf{M}^{-1} \mathbf{g} \\ \mathbf{M} \mathbf{v} = \mathbf{g} + \mathbf{B}^T \mathbf{p} \end{cases}, \quad (4)$$

where  $\mathbf{S}_M = \mathbf{B} \mathbf{M}^{-1} \mathbf{B}^T$ . A solution of this system of equations is possible but difficult;  $\mathbf{S}_M$  must be inverted to solve for  $\mathbf{p}$ , and subsequently,  $\mathbf{M}$  must be inverted to solve for  $\mathbf{v}$ . The system involving  $\mathbf{S}_M$  can be solved using the preconditioned conjugate gradient (PCG) method [17], but requires the evaluation of  $\mathbf{M}^{-1}$  for each PCG iteration and an effective preconditioner. These difficulties are overcome by approximating  $\mathbf{M}$  with a positive-definite matrix  $\mathbf{Q}_M$ , which is relatively easy to invert, and approximating the inverse of the resulting Schur complement  $\mathbf{S}_{Q_M} = \mathbf{B} \mathbf{Q}_M^{-1} \mathbf{B}^T$  using the PCG method preconditioned by cell-centered multigrid [6, 8, 30]. The system of equations in Eq. 4 is then solved using an iterative method preconditioned by these approximations. In this paper, we subsequently describe the nested iteration method where the pressure is approximated in the inner iteration and the flux is approximated in the outer iteration.

An alternative approach is the mixed-hybrid method. In this method, Lagrange multipliers are introduced on the cell boundaries to insure continuity of normal components of the velocity variable. The velocity and pressure variables are eliminated to obtain a symmetric and positive definite system of equations.

The system of equations can then be solved using an efficient multigrid method. A flux recovery technique for the computed solution is also needed. An example of the mixed-hybrid method and a flux recovery technique is given in [13]. A multigrid algorithm for the mixed method is described in [9].

The idea of nested iteration is widely used and can be found in the literature. The inexact Uzawa algorithm is studied in [7, 19] and is the motivation behind the nested iteration. We are also motivated by [2] as it concerns problems related to porous-media flows and employs a multigrid method. The iterative scheme in [2], for the mixed finite element method, is roughly equivalent to nested iteration. The convergence analysis given in [2] is desirable for its simplicity, and we make our analysis along these lines. An important aspect of the nested iteration described in this paper is the idea of using conjugate gradient as the inner iteration preconditioner. It is shown in [4] that the PCG approximation of the inverse of  $\mathbf{S}_{Q_M}$  is equivalent to the existence of another symmetric positive definite operator, which we will denote as  $\mathbf{Q}_B^{-1} \approx \mathbf{S}_{Q_M}^{-1}$ . We use this operator to define the nested iteration presented in this paper. The iterative convergence properties of nested iteration are further analyzed in [27, 28].

It is shown herein that nested iteration can be an efficient numerical solution of Eq. 3 with minimal fine tuning of inner and outer stopping criteria. Nested iteration in conjunction with PCG and cell-centered multigrid are used for the numerical solution. These methods are desirable due to their relative simplicity and efficiency. For the case of a diagonal conductivity tensor  $\mathbf{K}$  on an orthogonal hexahedral grid, the nested iteration algorithm reduces to the simpler PCG algorithm associated with Eq. 4. In this case, the matrix  $\mathbf{M}$  is tridiagonal and an exact factorization is obtained with minimal cost. The PCG algorithm can be used to approximately invert  $\mathbf{S}_M$  without directly forming  $\mathbf{B} \mathbf{M}^{-1} \mathbf{B}^T$  [14], and multigrid can be used to form an optimal preconditioner. The more general case occurs when the grid is distorted and/or  $\mathbf{K}$  is no longer a diagonal tensor; in this event,  $\mathbf{M}$ , containing a greater level of fill, is no longer tridiagonal. An incomplete factorization is used to approximate  $\mathbf{M}$  and nested iteration is used to solve Eq. 3.

In Section 2, the mixed finite-element method and its relationship to the cell-centered finite difference method is described. Section 3 defines the nested iteration algorithm with a convergence analysis given in Section 3.1 and a discussion of the conjugate-gradient inner-iteration given in Sections 3.2 and 3.2.1. Numerical results confirming the theoretical results are given in Section 4.

## 2 Mixed finite-element method (MFE)

Using Eq. 2, Eq. 1 can be rewritten as a system of first-order equations:

$$\begin{aligned} \mathbf{K}^{-1}\mathbf{v} + \nabla p &= 0 \\ \nabla \cdot \mathbf{v} &= f \\ p &= q, \text{ on } \partial\Omega_D, \\ \mathbf{v} \cdot \mathbf{n} &= g, \text{ on } \partial\Omega_N. \end{aligned} \tag{5}$$

The first equation in Eq. 5 is referred to as the momentum equation and the second as the continuity equation. The function spaces that contain  $(\mathbf{v}, p)$  are defined to be

$$\begin{aligned} \mathbf{V} &= \{\mathbf{v} \in L^2(\Omega)^3 : \nabla \cdot \mathbf{v} \in L^2(\Omega)\}, \\ \mathbf{V}_0 &= \{\mathbf{v} \in \mathbf{V} : \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega_N\} \\ P &= L^2(\Omega). \end{aligned}$$

Let  $\mathbf{v} \in \mathbf{V}$  be the velocity given by  $\mathbf{v} = \mathbf{v}_0 + \mathbf{v}_g$ , where  $\mathbf{v}_0 \in \mathbf{V}_0$  and  $\mathbf{v}_g \in \mathbf{V}$  satisfy the fourth equation in Eq. 5. The first equation in Eq. 5 is multiplied by  $\mathbf{w} \in \mathbf{V}_0$  and integrated by parts to produce a weak version of the momentum equation:

$$\int_{\Omega} \mathbf{K}^{-1}\mathbf{v}_0 \cdot \mathbf{w} - \int_{\Omega} p \nabla \cdot \mathbf{w} = - \int_{\partial\Omega_D} q \mathbf{w} \cdot \mathbf{n} - \int_{\Omega} \mathbf{K}^{-1}\mathbf{v}_g \cdot \mathbf{w}. \tag{6}$$

Multiplying the second equation in Eq. 5 by  $u \in P$  and integrating, the weak form of the continuity equation is obtained:

$$\int_{\Omega} (\nabla \cdot \mathbf{v}_0)u = \int_{\Omega} fu - \int_{\Omega} (\nabla \cdot \mathbf{v}_g)u. \tag{7}$$

The problem can be summarized using the bilinear forms corresponding to Eqs. 6 and 7 as follows:

$$\begin{cases} \text{Find } \mathbf{v} = \mathbf{v}_0 + \mathbf{v}_g \in \mathbf{V} \text{ and } p \in P \text{ such that:} \\ M(\mathbf{v}_0, \mathbf{w}) + B^*(p, \mathbf{w}) = -G(q, \mathbf{w}) - M(\mathbf{v}_g, \mathbf{w}), \\ B(\mathbf{v}_0, u) = L(f, u) - B(\mathbf{v}_g, u), \\ \forall (\mathbf{w}, u) \in \mathbf{V}_0 \times P. \end{cases} \tag{8}$$

When  $\partial\Omega_D = \emptyset$ , an additional compatibility condition, to guarantee existence of a solution, is needed:

$$B(\mathbf{v}_0, u) = 0, \forall u \in P.$$

The discretization of  $\Omega$  is defined on a logically cubic grid consisting of  $l \times m \times n$  hexahedral elements

with an  $i, j, k$  indexing and are trilinear mappings of the reference cube [20]. The finite-dimensional vector spaces  $\mathbf{V}^h \subset \mathbf{V}$  and  $\mathbf{V}_0^h \subset \mathbf{V}$  consist of vector functions, which are Piola transformations of lowest-order Raviart–Thomas vector functions on the reference cube [25]. A lowest-order Raviart–Thomas vector function on the reference cube is defined by

$$\hat{\mathbf{v}} = (a_0 + a_1\hat{x}, b_0 + b_1\hat{y}, c_0 + c_1\hat{z})^T,$$

and the degrees of freedom of  $\hat{\mathbf{v}}$  are chosen to be the values of  $\hat{\mathbf{v}} \cdot \mathbf{n}$  at the center of the faces of the reference cube. The nodal values of  $\mathbf{v} \in \mathbf{V}^h$  are taken to be the integrated fluxes across faces and are denoted by  $v_{i+1/2,j,k}^x, v_{j+1/2,k,i}^y$ , and  $v_{k+1/2,i,j}^z$ . An alternative to the Piola transformation is proposed in [22–24]; this alternative transformation leads to more accurate velocity interpolation in general hexahedral cells. The differences between the transformations are not important to the main topic of this paper.

The finite-dimensional scalar space  $P^h \subset P$  consists of piecewise constant functions with cell-centered nodal values  $p_{i,j,k}$ . Throughout this paper, we will assume that the hydraulic conductivity tensor is symmetric positive definite and constant within each element. The discrete divergence operator  $\mathbf{B}$  has  $\pm 1$  values for its components. The discrete adjoint of the divergence operator (the discrete gradient) is given by  $-\mathbf{B}^T$ . The  $\mathbf{B}$  and  $\mathbf{B}^T$  matrices are not explicitly assembled. The mass matrix  $\mathbf{M}$  is assembled element by element using a Gauss quadrature rule for the integration. The essential flux boundary conditions [21] can be dealt with in various ways. For example, by use of local to global ID arrays [20] or by application of nodal parameter boundary constraints [1]. The compressed row storage (CRS) [5] data structure can be used for storage of the mass matrix. In general, the finite-dimensional equivalent of problem 8 is given in matrix form by Eq. 3.

The number of off-diagonal elements of the matrix  $\mathbf{M}$  can vary, depending on distortions of the grid and fullness of the hydraulic conductivity tensor. For diagonal hydraulic conductivity tensors on orthogonal grids, the mass matrix is block tridiagonal, and satisfies

$$\mathbf{M}_{i,i} = 2 \sum_{j \neq i} \mathbf{M}_{i,j}. \tag{9}$$

Equation 9 implies  $\mathbf{M}$  is diagonally dominant with

$$\sum_{j \neq i} |\mathbf{M}_{i,j}| \leq 1/2 |\mathbf{M}_{i,i}|, \tag{10}$$

for all  $i$  independent of  $\mathbf{K}$ . A distorted grid or a full tensor  $\mathbf{K}$  can result in the loss of this diagonal dominance.

### 2.1 Cell-centered finite-difference matrix problem

In this section, the relationship between the mixed finite-element method and the cell-centered finite difference (CCFD) scheme is illuminated. The CCFD scheme is a special case of the mixed finite-element method when a diagonal conductivity tensor on an orthogonal grid is assumed and a low-order quadrature rule is used for integration. This analogy between the mixed finite-element method and CCFD is also given in [26].

For simplicity, a uniform partition of size  $h_x, h_y,$  and  $h_z$  in the  $x, y,$  and  $z$  directions is assumed. The hydraulic conductivity tensor on cell  $(i, j, k)$  is given by

$$\mathbf{K}_{i,j,k} = \begin{bmatrix} K_{i,j,k}^x & 0 & 0 \\ 0 & K_{i,j,k}^y & 0 \\ 0 & 0 & K_{i,j,k}^z \end{bmatrix}.$$

Let  $\mathbf{w} \in \mathbf{V}_0^h$  be a test function in the  $x$  direction with nodal value  $w_{i+1/2,j,k}^x = 1$  and zero otherwise. The discrete momentum equation (Eq. 6), upon integration using the trapezoidal rule and ignoring the boundary terms, becomes

$$\left( \Lambda_{i+1/2,j,k}^x \right)^{-1} v_{i+1/2,j,k}^x - (p_{i+1,j,k} - p_{i,j,k}) = 0,$$

where

$$\Lambda_{i+1/2,j,k}^x = \left( \frac{2h_x}{h_y h_z} \right) \left( \frac{K_{i,j,k}^x K_{i+1,j,k}^x}{K_{i,j,k}^x + K_{i+1,j,k}^x} \right).$$

This result represents the hydraulic conductance in the  $x$  direction between two adjacent cells and is consistent with the CCFD coefficients (see [18] for example). Similar expressions for difference equations in the  $y$  direction and  $z$  direction are obtained using  $y$  and  $z$  directional test functions. The discrete continuity equation (Eq. 7), under these conditions, becomes

$$\begin{aligned} f_{i,j,k} &= v_{i+1/2,j,k}^x - v_{i-1/2,j,k}^x \\ &+ v_{j+1/2,k,i}^y - v_{j-1/2,k,i}^y \\ &+ v_{k+1/2,i,j}^z - v_{k-1/2,i,j}^z. \end{aligned}$$

By substituting the appropriate expressions for  $v_{i+1/2,j,k}^x$ , etc., the following form results:

$$\begin{aligned} -f_{i,j,k} &= -\Lambda_{i-1,j,k}^x p_{i-1,j,k} - \Lambda_{i,j,k}^x p_{i+1,j,k} \\ &- \Lambda_{i,j-1,k}^y p_{i,j-1,k} - \Lambda_{i,j,k}^y p_{i,j+1,k} \\ &- \Lambda_{i,j,k-1}^z p_{i,j,k-1} - \Lambda_{i,j,k}^z p_{i,j,k+1} \\ &+ E_{i,j,k} p_{i,j,k} \end{aligned}$$

Here, the coefficients of the pressures correspond to the conductances of the CCFD scheme and

$$\begin{aligned} E_{i,j,k} &= \Lambda_{i-1,j,k}^x + \Lambda_{i,j,k}^x + \Lambda_{i,j-1,k}^y + \Lambda_{i,j,k}^y \\ &+ \Lambda_{i,j,k-1}^z + \Lambda_{i,j,k}^z. \end{aligned}$$

The CCFD scheme, in general, can be written as the Schur complement problem

$$\begin{cases} \mathbf{S}_A \mathbf{p} = \mathbf{f} - \mathbf{B} \Lambda^{-1} \mathbf{g} \\ \Lambda \mathbf{v} = \mathbf{g} + \mathbf{B}^T \mathbf{p} \end{cases} \quad (11)$$

where

$$\Lambda^{-1} = \left\{ \Lambda_{i+1/2,j,k}^x, \Lambda_{j+1/2,k,i}^y, \Lambda_{k+1/2,i,j}^z \right\}$$

and

$$\mathbf{S}_A = \mathbf{B} \Lambda^{-1} \mathbf{B}^T.$$

The CCFD matrix  $\mathbf{S}_A$  is symmetric positive definite (with appropriate boundary conditions or specified heads). An efficient cell-centered multigrid algorithm, well suited for solving the first equation in problem 11, is described in [6, 8, 15]. An implementation of this cell-centered multigrid algorithm as a PCG preconditioner is described in [30] for the solution of CCFD equations used in ground water modeling.

Let  $\mathbf{D} = \text{diag}(\mathbf{M})$ , where  $\mathbf{M}$  is obtained from a higher quadrature rule. For orthogonal grids with diagonal tensor  $\mathbf{K}$ , we have  $\Lambda^{-1} = (2/3)\mathbf{D}^{-1}$ . These results suggest cell-centered multigrid will make an effective preconditioner for  $\mathbf{S}_M$ .

### 3 Nested iteration

In this section, a nested iteration algorithm for solving Eq. 3 is presented. The nested iteration algorithm is derived from the following matrix splitting:

$$\begin{bmatrix} \mathbf{M} - \mathbf{B}^T \\ \mathbf{B} \quad \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_M & -\mathbf{B}^T \\ \mathbf{B} & \mathbf{Q}_B - \mathbf{S}_{Q_M} \end{bmatrix} - \begin{bmatrix} \mathbf{Q}_M - \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_B - \mathbf{S}_{Q_M} \end{bmatrix}$$

where  $\mathbf{Q}_M$  approximates  $\mathbf{M}$ ,  $\mathbf{S}_{Q_M} = \mathbf{B} \mathbf{Q}_M^{-1} \mathbf{B}^T$ , and  $\mathbf{Q}_B$  approximates  $\mathbf{S}_{Q_M}$ . The corresponding basic iteration is given by

$$\begin{bmatrix} \mathbf{v}^{k+1} \\ \mathbf{p}^{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^k \\ \mathbf{p}^k \end{bmatrix} + \begin{bmatrix} \mathbf{Q}_M & -\mathbf{B}^T \\ \mathbf{B} & \mathbf{Q}_B - \mathbf{S}_{Q_M} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{r}_v^k \\ \mathbf{r}_p^k \end{bmatrix},$$

where

$$\begin{bmatrix} \mathbf{r}_v^k \\ \mathbf{r}_p^k \end{bmatrix} = \begin{bmatrix} \mathbf{g} - (\mathbf{M} \mathbf{v}^k - \mathbf{B}^T \mathbf{p}^k) \\ \mathbf{f} - \mathbf{B} \mathbf{v}^k \end{bmatrix}.$$

Here,  $k$  denotes the iteration index. Let

$$\begin{bmatrix} \mathbf{Q}_M & -\mathbf{B}^T \\ \mathbf{B} & \mathbf{Q}_B - \mathbf{S}_{\mathbf{Q}_M} \end{bmatrix} \begin{bmatrix} \mathbf{u}_v^k \\ \mathbf{u}_p^k \end{bmatrix} = \begin{bmatrix} \mathbf{r}_v^k \\ \mathbf{r}_p^k \end{bmatrix}. \tag{12}$$

Using block-Gauss elimination to solve for  $\mathbf{u}_v^k$  and  $\mathbf{u}_p^k$  in Eq. 12, the algorithm for nested iteration follows:

---

**Algorithm 1** Nested iteration

---

1. Initialize  $\mathbf{v}^0$  and  $\mathbf{p}^0$ .
  2. For  $k = 0, \dots$ , until convergence Do:
    - a.  $\mathbf{r}_v^k = \mathbf{g} - (\mathbf{M}\mathbf{v}^k - \mathbf{B}^T\mathbf{p}^k)$ .
    - b.  $\mathbf{r}_p^k = \mathbf{f} - \mathbf{B}\mathbf{v}^k$ .
    - c.  $\mathbf{u}_p^k = \mathbf{Q}_B^{-1}(\mathbf{r}_p^k - \mathbf{B}\mathbf{Q}_M^{-1}\mathbf{r}_v^k)$ .
    - d.  $\mathbf{u}_v^k = \mathbf{Q}_M^{-1}(\mathbf{r}_v^k + \mathbf{B}^T\mathbf{u}_p^k)$ .
    - e.  $\mathbf{v}^{k+1} = \mathbf{v}^k + \mathbf{u}_v^k$ .
    - f.  $\mathbf{p}^{k+1} = \mathbf{p}^k + \mathbf{u}_p^k$ .
  3. EndDo.
- 

Nested iteration results from  $\mathbf{Q}_B^{-1}$  in step 2c, being an iterative approximation to  $\mathbf{S}_{\mathbf{Q}_M}^{-1}$ . We refer to the computation of  $\mathbf{Q}_B^{-1}$  as the inner iteration.

### 3.1 Outer iteration

The iteration error,  $\mathbf{e}^k = (\mathbf{e}_v^k, \mathbf{e}_p^k)^T$ , after the  $k$ -th iteration of algorithm 1, is related to the previous iteration error  $\mathbf{e}^{k-1}$  via

$$\mathbf{e}^k = \mathbf{G}\mathbf{e}^{k-1}, \tag{13}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{Q}_M & -\mathbf{B}^T \\ \mathbf{B} & \mathbf{Q}_B - \mathbf{S}_{\mathbf{Q}_M} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{Q}_M - \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_B - \mathbf{S}_{\mathbf{Q}_M} \end{bmatrix}. \tag{14}$$

In the next theorem, the iteration error for the velocity and pressure, respectively, are measured in terms of the following norms:

$$\|\mathbf{e}_v^k\|_{\mathbf{Q}_M}^2 = (\mathbf{e}_v^k)^T \mathbf{Q}_M \mathbf{e}_v^k, \quad \|\mathbf{e}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}}^2 = (\mathbf{e}_p^k)^T \mathbf{S}_{\mathbf{Q}_M} \mathbf{e}_p^k,$$

and

$$\|\mathbf{e}^k\|_{\mathbf{G}} = \max \left\{ \|\mathbf{e}_v^k\|_{\mathbf{Q}_M}, \|\mathbf{e}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}} \right\}. \tag{15}$$

The iterative convergence rate,  $\gamma$ , can be bounded by these norms and the following spectral radii:

$$\alpha = \rho(\mathbf{I} - \mathbf{Q}_M^{-1}\mathbf{M}) \tag{16}$$

and

$$\beta = \rho(\mathbf{I} - \mathbf{Q}_B^{-1}\mathbf{S}_{\mathbf{Q}_M}). \tag{17}$$

Equations 16 and 17 are equivalent to

$$\left| (\mathbf{Q}_M^{1/2}\mathbf{v})^T (\mathbf{I} - \mathbf{G}_V) \mathbf{Q}_M^{1/2}\mathbf{v} \right| \leq \alpha \|\mathbf{v}\|_{\mathbf{Q}_M}^2, \tag{18}$$

$$\left| (\mathbf{S}_{\mathbf{Q}_M}^{1/2}\mathbf{p})^T (\mathbf{I} - \mathbf{G}_P) \mathbf{S}_{\mathbf{Q}_M}^{1/2}\mathbf{p} \right| \leq \beta \|\mathbf{p}\|_{\mathbf{S}_{\mathbf{Q}_M}}^2, \tag{19}$$

$$\left| (\mathbf{S}_{\mathbf{Q}_M}^{1/2}\mathbf{p})^T \mathbf{G}_P \mathbf{S}_{\mathbf{Q}_M}^{1/2}\mathbf{p} \right| \leq (1 + \beta) \|\mathbf{p}\|_{\mathbf{S}_{\mathbf{Q}_M}}^2 \tag{20}$$

for all  $\mathbf{v}$  and  $\mathbf{p}$  where

$$\mathbf{G}_V = \mathbf{Q}_M^{-1/2} \mathbf{M} \mathbf{Q}_M^{-1/2} \tag{21}$$

$$\mathbf{G}_P = \mathbf{S}_{\mathbf{Q}_M}^{1/2} \mathbf{Q}_B^{-1} \mathbf{S}_{\mathbf{Q}_M}^{1/2}. \tag{22}$$

Also, the following relation derived from the Cauchy–Schwartz inequality is useful:

$$(\mathbf{v}, \mathbf{B}^T\mathbf{p}) = (\mathbf{Q}_M^{1/2}\mathbf{v}, \mathbf{Q}_M^{-1/2}\mathbf{B}^T\mathbf{p}) \leq \|\mathbf{v}\|_{\mathbf{Q}_M} \|\mathbf{p}\|_{\mathbf{S}_{\mathbf{Q}_M}}. \tag{23}$$

In the special case where  $\mathbf{Q}_M = \mathbf{M}$ , the iteration error is bounded by  $\beta$  only. In the other special case, where  $\mathbf{Q}_B = \mathbf{S}_{\mathbf{Q}_M}$ , the iteration error is bounded by  $\alpha$  only. In the general case, the iteration error is bounded by a composite of  $\alpha$  and  $\beta$ . A general result, similar to the one found in [2], is obtained in the following theorem:

**Theorem 1**

$$\|\mathbf{e}^k\|_{\mathbf{G}} \leq \gamma^k \|\mathbf{e}^0\|_{\mathbf{G}}, \tag{24}$$

where

$$\gamma = \alpha(1 + \beta) + \beta. \tag{25}$$

*Proof* From Eqs. 13 and 14, we have

$$\mathbf{Q}_M \mathbf{e}_v^k - \mathbf{B}^T \mathbf{e}_p^k = (\mathbf{Q}_M - \mathbf{M}) \mathbf{e}_v^{k-1}, \tag{26}$$

$$\mathbf{B} \mathbf{e}_v^k + (\mathbf{Q}_B - \mathbf{S}_{\mathbf{Q}_M}) \mathbf{e}_p^k = (\mathbf{Q}_B - \mathbf{S}_{\mathbf{Q}_M}) \mathbf{e}_p^{k-1}. \tag{27}$$

Multiplying Eq. 26 by  $\mathbf{B}\mathbf{Q}_M^{-1}$ , subtracting from Eq. 27, and multiplying by  $\mathbf{S}_{\mathbf{Q}_M}\mathbf{Q}_B^{-1}$  results in

$$\begin{aligned} \mathbf{S}_{\mathbf{Q}_M} \mathbf{e}_p^k &= \mathbf{S}_{\mathbf{Q}_M}^{1/2} (\mathbf{I} - \mathbf{G}_P) \mathbf{S}_{\mathbf{Q}_M}^{1/2} \mathbf{e}_p^{k-1} \\ &\quad - \mathbf{S}_{\mathbf{Q}_M} \mathbf{Q}_B^{-1} \mathbf{B} \mathbf{Q}_M^{-1/2} (\mathbf{I} - \mathbf{G}_V) \mathbf{Q}_M^{1/2} \mathbf{e}_v^{k-1}. \end{aligned} \tag{28}$$

Multiplying both sides of Eq. 28 by  $(\mathbf{e}_p^k)^T$ , using the triangle inequality, the Cauchy–Schwartz inequality, and Eqs. 18–20 give

$$\|\mathbf{e}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}} \leq \beta \|\mathbf{e}_p^{k-1}\|_{\mathbf{S}_{\mathbf{Q}_M}} + \alpha(1 + \beta) \|\mathbf{e}_v^{k-1}\|_{\mathbf{Q}_M}. \tag{29}$$

To obtain a bound on  $\|\mathbf{e}_v^k\|_{\mathbf{Q}_M}$ ,  $\tilde{\mathbf{e}}_v^k$  and  $\tilde{\mathbf{e}}_p^k$  are set to satisfy

$$\mathbf{Q}_M \tilde{\mathbf{e}}_v^k - \mathbf{B}^T \tilde{\mathbf{e}}_p^k = (\mathbf{Q}_M - \mathbf{M}) \mathbf{e}_v^{k-1}, \tag{30}$$

$$\mathbf{B} \tilde{\mathbf{e}}_v^k = 0. \tag{31}$$

This is equivalent to solving the inner iteration exactly on the  $k$ -th outer iteration (i.e.,  $\beta = 0$ ) and results in

$$\|\tilde{\mathbf{e}}_v^k\|_{\mathbf{Q}_M} \leq \alpha \|\mathbf{e}_v^{k-1}\|_{\mathbf{Q}_M}. \tag{32}$$

Taking the difference between Eqs. 26 and 30, then

$$\mathbf{Q}_M \mathbf{z}_v^k = \mathbf{B}^T \mathbf{z}_p^k, \tag{33}$$

where  $\mathbf{z}_v^k = \mathbf{e}_v^k - \tilde{\mathbf{e}}_v^k$  and  $\mathbf{z}_p^k = \mathbf{e}_p^k - \tilde{\mathbf{e}}_p^k$ . Multiplying both sides of Eq. 33 by  $(\mathbf{z}_v^k)^T$  and using Eq. 23, one obtains

$$\|\mathbf{z}_v^k\|_{\mathbf{Q}_M} \leq \|\mathbf{z}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}}. \tag{34}$$

The Schur complement of the system in Eqs. 30–31 can be expressed as

$$\mathbf{S}_{\mathbf{Q}_M} \tilde{\mathbf{e}}_p^k = -\mathbf{B} \mathbf{Q}_M^{-1/2} (\mathbf{I} - \mathbf{G}_V) \mathbf{Q}_M^{1/2} \mathbf{e}_v^{k-1}. \tag{35}$$

After subtracting Eq. 35 from Eq. 28, the Schur complement of  $\mathbf{z}_p^k$  is found to be

$$\begin{aligned} \mathbf{S}_{\mathbf{Q}_M} \mathbf{z}_p^k &= \mathbf{S}_{\mathbf{Q}_M}^{1/2} (\mathbf{I} - \mathbf{G}_P) \mathbf{S}_{\mathbf{Q}_M}^{1/2} \mathbf{e}_p^{k-1} \\ &+ \left[ \mathbf{S}_{\mathbf{Q}_M}^{1/2} (\mathbf{I} - \mathbf{G}_P) \mathbf{S}_{\mathbf{Q}_M}^{-1/2} \mathbf{B} \mathbf{Q}_M^{-1/2} \right] \\ &\times (\mathbf{I} - \mathbf{G}_V) \mathbf{Q}_M^{1/2} \mathbf{e}_v^{k-1}. \end{aligned} \tag{36}$$

Multiplying both sides of Eq. 36 by  $(\mathbf{z}_p^k)^T$  and applying the Cauchy–Schwartz inequality and inequalities 18 and 19 gives the result

$$\|\mathbf{z}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}} \leq \beta \|\mathbf{e}_p^{k-1}\|_{\mathbf{S}_{\mathbf{Q}_M}} + \beta \alpha \|\mathbf{e}_v^{k-1}\|_{\mathbf{Q}_M}. \tag{37}$$

Finally, using Eqs. 32, 34, 37, and the triangle inequality, one obtains

$$\begin{aligned} \|\mathbf{e}_v^k\|_{\mathbf{Q}_M} &= \|\tilde{\mathbf{e}}_v^k + \mathbf{e}_v^k - \tilde{\mathbf{e}}_v^k\|_{\mathbf{Q}_M} \\ &\leq \|\tilde{\mathbf{e}}_v^k\|_{\mathbf{Q}_M} + \|\mathbf{z}_v^k\|_{\mathbf{Q}_M} \\ &\leq \alpha (1 + \beta) \|\mathbf{e}_v^{k-1}\|_{\mathbf{Q}_M} + \beta \|\mathbf{e}_p^{k-1}\|_{\mathbf{S}_{\mathbf{Q}_M}}. \end{aligned} \tag{38}$$

The desired result follows immediately.  $\square$

### 3.2 Inner iteration

PCG iteration is used to obtain the action of  $\mathbf{Q}_B^{-1}$  in step 2c of Algorithm 1. That is, if  $\mathbf{S}_{\mathbf{Q}_M}^{-1}$  represents the

inversion of  $\mathbf{S}_{\mathbf{Q}_M}$  by the PCG algorithm, then  $\mathbf{Q}_B^{-1}$  is obtained by relaxing the convergence of the PCG algorithm so that only  $m$  iterations are required. As these  $m$  PCG iterations are used for each outer iteration  $k$ , then  $\mathbf{Q}_B^{-1}$  represents  $m(k)$  iterations of the PCG algorithm. Because of the afore-noted similarities with the CCFD matrix, the cell-centered multigrid algorithm is used as the preconditioner. The cell-centered multigrid algorithm approximates the inverse of

$$\mathbf{S}_D = \mathbf{B} \mathbf{D}^{-1} \mathbf{B}^T, \tag{39}$$

where

$$\mathbf{D} = \text{diag}(\mathbf{Q}_M). \tag{40}$$

Matrix-vector multiplications of the form  $\mathbf{y} = \mathbf{B} \mathbf{Q}_M^{-1} \mathbf{B}^T \mathbf{x}$  are also required in the PCG algorithm. This multiplication is performed sequentially by computing the action of  $\mathbf{B}^T$  followed by the action of  $\mathbf{Q}_M^{-1}$  followed by the action of  $\mathbf{B}$ .

The approximation  $\mathbf{Q}_M$  of  $\mathbf{M}$  is obtained from a level-0 incomplete Cholesky factorization ( $ILU(0)$ ) of  $\mathbf{M}$ :

$$\mathbf{Q}_M = (\mathbf{L}_0 + \mathbf{D}_0) \mathbf{D}_0^{-1} (\mathbf{D}_0 + \mathbf{L}_0^T)$$

where  $\mathbf{L}_0$  is lower triangular and  $\mathbf{D}_0$  is diagonal. The action of  $\mathbf{Q}_M^{-1}$  on a vector is obtained, essentially, from a back substitution of  $\mathbf{L}_0$  followed by a forward substitution of  $\mathbf{L}_0^T$ . When  $\mathbf{M}$  is tridiagonal, then  $\mathbf{Q}_M$  obtained by this procedure is equivalent to  $\mathbf{M}$  and, effectively,  $\mathbf{Q}_M^{-1} = \mathbf{M}^{-1}$ . In this event, if  $\mathbf{Q}_B^{-1}$  is sufficiently close to  $\mathbf{S}_{\mathbf{Q}_M}^{-1}$  (that is, a sufficient number of PCG iterations  $m$  are used), then only a single outer iteration is necessary to obtain a solution to Eq. 3; the algorithm is then equivalent to the PCG algorithm presented in [14].

Computing the action of  $\mathbf{Q}_B^{-1}$  in Algorithm 1 is accomplished using incomplete PCG iteration; as noted previously, this application of PCG represents the inner iteration of the nested iteration algorithm. Because  $\beta$  of Theorem 1 encompasses the inversion of  $\mathbf{Q}_B$ , the convergence factor  $\delta$  of the PCG algorithm is examined and related to  $\beta$ . In this regard,  $\mathbf{u}_p^k$  is an approximation to the exact solution  $\tilde{\mathbf{u}}_p^k$  of the linear system

$$\mathbf{S}_{\mathbf{Q}_M} \tilde{\mathbf{u}}_p^k = \mathbf{r}_p^k - \mathbf{B} \mathbf{Q}_M^{-1} \mathbf{r}_v^k \tag{41}$$

and the following result is obtained [4]:

**Theorem 2** *If  $\|\tilde{\mathbf{u}}_p^k - \mathbf{u}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}} \leq \zeta \|\tilde{\mathbf{u}}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}}$ , then there exists a symmetric positive definite matrix  $\mathbf{Q}_B$  such that  $\mathbf{Q}_B \mathbf{u}_p^k = \mathbf{S}_{\mathbf{Q}_M} \tilde{\mathbf{u}}_p^k$  and  $\rho(\mathbf{I} - \mathbf{Q}_B^{-1} \mathbf{S}_{\mathbf{Q}_M}) \leq \zeta$ .*

*Proof* See Lemma 4.2 in [4]  $\square$

If  $m(k)$  PCG iterations are needed to approximate the solution to Eq. 41 with an initial guess of zero and a convergence factor of  $\delta$ , then the well-known bound [17, 31] given by

$$\|\tilde{\mathbf{u}}_p^k - \mathbf{u}_p^k\|_{\mathbf{S}_{Q_M}} \leq 2\delta^{m(k)} \|\tilde{\mathbf{u}}_p^k\|_{\mathbf{S}_{Q_M}} \tag{42}$$

applies to the convergence of Eq. 41. Theorem 2 and relation 42 provide a bound such that  $\beta \leq 2\delta^{m(k)}$ . As  $\delta < 1$  for convergence of the PCG algorithm,  $\beta$  becomes small as  $m(k)$  increases. If  $m(k)$  is allowed to become sufficiently large, then convergence of Algorithm 1 will depend on  $\alpha$  alone.

### 3.2.1 CCFD preconditioner

In this section, a bound for the convergence factor  $\delta$  of the PCG method is obtained when applied to Eq. 41 while using CCFD preconditioning. The condition number  $\kappa(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M})$  is defined by

$$\kappa(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M}) = \frac{\lambda_{\max}(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M})}{\lambda_{\min}(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M})}. \tag{43}$$

It is well known [17, 31] that the convergence factor  $\delta$  of the PCG algorithm is bounded by

$$\delta \leq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}. \tag{44}$$

In the following theorem, the condition number of  $(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M})$  is bounded in terms of the condition number of  $(\mathbf{D}^{-1}\mathbf{Q}_M)$ :

### Theorem 3

$$\kappa(\mathbf{S}_D^{-1}\mathbf{S}_{Q_M}) \leq \kappa(\mathbf{D}^{-1}\mathbf{Q}_M). \tag{45}$$

*Proof* Let  $\lambda_{\min}$  and  $\lambda_{\max}$  be the minimum and maximum eigenvalues of  $\mathbf{D}^{-1}\mathbf{Q}_M$ , respectively. Then, the following are equivalent for all  $\mathbf{v}$

$$\begin{aligned} \lambda_{\min} \mathbf{v}^T \mathbf{v} &\leq \mathbf{v}^T \mathbf{D}^{-1} \mathbf{Q}_M \mathbf{v} \leq \lambda_{\max} \mathbf{v}^T \mathbf{v} \\ \lambda_{\min} \mathbf{v}^T \mathbf{v} &\leq \mathbf{v}^T \mathbf{Q}_M^{1/2} \mathbf{D}^{-1} \mathbf{Q}_M^{1/2} \mathbf{v} \leq \lambda_{\max} \mathbf{v}^T \mathbf{v} \\ \lambda_{\min} \mathbf{v}^T \mathbf{Q}_M^{-1} \mathbf{v} &\leq \mathbf{v}^T \mathbf{D}^{-1} \mathbf{v} \leq \lambda_{\max} \mathbf{v}^T \mathbf{Q}_M^{-1} \mathbf{v} \\ \lambda_{\max}^{-1} \mathbf{v}^T \mathbf{D}^{-1} \mathbf{v} &\leq \mathbf{v}^T \mathbf{Q}_M^{-1} \mathbf{v} \leq \lambda_{\min}^{-1} \mathbf{v}^T \mathbf{D}^{-1} \mathbf{v} \end{aligned}$$

Letting  $\mathbf{v} = \mathbf{B}^T \mathbf{S}_D^{-1/2} \mathbf{p}$  finishes the proof. □

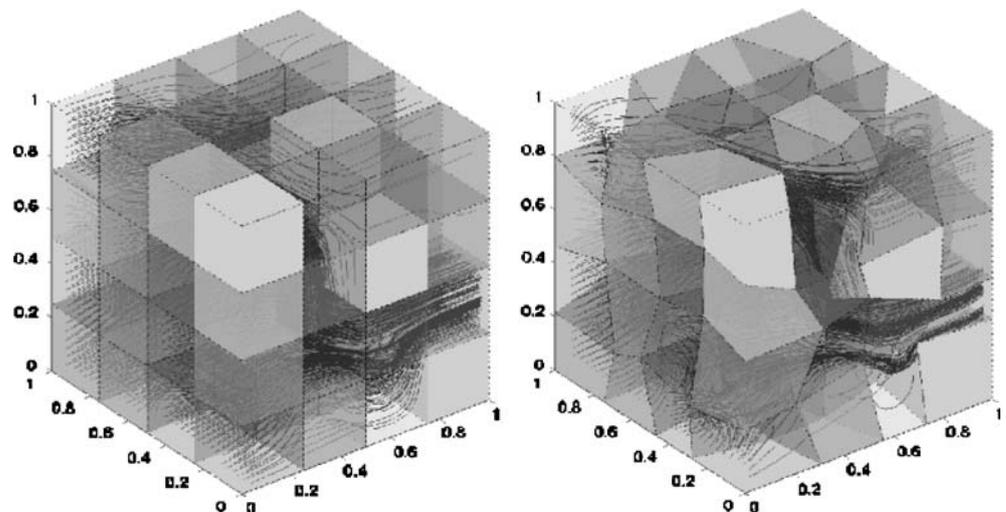
In the case of a tridiagonal matrix  $\mathbf{M}$ ,  $\mathbf{Q}_M = \mathbf{M}$  and Algorithm 1 reduces to the method suggested in [14] for the direct solution of Eq. 4 using the PCG method. From Eq. 9 and the Gershgorin circle theorem [17], the convergence factor of the PCG algorithm is

$$\delta \leq \frac{\sqrt{3} - 1}{\sqrt{3} + 1} \approx 0.27. \tag{46}$$

### 3.3 Adaptive nested iteration

Forcing  $\beta$  to become small by increasing the number of PCG iterations is costly because each PCG iteration

**Fig. 1** Flow lines obtained from computed fluxes. Orthogonal grid on the left and distorted grid on the right



**Table 1** The convergence factor  $\delta$  and number of iterations for PCG algorithm preconditioned by cell-centered multigrid with scalar conductivity on orthogonal grid

$s$	$\delta$	Iterations
2	0.206	23
3	0.231	25
4	0.254	26
5	0.266	27
6	0.274	28

requires the action of  $\mathbf{Q}_M^{-1}$  and the approximation of  $\mathbf{S}_D^{-1}$  using multigrid. For a fixed  $\alpha < 1$ , Theorem 1 implies that Algorithm 1 is convergent for

$$\beta < \frac{1 - \alpha}{1 + \alpha}. \tag{47}$$

Condition 47 is satisfied by letting

$$\beta = \alpha \left( \frac{1 - \alpha}{1 + \alpha} \right). \tag{48}$$

Substituting Eq. 48 into Eq. 25 results in the following reduction rate for Algorithm 1:

$$\gamma = \alpha(2 - \alpha). \tag{49}$$

Using definitions from Algorithm 1, let

$$\alpha_k = \frac{\|(\mathbf{I} - \mathbf{M}\mathbf{Q}_M^{-1})\mathbf{r}_v^k\|_{\mathbf{Q}_M}}{\|\mathbf{r}_v^k\|_{\mathbf{Q}_M}}, \tag{50}$$

$$\beta_k = \alpha_k \left( \frac{1 - \alpha_k}{1 + \alpha_k} \right), \tag{51}$$

$$\mathbf{z}_p^k = \mathbf{r}_p^k - \mathbf{B}\mathbf{Q}_M^{-1}\mathbf{r}_v^k,$$

and

$$\mathbf{u}_p^k = PCG(\mathbf{S}_{\mathbf{Q}_M}, \mathbf{z}_p^k, \beta_k) \tag{52}$$

**Table 2** Power method estimates of  $\alpha$ ,  $\lambda_{\max}(\mathbf{D}^{-1}\mathbf{Q}_M)$ , and  $\lambda_{\min}(\mathbf{D}^{-1}\mathbf{Q}_M)$  for the 25% distorted grid case with variable permeability

$s$	$\alpha$	$\lambda_{\max}$	$\lambda_{\min}$	$\delta$
2	0.06	2.12	0.47	0.36
3	0.10	2.34	0.47	0.38
4	0.17	2.53	0.47	0.40
5	0.27	2.68	0.40	0.44
6	0.40	2.79	0.35	0.48

The value of  $\delta$  is estimated from Theorem 3

**Table 3** Values of  $\gamma$ ,  $\alpha$ , and  $\delta$  for the 25% distorted grid case of Algorithm 1

$s$	$\gamma$	$\alpha$	$\delta$	Iterations		CPU
				Outer	Inner	
2	0.02	0.04	0.23	6	102	0
3	0.03	0.07	0.25	7	128	0
4	0.05	0.11	0.29	8	160	2
5	0.12	0.19	0.32	11	220	26
6	0.20	0.30	0.34	15	301	314

Execution times (CPU) given in seconds rounded to zero decimal places

where Eq. 52 is the approximate solution of Eq. 41 using the PCG method such that the preconditioned norm of the residual is reduced by a factor less than or equal to  $\beta_k$ . The actual reduction factor  $\beta_k$  can be estimated as follows:

$$\beta_k \approx \frac{\|\mathbf{z}_p^k - \mathbf{S}_{\mathbf{Q}_M}\mathbf{u}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}}}{\|\mathbf{z}_p^k\|_{\mathbf{S}_{\mathbf{Q}_M}}}. \tag{53}$$

#### 4 Numerical results

The model problem is a cubic domain with random heterogeneity in the hydraulic conductivity coefficient and a flow induced by a pressure gradient imposed on two opposite boundaries. The other boundaries have no-flow conditions. The hydraulic conductivity tensor,  $\mathbf{K} = a(e)\mathbf{I}$ , is aligned with a not necessarily orthogonal  $4 \times 4 \times 4$  coarse-grid, and in each block,  $e$  has a scalar value  $a(e) = 10^{-p}$  where  $p$  is a random integer such that  $0 \leq p \leq 5$ . Numerical approximations of the fluxes and pressures are obtained from Algorithm 1 on different-size grids with diameter  $h = 2^{-s}$  for  $s = 2, 3, 4, 5, 6$ . Flow lines computed from the simulated fluxes are illustrated in Fig. 1. As previously noted, the action of  $\mathbf{Q}_B^{-1}$  is best computed by the PCG algorithm preconditioned

**Table 4** Values of  $\gamma$ ,  $\alpha$ , and  $\beta$  for the 25% distorted grid case of Algorithm 1 using one inner iteration for each outer iteration

$s$	$\gamma$	$\alpha$	$\beta$	Iterations		CPU
				Outer	Inner	
2	0.44	0.03	0.48	30	30	0
3	0.44	0.04	0.44	30	30	0
4	0.50	0.07	0.50	36	36	3
5	0.54	0.08	0.55	41	41	31
6	0.59	0.09	0.59	48	48	315

Execution times (CPU) given in seconds rounded to zero decimal places

**Table 5** Values of  $\gamma$ ,  $\alpha$ , and  $\beta$  for the 25% distorted grid case of Algorithm 1 using adaptive inner iteration tolerance

$s$	$\gamma$	$\alpha$	$\beta$	Iterations		CPU
				Outer	Inner	
2	0.02	0.03	0.02	6	27	0
3	0.03	0.06	0.03	7	31	0
4	0.05	0.11	0.05	8	37	0
5	0.12	0.18	0.06	12	44	11
6	0.29	0.30	0.06	15	59	124

Execution times (CPU) given in seconds rounded to zero decimal places

by cell-centered multigrid. The cell-centered multigrid preconditioning consists of one V-Cycle [11, 29] with  $D - ILU$  smoothing as described in [5, 30]. The  $D - ILU$  factorization is a simpler version of  $ILU(0)$ . Not only does it have zero fill-in, it ignores the off-diagonal modifications and only the pivots are stored in the diagonal. For CCFD matrices, the  $D - ILU$  factorization coincides with the  $ILU(0)$  factorization.

#### 4.1 Orthogonal grids

In the case of an orthogonal grid with a diagonal conductivity tensor, the matrix  $\mathbf{M}$  is tridiagonal and has a simple factorization. The Schur complement problem (Eq. 4) is solved using the PCG algorithm preconditioned by cell-centered multigrid. The norm of the error is approximated by the actual residuals  $\mathbf{r}^k$  within the PCG loop:

$$\|\mathbf{e}^k\|_{\mathbf{S}_{\mathbf{Q}_M}} = \sqrt{(\mathbf{r}^k, \mathbf{e}^k)} \approx \beta_k,$$

where  $\beta_k = \sqrt{(\mathbf{r}^k, \mathbf{S}_D^{-1}\mathbf{r}^k)}$ . The convergence factor  $\delta$  shown in Table 1 agrees closely with the theoretical

bound  $\delta \approx 0.27$  of Theorem 3 and Eq. 46. The  $\delta$  shown in this table is approximated by

$$\delta \approx \sqrt[m]{\frac{\beta_m}{2\beta_0}},$$

where  $m$  is the number of PCG iterations such that  $\beta_m \leq 10^{-15}$ .

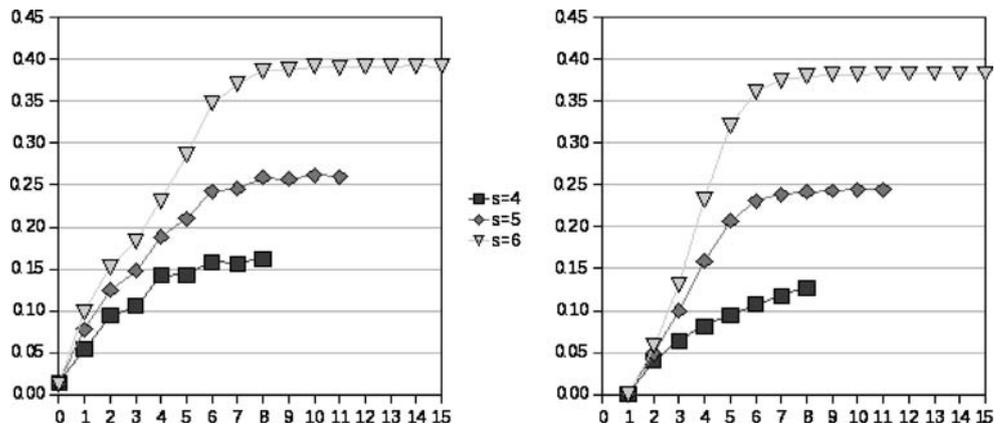
#### 4.2 Distorted grids

As noted earlier, the matrix  $\mathbf{M}$  for a distorted grid is not tridiagonal because of additional fill. In this event, an approximation  $\mathbf{Q}_M$  of  $\mathbf{M}$  is used;  $\mathbf{Q}_M$  takes the form of a level-0 incomplete Cholesky factorization of  $\mathbf{M}$  and nested iteration results. The cell faces on the boundary of the domain are kept rectangular, but the interior intersections of the hexahedral cells are allowed to be distorted from their orthogonal position by up to 25% of the orthogonal cell diameter. The distorted grid is illustrated in the right-hand side of Fig. 1. To estimate a bound on  $\alpha$  and  $\beta$ , as applied in Theorem 1, we use the power method [3, 17]. The power method is used to estimate the dominant eigenvalues of  $\mathbf{I} - \mathbf{Q}_M^{-1}\mathbf{M}$ ,  $\mathbf{D}^{-1}\mathbf{Q}_M$ , and  $\mathbf{Q}_M^{-1}\mathbf{D}$  in order to estimate  $\alpha$ ,  $\lambda_{\max}(\mathbf{D}^{-1}\mathbf{Q}_M)$ , and  $\lambda_{\min}(\mathbf{D}^{-1}\mathbf{Q}_M)$ , respectively. The last two estimates are used in Theorem 3 to obtain a theoretical estimate of  $\delta$ . These estimates are given in Table 2.

The preconditioned residual,  $\mathbf{u}^k = (\mathbf{u}_v^k, \mathbf{u}_p^k)^T$ , after the  $k$ -th iteration of Algorithm 1 is related to the previous preconditioned residual  $\mathbf{u}^{k-1}$  via  $\mathbf{u}^k = \mathbf{G}\mathbf{u}^{k-1}$ . The reduction factor,  $\gamma_k$ , on the  $k$ -th iteration is approximated by

$$\gamma_k \approx \frac{\|\mathbf{u}^{k-1}\|_G}{\|\mathbf{u}^k\|_G}.$$

**Fig. 2** Iteration histories of  $\alpha_k$  (on left) and  $\gamma_k$  (on right) on distorted grid with grid diameter  $h = 2^{-s}$  for  $s = 4, 5, 6$



The values of  $\gamma$  shown in Tables 3 and 4 are approximated by

$$\gamma \approx \left( \frac{\|\mathbf{u}^m\|_G}{\|\mathbf{u}^0\|_G} \right)^{1/m},$$

where  $m$  is the number of outer iterations such that  $\|\mathbf{u}^m\|_G \leq 10^{-10}$ . The values of  $\alpha$  in Tables 3, 4, and 5 and the values of  $\beta$  in Tables 4 and 5 are the arithmetic means of  $\alpha_k$  and  $\beta_k$  from Eqs. 50 and 53, respectively.

The numerical results obtained from nested iteration and the results obtained in Table 2 and the theoretical bounds from Theorem 1 can be compared. Theoretically, the convergence rate,  $\delta$ , of the PCG method in the inner iteration should be bounded by the value of  $\delta$  given in Table 2. If a sufficient number of inner iterations are allowed so that  $\mathbf{Q}_B^{-1}$  is a close approximation to  $\mathbf{S}_{\mathbf{Q}_M}^{-1}$ , then the outer convergence rate,  $\gamma$ , of Algorithm 1 should be bounded by the value of  $\alpha$  given in Table 2 since Theorem 1 indicates that  $\beta$  should become negligible. The values in Table 3 can be compared with this theoretical result. The iteration histories of  $\alpha_k$  and  $\gamma_k$ , shown in Fig. 2 for  $s = 4, 5, 6$ , indicate that these values converge to the theoretical limit.

Tables 4 and 5 show the results of limiting the number of PCG iterations. In Table 4, the number of inner iterations is limited to one for each outer iteration. This results in a reduction factor of  $\beta$  sufficient for Algorithm 1 to converge. Although the number of outer iterations increase, the CPU times remain approximately the same (compare with outer iterations and CPU times in Table 3). The convergence factor  $\gamma$  in Table 4 agrees with the theoretical results from Theorem 1.

Table 5 shows the results of using the adaptive approach described in Section 3.3. The total number of inner iterations is significantly smaller, compared to Table 3, while the number of outer iterations remain the same. This results in the best CPU times of all three comparisons.

**Table 6** Power method estimates of  $\alpha$ ,  $\lambda_{\max}(\mathbf{D}^{-1}\mathbf{Q}_M)$ , and  $\lambda_{\min}(\mathbf{D}^{-1}\mathbf{Q}_M)$  for the 35% distorted grid case with variable permeability

$s$	$\alpha$	$\lambda_{\max}$	$\lambda_{\min}$	$\delta$
2	0.08	2.35	0.41	0.41
3	0.17	2.63	0.38	0.45
4	0.32	2.90	0.28	0.53
5	0.54	3.09	0.18	0.61
6	0.88	3.29	0.14	0.66

The value of  $\delta$  is estimated from Theorem 3

**Table 7** Values of  $\gamma$ ,  $\alpha$ , and  $\beta$  for the 35% distorted grid case of Algorithm 1 using adaptive inner iteration tolerance

$s$	$\gamma$	$\alpha$	$\beta$	Iterations		CPU
				Outer	Inner	
2	0.03	0.05	0.03	8	26	0
3	0.06	0.10	0.05	10	26	0
4	0.12	0.22	0.06	13	32	1
5	0.37	0.49	0.06	26	55	22
6	0.67	0.78	0.05	57	122	421

Execution times (CPU) given in seconds rounded to zero decimal places

### 4.3 Further grid distortion

A higher degree of grid distortion can negatively impact the effectiveness of the outer iteration preconditioner  $\mathbf{Q}_M$  and also effect the inner iteration preconditioner  $\mathbf{Q}_B$ . We experienced a failure in the  $ILU(0)$  factorization when testing a grid distortion of up to 45% with  $h = 2^{-3}$ . We are able to test a grid distortion of up to 35%. Table 6 shows that the value of  $\alpha$  can become detrimentally large in this case and the results of Algorithm 1 using adaptive inner iteration tolerance is shown in Table 7. The numerical examples presented in this paper are based on random examples. Further testing is needed with more practical examples, and more robust preconditioners might need to be investigated.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

### References

- Akin, J.E.: Application and Implementation of Finite Element Methods, pp. 153–158. Academic, London (1982)
- Allen, M.B., Ewing, R.E., Lu, P.: Well-conditioned iterative schemes for mixed finite-element models of porous-media flows. *SIAM J. Sci. Statist. Comput.* **13**(3), 794–814 (1992)
- Atkinson, K.E.: An Introduction to Numerical Analysis. Wiley, New York (1988)
- Bank, R., Welfert, B., Yserentant, H.: A class of iterative methods for solving saddle point problems. *Numer. Math.* **55**, 645–666 (1990)
- Barrett, R., Berry, M., Chan, T.F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C., der Vorst, H.V.: Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd edn. SIAM, Philadelphia (1994). [www.netlib.org/linalg/html\\_templates/report.html](http://www.netlib.org/linalg/html_templates/report.html)
- Bramble, J., Ewing, R., Pasciak, J., Shen, J.: The analysis of multigrid algorithms for cell centered finite difference methods. *Adv. Comput. Math.* **5**(1), 15–29 (1996)

7. Bramble, J., Pasciak, J., Apostol, T.: Analysis of the inexact uzawa algorithm for saddle point problems. *SIAM J. Numer. Anal.* **34**(3), 1072–1092 (1987)
8. Bramble, J., Pasciak, J., Xu, J.: The analysis of multigrid algorithms with nonnested spaces or noninherited quadratic forms. *Math. Comput.* **56**(193), 1–34 (1991)
9. Brenner, S.C.: A multigrid algorithm for the lowest-order Raviart–Thomas mixed triangular finite element method. *SIAM J. Numer. Anal.* **29**(3), 647–678 (1992)
10. Brezzi, F., Fortin, M.: *Mixed and Hybrid Finite Element Methods*. Springer, New York (1991)
11. Briggs, W.L.: *A Multigrid Tutorial*. Siam, Philadelphia (1987)
12. Cai, Z., Jones, J., McCormick, S., Russell, T.: Control-volume mixed finite elements methods. *Comput. Geosci.* **1**, 289–315 (1997)
13. Chou, S., Kwak, D.Y., Kim, K.Y.: Flux recovery from primal hybrid finite element methods. *SIAM J. Numer. Anal.* **40**(2), 403–415 (2003)
14. Dougherty, D.: PCG solutions of flow problems in random porous media using mixed finite elements. *Adv. Water Resour.* **13**(1) (1990)
15. Ewing, R., Shen, J.: A multigrid algorithm for the cell-centered finite difference scheme. In: *The Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*. NASA Conference Publication 3224 (1993)
16. Ewing, R., Wheeler, M.: Computational aspects of mixed finite element methods. In: *Numerical Methods for Scientific Computing*, pp. 163–172 (1983)
17. Golub, G.H., Van Loan, C.F.: *Matrix Computations*. John Hopkins University Press, Baltimore (1989)
18. Harbaugh, A.W., Banta, E.R., Hill, M.C., McDonald, M.G.: *Modflow-2000, the U.S. geological survey modular ground-water model user guide to modularization concepts and the ground-water flow process*. Tech. rep., U.S. Geological Survey. Open-File Report 00-92 (2000)
19. Howard, C.E., Golub, G.H.: Inexact and preconditioned uzawa algorithms for saddle point problems. *SIAM J. Numer. Anal.* **31**(6), 1645–1661 (1994)
20. Hughes, T.: *The Finite Element Method*, pp. 123–125. Prentice-Hall, Englewood Cliffs (1987)
21. Johnson, C.: *Numerical Solutions of Partial Differential Equations by the Finite Element Method*, pp. 141–144. Cambridge University Press, Cambridge (1987)
22. Naff, R., Russell, T., Wilson, J.: Test functions for three-dimensional control-volume finite-element methods on irregular grids. In: *Computational Methods in Water Resources*, vol. 2, pp. 677–684 (2000). [http://wwwbrr.cr.usgs.gov/projects/GW\\_stoch/index.html](http://wwwbrr.cr.usgs.gov/projects/GW_stoch/index.html)
23. Naff, R., Russell, T., Wilson, J.: Shape functions for three-dimensional control-volume mixed finite-element methods on irregular grids. In: *Computational Methods in Water Resources*, pp. 359–366 (2002). [http://wwwbrr.cr.usgs.gov/projects/GW\\_stoch/index.html](http://wwwbrr.cr.usgs.gov/projects/GW_stoch/index.html)
24. Naff, R., Russell, T.R., Wilson, J.: Shape functions for velocity interpolation in general hexahedral cells. *Comput. Geosci.* **6**, 285–314 (2002). [http://wwwbrr.cr.usgs.gov/projects/GW\\_stoch/index.html](http://wwwbrr.cr.usgs.gov/projects/GW_stoch/index.html)
25. Raviart, P., Thomas, J.: A mixed finite element method for 2nd order elliptic problems. In: Galligani, I., Magenes, E. (eds.) *Mathematical Aspects of Finite Element Methods*. Lecture Notes in Mathematics, pp. 292–315. Springer, New York (1977)
26. Russell, T.F., Wheeler, M.F.: Finite element and finite difference methods for continuous flows in porous media. In: Ewing, R.E. (ed.) *The Mathematics of Reservoir Simulation*, pp. 35–106. Society of Industrial and Applied Mathematics, Philadelphia (1983)
27. Sameh, A., Baggag, A.: Nested iterative schemes for indefinite linear systems. In: Mang, H.A., Rammerstorfer, F.G., Eberhardsteiner, J. (eds.) *Fifth World Congress on Computational Mechanics* (2002)
28. Tong, Z., Sameh, A.: On an iterative method for saddle point problems. *Numer. Math.* **79**, 643–646 (1998)
29. Trottenberg, U., Oosterlee, C., Schüller, A.: *Multigrid*. Academic, London (2001)
30. Wilson, J., Naff, R.: *Modflow-2000, the U.S. Geological Survey modular ground-water model – GMG linear equation solver package documentation*. Tech. rep., U.S. Geological Survey (2004). <http://pubs.water.usgs.gov/ofr2004-1261/>
31. Xu, J.: Iterative methods by space decomposition and subspace corrections. *SIAM Rev.* **34**(4), 581–613 (1992)