Efficient Mixed Methods for Groundwater Flow on Triangular or Tetrahedral Meshes

Todd Arbogast
Clint N. Dawson
Philip T. Keenan

CRPC-TR94377
January, 1994

Center for Research on Parallel Computation
Rice University
P.O. Box 1892
Houston, TX 77251-1892

Revised June, 1994.

EFFICIENT MIXED METHODS FOR GROUNDWATER FLOW ON TRIANGULAR OR TETRAHEDRAL MESHES

TODD ARBOGAST, CLINT N. DAWSON, and PHILIP T. KEENAN
Department of Computational and Applied Mathematics, Rice University,
P.O. Box 1892, Houston, TX 77251–1892, U.S.A.

Simulating flow in porous media requires the solution of elliptic or parabolic partial differential equations. When the computational domain is irregularly shaped, applying finite element methods with triangular elements offers great flexibility.

The mixed finite element method has proven useful for solving flow equations. The difficulty with mixed methods in general is in solving the linear algebraic systems that arise. On rectangular elements, the mixed method with lowest-order approximating spaces can be reduced to a simple finite difference method in the primary variable, thus reducing the linear system to a sparse, symmetric, positive (semi-)definite matrix, for which many solution techniques are known. This type of reduction is not straightforward for triangular elements. In this paper we outline new mixed type methods which generalize finite differences in a manner suitable for use with triangular elements. Numerical examples illustrate the accuracy and efficiency of these new methods.

INTRODUCTION

Simulating flow in porous media requires the solution of elliptic or parabolic partial differential equations. In this paper, we discuss several variants of the mixed finite element method for solving elliptic equations of the form

$$-\nabla \cdot (K(\mathbf{x})\nabla p(\mathbf{x})) \equiv \nabla \cdot \mathbf{u}(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{1}$$

where the aquifer region is represented by a polygonal domain Ω in $\mathbb{R}^2$ partitioned into triangular elements, and K is a general tensor. We remark briefly on tetrahedral meshes defined on $\Omega \subset \mathbb{R}^3$ at the end of the paper. For simplicity, we assume the boundary condition

$$p(\mathbf{x}) = 0, \quad \mathbf{x} \in \partial\Omega, \tag{2}$$

but the methods apply to general boundary conditions. We solve (1) and (2) using the lowest-order Raviart-Thomas approximating spaces (RT_0) on triangular elements [8]. A C++ package which implements all of the methods presented here has been developed by the third author [6].

The mixed finite element method (MFEM) for solving elliptic equations was first described in [8]. The MFEM is especially useful in groundwater flow problems because mixed methods approximate the velocity $\mathbf{u}$ and the pressure p to the same order of accuracy. Furthermore, the approximate velocity calculated by the mixed method is locally mass conservative, satisfying $\nabla \cdot \mathbf{u} = f$ almost everywhere. In contrast, Galerkin finite element methods only conserve mass globally.

On rectangular elements, if K in (1) is a scalar or a diagonal matrix, one finds that numerical quadrature reduces the mixed method to a five-point finite difference method for pressure [10]. The new AWYM (Arbogast-Wheeler-Yotov method) extends these results to general coefficient tensors [2, 3, 4]. On triangular elements, however, the linear system arising from the MFEM is sparse but indefinite, making it expensive to solve. We have developed a cell-centered stencil method (CCSM) which reduces the MFEM on triangular elements to a ten-point finite difference type stencil. This provides a locally conservative finite difference method on triangles that is efficient and simple to implement. The CCSM is based on the AWYM with special quadrature rules and reduces to the usual finite difference method when applied to rectangular elements. The method is highly accurate on smooth triangulations. In more general cases the method loses accuracy, which can however be avoided by adding Lagrange multipliers on element faces where the discontinuities appear.

THE NUMERICAL MODELS

This section briefly defines some variants of the mixed method. See [1] for more details.

Let $(\cdot,\cdot)$ denote the $L^2(\Omega)$ inner product: for functions ϕ and ψ,

$$(\phi, \psi) = \int_\Omega \phi(\mathbf{x})\psi(\mathbf{x})\, d\mathbf{x} \quad \text{and} \quad L^2(\Omega) = \{\phi : ||\phi|| \equiv (\phi,\phi)^{1/2} < \infty\}.$$

Let $H(\text{div};\Omega) = \{\mathbf{v} \in (L^2(\Omega))^2 : \text{div}\,\mathbf{v} \in L^2(\Omega)\}$. Let $W_h \subset L^2(\Omega)$ and $\mathbf{V}_h \subset H(\text{div};\Omega)$ be the lowest-order Raviart-Thomas (RT_0) [8] approximating spaces defined on a triangulation of Ω into elements with maximum diameter $h > 0$. We note that W_h consists of functions that are constant on each element in the triangulation. Any function $\mathbf{v} \in \mathbf{V}_h$ is completely determined by the values of $\mathbf{v} \cdot \eta$ on the edges of the triangulation, where η is the unit normal to an edge. In the standard MFEM, we seek $\mathbf{U} \in V_h$ and $P \in W_h$ that satisfy

$$(K^{-1}\mathbf{U}, \mathbf{v}) - (P, \nabla \cdot \mathbf{v}) = 0, \quad \mathbf{v} \in V_h, \tag{3}$$

$$(\nabla \cdot \mathbf{U}, w) = (f, w), \quad w \in W_h. \tag{4}$$

It is well-known that for the RT_0 spaces on rectangular elements, numerical quadrature can be used in (3) to reduce the linear system of equations to a sparse, symmetric system in P only. Below we describe similar approaches for triangular elements.

The variants of the mixed method we investigate are based on a new formulation developed by Arbogast, Wheeler, and Yotov [2, 4]. In this approach, a mesh-dependent matrix function $\mathbf{s}_g$ and an additional variable $\mathbf{y}$ are introduced, with

$$\mathbf{s}_g \mathbf{y} = -\nabla p, \tag{5}$$

$$\mathbf{s}_g \mathbf{u} = \mathbf{s}_g K \mathbf{s}_g \mathbf{y}, \tag{6}$$

$$\nabla \cdot \mathbf{u} = q. \tag{7}$$

If $\mathbf{s}_g(x)$ is invertible, (5)–(7) is equivalent to (1). Let T denote any triangle in $\mathcal{T}_h$, and let T_{ref} denote a reference element, which we assume is the equilateral triangle with vertices at $(-1,0)$, $(1,0)$, and $(0,\sqrt{3})$. Let D_T denote the Jacobian matrix of the affine mapping between element T and T_{ref}, and let $J_T = |\det D_T|$. On T, define $\mathbf{s}_g$ by

$$\mathbf{s}_g|_T = J_T (D_T)^T D_T.$$

Note that $\mathbf{s}_g$ on each element is symmetric and positive definite. We approximate $\mathbf{y}$ by $\mathbf{Y} \in \mathbf{V}_h$, $\mathbf{u}$ by $\mathbf{U} \in \mathbf{V}_h$, and p by $P \in W_h$, where $\mathbf{Y}$, $\mathbf{U}$, and P satisfy

$$(\mathbf{s}_g \mathbf{Y}, \mathbf{v}) = (P, \nabla \cdot \mathbf{v}), \quad \mathbf{v} \in \mathbf{V}_h, \tag{8}$$

$$(\mathbf{s}_g \mathbf{U}, \mathbf{z}) = (\mathbf{s}_g K \mathbf{s}_g \mathbf{Y}, \mathbf{z}), \quad \mathbf{z} \in \mathbf{V}_h, \tag{9}$$

$$(\nabla \cdot \mathbf{U}, w) = (f, w), \quad w \in W_h. \tag{10}$$

In matrix form, (8)–(10) can be written as

$$\begin{bmatrix} S & 0 & -B \\ C & -S & 0 \\ 0 & B^T & 0 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{Y}} \\ \bar{\mathbf{U}} \\ \bar{P} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix}. \tag{11}$$

Reducing to an equation in $\bar{P}$ only, we find

$$A_{\text{awym}} \bar{P} \equiv (B^T S^{-1} C S^{-1} B) \bar{P} = R.$$

One advantage of the AWYM over the MFEM is that it does not require calculating K^{-1}, and thus is definable in cases where $K(x) = 0$. For time-dependent problems where $K = K(\cdot, t)$, the AWYM has the advantage over the MFEM that only S^{-1} is needed in the computation, not M^{-1}. Unlike M, S is not time-dependent and hence can be factored once at the beginning of the computation, whereas M must be factored at each time-step. In general, however, the computational expense of the AWYM and the MFEM are roughly equivalent.

For the two methods described above, assuming the RT_0 approximating spaces are used, the pressure and velocity are globally first order accurate. Superconvergence of pressure to second order is observed at the center of mass of each cell, and for rectangular meshes, superconvergence of velocities to second order is observed at certain Gauss points [2, 3].

We note that if S were a diagonal matrix, then A_{awym} would be sparse. Recently we developed the cell-centered stencil method (CCSM) for triangular meshes [1] which reduces S to a diagonal matrix by numerical integration of the left side of (8). In two space dimensions, the resulting matrix has at most ten nonzero entries on any row.

On any triangle T, let $\mathbf{v}_k$ denote the basis function of $\mathbf{V}_h$ associated with edge k, denoted by e_k, $k = 1, 2, 3$. Assume that e_k is mapped into edge k of T_{ref}, denoted by $\hat{e}_k$. Define the Piola transformation [9] on vectors by

$$\hat{\mathbf{v}}_k = J_T D_T \mathbf{v}_k, \quad k = 1, 2, 3. \tag{12}$$

It can be shown that

$$\int_T (\mathbf{s}_g \mathbf{v}_k) \cdot \mathbf{v}_l \, dx = \int_{T_{\mathrm{ref}}} \hat{\mathbf{v}}_k \cdot \hat{\mathbf{v}}_l \, dx, \tag{13}$$

for $k, l = 1, 2, 3$. Moreover, from (12), one can show that $\hat{\mathbf{v}}_k$ is a scalar multiple of the standard basis function for $\mathbf{V}_h$ corresponding to edge k of T_{ref}.

Using these facts, we define a quadrature rule $Q_T(g)$ on T_{ref} by

$$Q_T(g) = \frac{\sqrt{3}}{6}\left[g(-1,0) + g(1,0) + g(0,\sqrt{3}) + 3g\left(0, \frac{\sqrt{3}}{3}\right)\right]. \tag{14}$$

It has the two properties that $Q_T(g)$ is exact for polynomials of degree one, and $Q_T(\hat{\mathbf{v}}_k \cdot \hat{\mathbf{v}}_l) = 0$ for $k \neq l$. Thus, by (13),

$$\int_T (\mathbf{s}_g \mathbf{v}_k) \cdot \mathbf{v}_l \, dx \approx Q_T(\hat{\mathbf{v}}_k \cdot \hat{\mathbf{v}}_l) = \begin{cases} 0, & k \neq l, \\ \frac{\sqrt{3}}{6}(\mathrm{length}(e_k))^2, & k = l. \end{cases} \tag{15}$$

Let S_D denote the diagonal approximation to S determined by the quadrature rule given above. Then the CCSM reduces to finding $\bar{P}$ which satisfies

$$A_{\mathrm{ccsm}} \bar{P} = (B^T S_D^{-1} C S_D^{-1} B) \bar{P} = R. \tag{16}$$

Note that A_{ccsm} is a sparse approximation to A_{awym}.

The CCSM has been implemented for elliptic problems in two space dimensions [1] and compared to the results given by the AWYM and MFEM. Recall that the AWYM and MFEM give $\mathcal{O}(h^2)$ errors for pressures at element centers and $\mathcal{O}(h)$ global L^2 errors for velocities. For smooth triangulations, the CCSM gives the same orders of convergence, but for domains where a smooth triangulation is not feasible, these errors can be reduced by a full order of h. The reason is that the function $\mathbf{s}_g$ can be nonsmooth across element edges, depending on the triangulation. In these regions, the accuracy of the quadrature rule is reduced by one order (from h^2 to h), leading to a corresponding loss of accuracy in the solution. Thus it appears the CCSM may not be useful for totally unstructured, nonsmooth meshes, however, for meshes with some structure, we can modify the CCSM slightly to regain accuracy.

One way to regain accuracy in the solution is to smooth the mesh as it is refined. If this is for some reason not possible, one can use the Enhanced Cell-Centered Stencil Method (ECCSM) [1], which is a combination of the CCSM with the standard mixed-hybrid finite element method (MHFEM) [5]; in particular, pressure unknowns are added at element edges where $\mathbf{s}_g$ is not smooth. The result is a system of equations in velocity unknowns $\bar{\mathbf{Y}}$ and $\bar{\mathbf{U}}$ and pressure unknowns $\bar{P}$ and $\bar{\lambda}$, where $\bar{\lambda}$ represents pressure unknowns on element boundaries. This system can be reduced to an equation for $\bar{P}$ and $\bar{\lambda}$ only. Depending on the structure of the mesh, the ECCSM lies somewhere between the MHFEM and the CCSM in total number of unknowns. For domains which can be divided into a relatively small number of regular domains, where smooth triangulations can be used, the ECCSM is roughly equivalent in number of unknowns to the CCSM.

As mentioned above, the purpose of introducing the boundary pressures is to regain the accuracy in the numerical solution lost by the CCSM on nonsmooth meshes. We have implemented the ECCSM for problems where the CCSM lost accuracy and observed $\mathcal{O}(h^2)$ errors for pressure at cell centers and $\mathcal{O}(h)$ global L^2 errors for velocity [1], which are the same rates obtained by the MFEM, MHFEM, and AWYM. Moreover, numerical experiments indicate that a new postprocessing scheme developed by Keenan [7] can improve the convergence rate for the velocities almost to $\mathcal{O}(h^2)$.

NUMERICAL RESULTS

We created a large suite of test problems to examine the behavior of the numerical methods described above. We varied the shape of the domain, the coefficient tensor K, and the analytic solution. In each case, the boundary conditions and the forcing term f were constructed to match the prescribed solution. We report in detail on one typical case and then summarize the results from the full test suite.

Among the domains considered was the one shown in Figure 1. This figure illustrates the initial decomposition of the domain into elements. The domain is neither simply connected nor convex; moreover we chose to use both rectangles and triangles in subdividing it, to illustrate the flexibility of the C++ program.

In the convergence study, the domain was refined uniformly to generate progressively finer meshes. Each application of uniform refinement replaced each triangle or rectangle with 4 smaller but geometrically similar ones. The finest mesh had 2432 elements. Uniform refinement generates hierarchical meshes: each new mesh contains all the edges of the previous one. However, discontinuities in the geometry mapping across edges of the original coarse triangulation are not smoothed out by refinement.

In Tables 1 and 2, we give detailed results for a test problem using Dirichlet boundary conditions, a non-diagonal tensor, and a known analytic solution. We

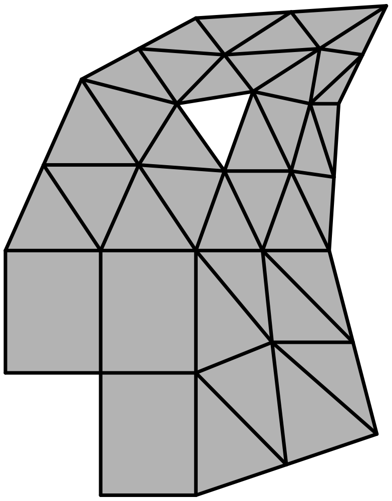

Figure 1: Example Domain

h	**ECCSM**	**CCSM**	**MHFEM**	**MFEM**
0.16	0.59	0.48	0.39	0.39
0.08	0.11	0.12	0.11	0.11
0.04	0.026	0.043	0.029	0.029
0.02	0.0062	0.019	0.0076	0.0076
Rate	h^2	$h^{1.4}$	h^2	h^2

Table 1: l^2 error in p for domain in Figure 1

report the l^2 norm of the error in the pressure p and the flux $K\nabla p$. (The l^2 norm is the discrete two-norm taken at the centers of elements.)

In the full test suite, the condition number of the linear system for most methods was $O(h^{-1})$, as estimated by the number of conjugate gradient iterations used. However, the ECCSM combined with uniform refinement produced better conditioned systems, with condition numbers around $O(h^{-0.9})$. Using a conjugate gradient solver with no preconditioning, the MFEM took much longer than the other three methods (approximately 50 times longer than the MHFEM on 2000 elements). On a typical smooth mesh problem the CCSM took approximately half as much CPU time as the MHFEM. The ECCSM was somewhat slower than the MHFEM on coarse meshes, since it solves for both pressures and Lagrange multipliers. By around four levels of mesh refinement it had caught up to the MHFEM, since it did not need Lagrange multipliers on every edge, and it should outperform it when additional refinement is used.

h	**ECCSM**	**CCSM**	**MHFEM**	**MFEM**	**Post-Proc**
0.16	6.4	9.3	6.0	6.0	6.4
0.08	3.5	5.9	3.1	3.1	1.8
0.04	1.6	3.7	1.5	1.5	0.60
0.02	0.80	2.5	0.77	0.77	0.20
Rate	h	$h^{0.6}$	h	h	$h^{1.6}$

Table 2: l^2 error in $K\nabla p$ for domain in Figure 1

The error in the pressure converged approximately like $O(h^2)$ for all methods except the CCSM in situations like the domain of Figure 1, where the geometry matrix changes discontinuously because of uniform refinement. Similarly the error in the flux converged like $O(h)$, except for CCSM with geometry discontinuities. Using smooth mesh refinement the CCSM achieved the same convergence orders as the other methods.

On rectangles one finds that the velocities are superconvergent at special points and can be post-processed to yield second order accurate vector approximations everywhere. Empirical evidence indicates that a new post processing scheme developed by the third author can recover close to second order accuracy for the velocities on triangular meshes as well [7]. The postprocessing method can be applied to any of the mixed method variants. The "Post-Proc" column in Table 2 shows the errors obtained with the ECCSM when post processing was used. The convergence rate for the post processed flux is generally between $h^{1.5}$ and $h^{1.9}$, depending in part on the smoothness of the mesh refinement process.

The C++ program can handle three dimensional elements such as bricks and tetrahedra. We observed numerically that the stencil approach breaks down on tetrahedral meshes. This appears to be due to the fact that regular tetrahedra do not fill space, whereas equilateral triangles do tile the plane. This means that the geometry matrix s_g is unavoidably discontinuous everywhere, no matter how much one attempts to smooth the tetrahedral mesh. Therefore, the MHFEM seems to be the best choice for tetrahedral meshes. The CCSM could however be used with prismatic elements on meshes constructed from the tensor product of a triangular mesh in two dimensions and a one dimensional collection of intervals.

CONCLUSIONS

The CCSM is an accurate and efficient method for smooth meshes of triangular elements, which appears to be about twice as fast as competing methods. On meshes of tetrahedral elements, however, the CCSM loses accuracy, so the MHFEM should be used instead.

ACKNOWLEDGMENTS

This research was supported in part by the Department of Energy, the State of Texas Governor's Energy Office, and project grants from the National Science Foundation. The third author was supported in part by an NSF Postdoctoral Fellowship.

REFERENCES

[1] Arbogast, T., Dawson, C., and Keenan, P. T. (1993) "Mixed Finite Element Methods as Finite Difference Methods for Solving Elliptic Equations on Triangular Elements", Technical Report #93–53, Department of Computational and Applied Mathematics, Rice University.

[2] Arbogast, T., Wheeler, M.F., and Yotov, I. (1994) "Logically rectangular mixed methods for groundwater flow and transport on general geometry", this volume.

[3] Arbogast, T., Wheeler, M.F., and Yotov, I. (in preparation) "Mixed finite elements for elliptic problems with tensor coefficients as finite differences".

[4] Arbogast, T., Wheeler, M.F., and Yotov, I. (in preparation) "Mixed finite element methods on general geometry".

[5] Brezzi, F. and Fortin, M. (1991) "Mixed and hybrid finite elements", Springer Series in Computational Mathematics, Vol. 15, Springer-Verlag, Berlin.

[6] Keenan, P. T. (1993) "Instructions for my C++ elliptic PDE solver programs using mixed methods on general geometry", Technical Report #93-56, Department of Computational and Applied Mathematics, Rice University.

[7] Keenan, P. T. (in preparation) "An efficient postprocessor for velocities from mixed methods on triangular and tetrahedral elements".

[8] Raviart, P. A., and Thomas, J. M. (1977) "A mixed finite element method for 2nd order elliptic problems", in Mathematical Aspects of the Finite Element Method, Lecture Notes in Math., Springer-Verlag, Berlin.

[9] Thomas, J. M. (1977) "Sur l'analyse numérique des méthodes d'éléments finis hybrides et mixtes", Thèse d'Etat, Université Pierre et Marie Curie.

[10] Russell, T.F., and Wheeler, M.F. (1983) "Finite element and finite difference methods for continuous flows in porous media", in Ewing, R.E. (ed.), The Mathematics of Reservoir Simulation, Frontiers in Applied Mathematics 1, Society for Industrial and Applied Mathematics, Philadelphia, Chapter II, pp. 35–106.