

Robust streamline tracing for the simulation of porous media flow on general triangular and quadrilateral grids

Sébastien F. Matringe^a, Ruben Juanes^{b,*}, Hamdi A. Tchelepi^a

^a Department of Petroleum Engineering, Stanford University, 65 Green Earth Sciences Building, Stanford, CA 94305, United States

^b Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Room 48-319, Cambridge, MA 02139, United States

Received 31 December 2005; received in revised form 12 July 2006; accepted 13 July 2006
Available online 30 August 2006

Abstract

Streamline methods for subsurface-flow simulation have received renewed attention as fast alternatives to traditional finite volume or finite element methods. Key aspects of streamline simulation are the accurate tracing of streamlines and the computation of travel time along individual streamlines. In this paper, we propose a new streamline tracing framework that enables the extension of streamline methods to unstructured grids composed of triangular or quadrilateral elements and populated with heterogeneous full-tensor coefficients. The proposed method is based on the mathematical framework of mixed finite element methods, which provides approximations of the velocity field that are especially suitable for streamline tracing. We identify and implement two classes of velocity spaces: the lowest-order Raviart–Thomas space (low-order tracing) and the Brezzi–Douglas–Marini space of order one (high-order tracing), both on triangles and quadrilaterals. We discuss the implementation of the streamline tracing method in detail, and we investigate the performance of the proposed tracing strategy by means of carefully designed test cases. We conclude that, for the same computational cost, high-order tracing is more accurate (smaller error in the time-of-flight) and robust (less sensitive to grid distortion) than low-order tracing.

© 2006 Elsevier Inc. All rights reserved.

Keywords: Streamlines; Time-of-flight; Porous media; Darcy flow; Groundwater transport; Petroleum reservoir simulation; Unstructured grids; Mixed finite elements

1. Introduction

The governing equations of multiphase flow in porous media can be expressed in terms of a pressure equation (of elliptic character) that imposes overall mass balance and a set of saturation equations (of hyperbolic character) that describes the evolution of the individual flowing phases [1–3]. Numerical simulation of the flow

* Corresponding author.

E-mail addresses: matringe@stanford.edu (S.F. Matringe), juanes@mit.edu (R. Juanes), tchelepi@stanford.edu (H.A. Tchelepi).

of oil, gas and water in petroleum reservoirs has traditionally been performed using finite difference methods. In the past decade, however, streamline methods have gained popularity as a fast alternative to finite differences [4–6].

In streamline simulation, the flow problem (pressure equation) and the transport problem (saturation equation) are solved sequentially using an operator-splitting technique. The solution of the flow problem provides the pressure field on the simulation grid. The velocity field is then interpolated from the pressure field and streamlines are traced. Finally, the transport problem is solved along the streamlines using a one-dimensional formulation of the saturation equation expressed in terms of the time-of-flight variable.

Although now accepted as a fast simulation tool, streamline simulation is still a relatively young technology and is only applicable to a limited range of problems. For instance, no commercial streamline simulator offers the ability to use unstructured simulation grids or full-tensor coefficients. The extension of the streamline method to advanced grids hinges on the ability to trace streamlines on these grids. An accurate velocity field description and streamline tracing are two of the key components that a streamline method must bring together. The complexity of the velocity field increases with grid distortion as well as heterogeneity and anisotropy in the coefficients describing the material properties. In this paper, we propose a streamline tracing framework that enables the extension of streamline methods to unstructured grids composed of triangular or quadrilateral elements and populated with heterogeneous full-tensor coefficients.

Most streamline simulators of groundwater transport and petroleum reservoirs employ Pollock's method [7] for streamline tracing. Pollock's method is a semi-analytical method that recovers the exit point of a streamline and the time-of-flight in an element of the simulation grid, by assuming that each component of the velocity field varies linearly within the element. The method is applicable, strictly speaking, only to rectangular elements and diagonal permeability tensors. Extensions of Pollock's method to irregular grids have been presented in the context of nodal-based finite element methods [8] and flux-continuous control volume methods [9,10,12,11]. At the core of these extensions is a proper mapping of the velocity field from the element in physical space to a reference element, known as the Piola transform [13,14].

Our approach is based on the mixed finite element method, in which pressure and Darcy velocity are approximated simultaneously. Theoretical and computational advantages of using mixed finite elements are discussed at length in [15–19]. These investigations argue rather convincingly that the mixed finite element method produces physical, high-quality streamlines even on coarse grids and highly heterogeneous media. Their work is limited, however, to the lowest-order mixed finite element approximation. Recently, it has been shown that low-order accurate streamline tracing may lead to large errors in both streamline location and time-of-flight [20,21].

In a recent paper [22], we present a unified formulation for the development of high-order accurate streamline tracing algorithms on unstructured triangular and quadrilateral grids. The main result of that investigation is the identification of velocity spaces that are suitable for streamline tracing, by imposing that the divergence-free part of the velocity must induce a stream function. Several classes of velocity spaces satisfying this requirement are identified from the theory of mixed finite element methods. Not surprisingly, the most widely used tracing algorithm (Pollock's method) emanates in fact from the lowest-order admissible velocity approximation. Therefore, we provide a sound theoretical justification for the low-order algorithms currently in use, and we show how to achieve higher-order accuracy both in the streamline tracing and the travel time computation. Here, we concentrate on a small subset of the mixed velocity–pressure approximations identified in [22], for which the pressure is constant over each element. This restricts the choice of velocity spaces to the following two: the lowest-order Raviart–Thomas space (RT_0) and the Brezzi–Douglas–Marini space of order one (BDM_1), both on triangles and quadrilaterals.

In Section 2 we pose the model mathematical problem of Darcy flow in porous media – the solution of which defines the velocity field to be used for streamline tracing – and discuss its numerical approximation by mixed finite elements. The specific functional forms of the velocity spaces employed are reviewed in Section 3. The overall strategy for streamline tracing and time-of-flight computation is described in Section 4. In Section 5 we illustrate the performance of the proposed streamline tracing method with representative test cases of increasing complexity. In particular, we compare the behavior of low-order and high-order tracing. Finally, in Section 6, we summarize the main conclusions and discuss the ongoing and future work.

2. Mathematical and numerical formulation

2.1. Continuum problem

In groundwater transport and petroleum reservoir engineering applications, streamlines are computed as integral paths of a velocity field governed by Darcy porous-media flow. The flow problem is composed of Darcy's law, Eq. (1), and the mass conservation condition expressed in Eq. (2):

$$\mathbf{k}^{-1}\mathbf{u} + \nabla p = \mathbf{0} \quad \text{in } \Omega, \quad (1)$$

$$\nabla \cdot \mathbf{u} = f \quad \text{in } \Omega. \quad (2)$$

In Eqs. (1) and (2), \mathbf{u} is the Darcy velocity, p is the pressure, \mathbf{k} is a possibly discontinuous full-tensor permeability coefficient (divided by the fluid viscosity), f is a source term and Ω is the simulation domain. In the context of multiphase flow problems with gravity, p is the flow potential and \mathbf{k} is the permeability multiplied by the total mobility. The flow problem is supplemented with the following boundary conditions:

$$p = \bar{p} \quad \text{on } \Gamma_p, \quad (3)$$

$$\mathbf{u} \cdot \mathbf{n} = \bar{u} \quad \text{on } \Gamma_u, \quad (4)$$

where $\Gamma_p \cap \Gamma_u = \emptyset$, $\Gamma_p \cup \Gamma_u = \partial\Omega$, and \mathbf{n} is the outward unit normal to the boundary of Ω .

Under certain regularity conditions on the source term f and the boundary condition \bar{p} , the continuum problem Eqs. (1)–(4) admits a unique solution [14]. The pressure solution belongs to the Sobolev space of square integrable functions in Ω :

$$L^2(\Omega) = \left\{ p : \int_{\Omega} |p|^2 d\Omega < +\infty \right\}. \quad (5)$$

The regularity of the velocity solution is more restrictive, as it needs to admit well-defined normal traces on $\partial\Omega$ [14]. Physically, the integral of the normal trace of the velocity field along a boundary is precisely the volumetric flux across this boundary. The space with the proper regularity for the velocity is:

$$H(\text{div}, \Omega) = \left\{ \mathbf{u} : \mathbf{u} \in (L^2(\Omega))^2, \nabla \cdot \mathbf{u} \in L^2(\Omega) \right\}. \quad (6)$$

In order to satisfy the flux boundary conditions, the velocity solution must belong to the space

$$\mathcal{S} = \{ \mathbf{u} : \mathbf{u} \in H(\text{div}, \Omega), \mathbf{u} \cdot \mathbf{n} = \bar{u} \quad \text{on } \Gamma_u \}, \quad (7)$$

that is, the restriction of $H(\text{div}, \Omega)$ to the functions that satisfy the Neumann boundary condition (4). We remark that the flux boundary condition appears in the mixed formulation as an *essential* boundary condition.

Having defined the functional spaces in which the pressure–velocity solution is sought, we define now two spaces of test functions: a space of scalar functions $q \in L^2(\Omega)$, and a space of vector functions $\mathbf{v} \in \mathcal{V}$ with:

$$\mathcal{V} = \{ \mathbf{v} : \mathbf{v} \in H(\text{div}, \Omega), \mathbf{v} \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_u \}. \quad (8)$$

The space \mathcal{V} is the restriction of $H(\text{div}, \Omega)$ to the functions satisfying the homogeneous counterpart of the Neumann boundary condition (4).

Assuming sufficient regularity of f and \bar{p} , the mixed variational form of the problem is obtained by multiplying Eq. (1) by the velocity test function \mathbf{v} , and Eq. (2) by the pressure test function q , and integrating both equations over Ω . Integration by parts of the first equation leads to the mixed variational form of the problem, which can be stated as follows:

Find $(\mathbf{u}, p) \in \mathcal{S} \times L^2(\Omega)$ such that

$$\int_{\Omega} \mathbf{v} \cdot \mathbf{k}^{-1} \mathbf{u} d\Omega - \int_{\Omega} \nabla \cdot \mathbf{v} p d\Omega = - \int_{\Gamma_p} \mathbf{v} \cdot \mathbf{n} \bar{p} d\Gamma \quad \forall \mathbf{v} \in \mathcal{V}, \quad (9)$$

$$\int_{\Omega} q \nabla \cdot \mathbf{u} d\Omega = \int_{\Omega} q f d\Omega \quad \forall q \in L^2(\Omega). \quad (10)$$

2.2. Discretization

Mixed finite element methods approximate the pressure and velocity fields simultaneously, and they are based on the mixed variational form (9), (10). Let \mathcal{T}_h be a partition of Ω into triangular or quadrilateral elements K_j , such that $\Omega = \bigcup_{j=1}^{n_{\text{elem}}} K_j$. Let \mathcal{Q}_h , \mathcal{S}_h and \mathcal{V}_h be finite-dimensional subspaces of $L^2(\Omega)$, \mathcal{S} and \mathcal{V} , respectively. The discrete mixed finite element approximation of the continuum problem (9) and (10) reads: Find $(\mathbf{u}_h, p_h) \in \mathcal{S}_h \times \mathcal{Q}_h$ such that

$$\int_{\Omega} \mathbf{v}_h \cdot \mathbf{k}^{-1} \mathbf{u}_h \, d\Omega - \int_{\Omega} \nabla \cdot \mathbf{v}_h p_h \, d\Omega = - \int_{\Gamma_p} \mathbf{v}_h \cdot \mathbf{n} \bar{p}_h \, d\Gamma \quad \forall \mathbf{v}_h \in \mathcal{V}_h, \tag{11}$$

$$\int_{\Omega} q_h \nabla \cdot \mathbf{u}_h \, d\Omega = \int_{\Omega} q_h f \, d\Omega \quad \forall q_h \in \mathcal{Q}_h. \tag{12}$$

The pressure and velocity spaces cannot be chosen independently of each other. To obtain a convergent approximation, they must satisfy two conditions [14,23]: a standard coercivity condition, and the discrete inf-sup condition [24,25]. As an example, we show in Fig. 1 a schematic of the locations of the pressure and flux unknowns on the triangular and quadrilateral reference elements for the lowest-order approximation used in this paper.

As in any other finite element procedure, the pressure and velocity fields are interpolated from the pressure and flux unknowns using shape functions with local support:

$$\mathbf{u}_h = \sum_{i=1}^{n_{\text{edge}}} U_i \mathbf{N}_i^u, \tag{13}$$

$$p_h = \sum_{j=1}^{n_{\text{elem}}} P_j N_j^p, \tag{14}$$

where N_i^u , N_j^p are the velocity and pressure shape functions and U_i , P_j the corresponding flux and pressure unknowns. The above discretization yields an indefinite linear system of the form

$$\begin{pmatrix} \mathbf{A} & -\mathbf{B}' \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U} \\ \mathbf{P} \end{pmatrix} = \begin{pmatrix} \mathbf{R}^U \\ \mathbf{R}^P \end{pmatrix}, \tag{15}$$

where \mathbf{U} and \mathbf{P} are the vectors of flux and pressure unknowns, \mathbf{R}^U and \mathbf{R}^P are the Darcy equation and mass balance equation residual vectors, \mathbf{A} is a square matrix of size $n_{\text{edge}} \times n_{\text{edge}}$ and \mathbf{B} is a matrix of size $n_{\text{elem}} \times n_{\text{edge}}$. Because of its indefinite character, an augmented Lagrangian method known as Uzawa’s algorithm [26] is used to solve the system (15). Alternatively, one may hybridize the system and solve a symmetric, positive definite system for the traces of the pressure at the element edges (see, e.g. [14]).

The mixed finite element method just described is mass conservative at the element level: the mass balance condition is enforced on each control volume on which the permeability coefficient is defined. Therefore, in the absence of sources and sinks, this discretization yields a divergence-free velocity field.

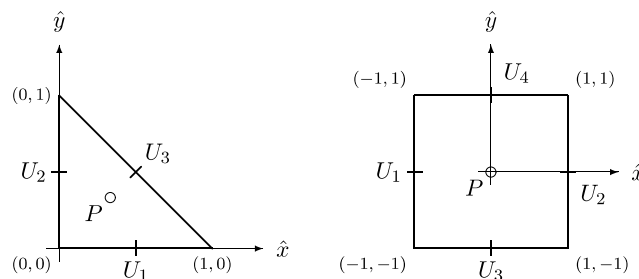


Fig. 1. Location of the pressure (P) and flux (U_1 – U_4) unknowns on the reference triangular and quadrilateral elements.

3. Velocity spaces

3.1. From global velocity field to local velocity spaces

In finite element methods, the global velocity spaces \mathcal{S}_h and \mathcal{V}_h are constructed using velocity spaces defined locally, on a reference element \widehat{K} . The reference element \widehat{K} is linked to any given element K in the physical space by a mapping, or change of coordinates $\boldsymbol{\varphi}$ defined by:

$$\begin{aligned} \boldsymbol{\varphi} : \mathbb{R}^2 &\rightarrow \mathbb{R}^2, \\ \hat{\mathbf{x}} \in \widehat{K} &\mapsto \mathbf{x} = \boldsymbol{\varphi}(\hat{\mathbf{x}}) \in K. \end{aligned} \tag{16}$$

The above mapping must be smooth and invertible for all elements. This ensures that, for any $\hat{\mathbf{x}} \in \widehat{K}$, the Jacobian matrix $\mathbf{D}(\hat{\mathbf{x}}) = \partial\boldsymbol{\varphi}/\partial\hat{\mathbf{x}}$ of the transformation is invertible, and that its determinant $J(\hat{\mathbf{x}}) = \det \mathbf{D}(\hat{\mathbf{x}})$ is bounded away from zero. The map $\boldsymbol{\varphi}$ from reference to physical space is shown in Figs. 2 and 3 for triangular and quadrilateral elements, respectively. It is given by the isoparametric mapping:

$$\mathbf{x} = \boldsymbol{\varphi}(\hat{\mathbf{x}}) = \sum_{a=1}^{n_{\text{node}}} N_a(\hat{\mathbf{x}}) \mathbf{x}_a, \tag{17}$$

where \mathbf{x}_a are the nodal coordinates (in physical space), and N_a are the usual finite element hat functions (in reference space):

$$N_1(\hat{x}, \hat{y}) = 1 - \hat{x} - \hat{y}, \quad N_2(\hat{x}, \hat{y}) = \hat{x}, \quad N_3(\hat{x}, \hat{y}) = \hat{y} \tag{18}$$

for triangular elements, and

$$\begin{aligned} N_1(\hat{x}, \hat{y}) &= \frac{1}{4}(1 - \hat{x})(1 - \hat{y}), & N_2(\hat{x}, \hat{y}) &= \frac{1}{4}(1 + \hat{x})(1 - \hat{y}), \\ N_3(\hat{x}, \hat{y}) &= \frac{1}{4}(1 + \hat{x})(1 + \hat{y}), & N_4(\hat{x}, \hat{y}) &= \frac{1}{4}(1 - \hat{x})(1 + \hat{y}) \end{aligned} \tag{19}$$

for quadrilateral elements.

Using $\mathbf{D}(\hat{\mathbf{x}})$ and $J(\hat{\mathbf{x}})$, we define the *Piola transform* used to map the velocity field from reference to physical space [14,13]:

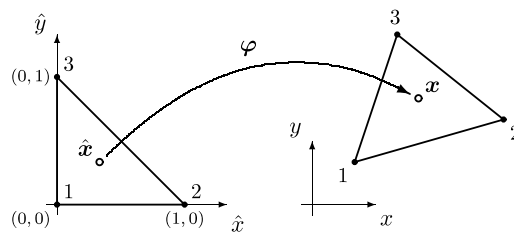


Fig. 2. Mapping of the reference element onto physical space for triangular elements.

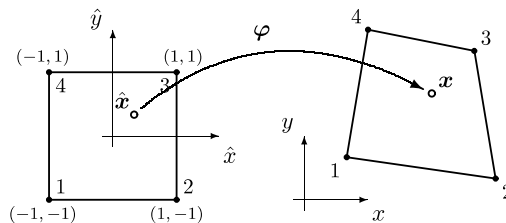


Fig. 3. Mapping of the reference element onto physical space for quadrilateral elements.

$$\mathbf{v}(\mathbf{x}) = \mathcal{P}(\hat{\mathbf{v}})(\mathbf{x}) = \frac{1}{J(\hat{\mathbf{x}})} \mathbf{D}(\hat{\mathbf{x}}) \hat{\mathbf{v}}(\hat{\mathbf{x}}), \quad \text{with } \mathbf{x} = \boldsymbol{\varphi}(\hat{\mathbf{x}}). \quad (20)$$

The key property of the Piola transform is that it preserves the normal trace of vector fields. In our case, this means that the fluxes through element edges in the reference and physical space are equal. Using the Piola transform, we can translate the necessary conditions on the global velocity field into conditions at the local level, on the reference element. Namely, two conditions must be verified:

- (1) The local velocity field on element \hat{K} must belong to $H(\text{div}, \hat{K})$.
- (2) The normal trace of the velocity must be continuous across element edges.

A large class of velocity spaces is in principle valid [14]. In a separate publication [22], we concluded that the family of Raviart–Thomas spaces [27] and Brezzi–Douglas–Marini spaces [28] are particularly well suited for streamline tracing (see Section 3.2 below). Here, we restrict our attention to spaces that satisfy the inf-sup condition in combination with a discretization of the pressure field that is *constant over each element* and, therefore, piecewise constant globally. From the set of velocity spaces analyzed in [22], only two spaces are compatible with the chosen pressure discretization: the lowest-order Raviart–Thomas space (RT_0) and the Brezzi–Douglas–Marini space of order one (BDM_1).

3.2. Existence of a stream function

The key property of the stream function is that it is constant along a streamline. Existence of a stream function can be exploited to obtain analytical streamline paths (see Section 4.2). A necessary condition for the existence of a stream function is that the velocity field be divergence-free. This is the case in groundwater transport and petroleum reservoir applications, because the source term is equal to zero everywhere except at the injection and production wells. In typical subsurface-flow applications, streamlines are only traced outside the blocks containing the wells [5], where the velocity field is divergence-free. This limitation is inconsequential from a practical viewpoint because the velocity in the near-well region is high and, therefore, the time-of-flight along the well block is insignificant compared to the travel time along the entire streamline.

In [22] we showed that a large class of velocity spaces emanating from the theory of mixed finite element methods induce a stream function. This means that a function $\Psi(x, y)$ exists such that the x - and y -components of the velocity are given by:

$$u_x = \frac{\partial \Psi}{\partial y}, \quad u_y = -\frac{\partial \Psi}{\partial x}. \quad (21)$$

In [22], the explicit functional form of the stream function was derived for the divergence-free part of the Raviart–Thomas and the Brezzi–Douglas–Marini spaces of *any* order, for both triangular and quadrilateral elements. In the following sections we give the specific functional forms of the shape functions, the velocity field and the stream function for the RT_0 and BDM_1 spaces on triangles and quadrilaterals.

3.3. The lowest-order Raviart–Thomas space: $\text{RT}_0(\hat{K})$

3.3.1. Definition

The simplest polynomial subspace conforming in $H(\text{div}, \hat{K})$ is the lowest-order Raviart–Thomas space [27], $\text{RT}_0(\hat{K})$. Velocity fields in $\text{RT}_0(\hat{K})$ are described by a constant normal trace along element edges, as sketched in Fig. 4. Knowledge of the fluxes across each of the edges of an element is sufficient to fully describe the RT_0 velocity field. Thus, three degrees of freedom are needed to fully characterize RT_0 on triangles and four on quadrilaterals.

Away from the well blocks, the velocity is divergence-free. Therefore, we use the restriction of $\text{RT}_0(\hat{K})$ to divergence-free functions:

$$\text{RT}_0^0(\hat{K}) := \{\hat{\mathbf{u}} : \hat{\mathbf{u}} \in \text{RT}_0(\hat{K}), \nabla \cdot \hat{\mathbf{u}} = 0\}. \quad (22)$$

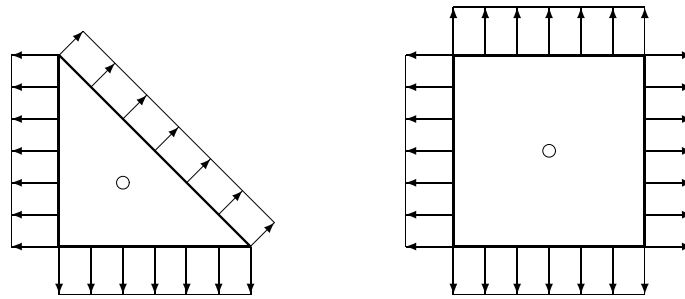


Fig. 4. RT_0 velocity fields are described by a constant normal trace across element edges.

Because of the added divergence-free constraint, the dimension of RT_0^0 is reduced by one:

$$\dim(RT_0^0(\hat{K})) = 3 \times 1 - 1 = 2 \quad \text{for triangular elements,} \tag{23}$$

$$\dim(RT_0^0(\hat{K})) = 4 \times 1 - 1 = 3 \quad \text{for quadrilateral elements.} \tag{24}$$

To describe an RT_0^0 velocity field, one shape function per edge is needed. The velocity shape function of an edge e is a vector field that has a unit outward flux across the edge e and a normal trace identically equal to zero on all other edges. In Fig. 5 we plot the RT_0^0 shape function for one of the edges on triangular and quadrilateral reference elements.

3.3.2. RT_0 on triangular elements

On triangular elements, the RT_0 velocity field is given by:

$$\hat{\mathbf{u}}(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b\hat{x} \\ a_2 + b\hat{y} \end{pmatrix}, \quad (a_1, a_2, b) \in \mathbb{R}^3. \tag{25}$$

This velocity field is defined through three shape functions:

$$\hat{N}_1^{\hat{u}} = \begin{pmatrix} \hat{x} \\ -1 + \hat{y} \end{pmatrix}, \quad \hat{N}_2^{\hat{u}} = \begin{pmatrix} -1 + \hat{x} \\ \hat{y} \end{pmatrix}, \quad \hat{N}_3^{\hat{u}} = \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix}. \tag{26}$$

The numbering of the shape functions in Eq. (26) corresponds to the local edge numbering on the reference element, shown in Fig. 1. The unknowns associated with these shape functions are the integral fluxes across the corresponding element edges.

The velocity field belongs to RT_0^0 if it is divergence-free, that is,

$$\nabla \cdot \hat{\mathbf{u}} = 2b = 0 \Rightarrow b = 0. \tag{27}$$

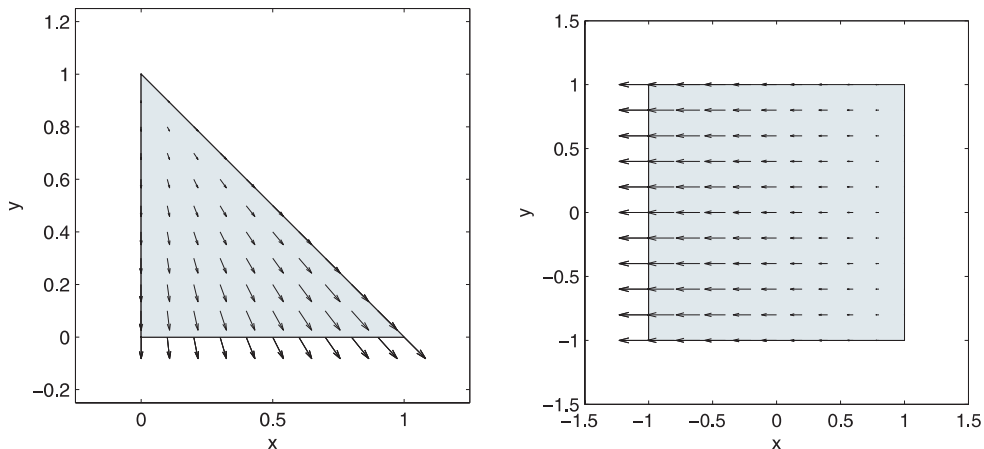


Fig. 5. RT_0^0 shape function for one of the edges on the triangular and quadrilateral reference elements.

Therefore, the divergence-free RT_0^0 velocity field is constant over the element, and takes the form:

$$\hat{\mathbf{u}}_0(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, \quad (a_1, a_2) \in \mathbb{R}^2. \tag{28}$$

The stream function associated with this velocity field is:

$$\Psi(\hat{x}, \hat{y}) = -a_2\hat{x} + a_1\hat{y}, \tag{29}$$

which means that RT_0^0 streamlines are straight lines inside a triangular element.

3.3.3. RT_0 on quadrilateral elements

On quadrilateral elements, each component of the RT_0 velocity field varies linearly with respect to its own coordinate:

$$\hat{\mathbf{u}}(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} \\ a_2 + b_2\hat{y} \end{pmatrix}, \quad (a_1, a_2, b_1, b_2) \in \mathbb{R}^4. \tag{30}$$

We note that the velocity field given in Eq. (30) is equal to the velocity interpolation used in Pollock’s method [7]. The velocity field is recovered using the following four RT_0 shape functions on the reference quadrilateral:

$$\begin{aligned} \hat{N}_1^{\hat{u}} &= -\frac{1}{4}(1 - \hat{x}) \begin{pmatrix} 1 \\ 0 \end{pmatrix}, & \hat{N}_2^{\hat{u}} &= \frac{1}{4}(1 + \hat{x}) \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \\ \hat{N}_3^{\hat{u}} &= -\frac{1}{4}(1 - \hat{y}) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, & \hat{N}_4^{\hat{u}} &= \frac{1}{4}(1 + \hat{y}) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \end{aligned} \tag{31}$$

where the indices 1–4 refer to the edge numbers shown in Fig. 1. The degrees of freedom associated with the shape functions above are precisely the integral fluxes across the corresponding edges.

For the velocity to be divergence-free, the four degrees of freedom (a_1, a_2, b_1, b_2) must satisfy the following relation:

$$\nabla \cdot \hat{\mathbf{u}} = b_1 + b_2 = 0 \Rightarrow b_2 = -b_1, \tag{32}$$

so a velocity field belonging to RT_0^0 has the form:

$$\hat{\mathbf{u}}_0(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} \\ a_2 - b_1\hat{y} \end{pmatrix}, \quad (a_1, a_2, b_1) \in \mathbb{R}^3, \tag{33}$$

and the corresponding stream function:

$$\Psi(\hat{x}, \hat{y}) = -a_2\hat{x} + a_1\hat{y} + b_1\hat{x}\hat{y}. \tag{34}$$

3.4. The Brezzi–Douglas–Marini space of order one: $BDM_1(\hat{K})$

3.4.1. Definition

The other velocity space compatible with a piecewise constant description of the pressure field and providing a conforming approximation of $H(\text{div}, \Omega)$ is the Brezzi–Douglas–Marini space of order one [28], $BDM_1(\hat{K})$. This space improves the description of the velocity field by allowing a linear variation of the normal trace of the velocity along element edges, as depicted in Fig. 6. One extra degree of freedom per edge is available for the description of the velocity, which brings the dimensionality of $BDM_1(\hat{K})$ to 6 for triangular elements, and 8 for quadrilaterals.

Once again, we define $BDM_1^0(\hat{K})$, the restriction of $BDM_1(\hat{K})$ to divergence-free functions, as:

$$BDM_1^0(\hat{K}) := \{\hat{\mathbf{u}} : \hat{\mathbf{u}} \in BDM_1(\hat{K}), \nabla \cdot \hat{\mathbf{u}} = 0\}. \tag{35}$$

The added divergence-free condition reduces the dimensionality of BDM_1^0 :

$$\dim(BDM_1^0(\hat{K})) = 3 \times 2 - 1 = 5 \quad \text{for triangular elements,} \tag{36}$$

$$\dim(BDM_1^0(\hat{K})) = 4 \times 2 - 1 = 7 \quad \text{for quadrilateral elements.} \tag{37}$$

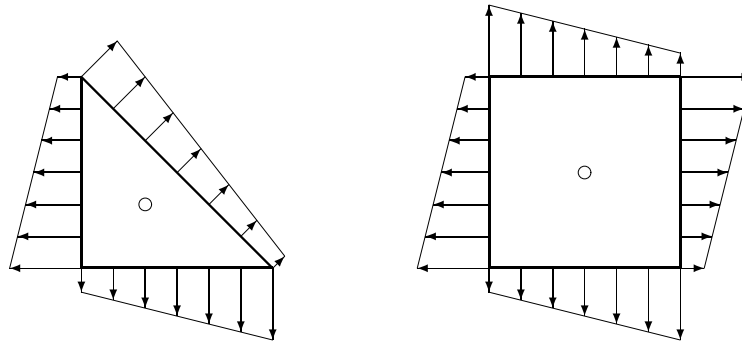


Fig. 6. BDM₁ velocity fields are described by a linearly varying normal trace along element edges.

In addition to the RT₀⁰ shape functions, associated with the integral fluxes through the element edges, *recirculation* shape functions are used to describe the linear variation of the velocity profile across edges. The recirculation shape function associated with an edge *e* has a zero net flux through any of the element edges, but has a linearly varying normal trace along edge *e*. As an illustration, we plot in Fig. 7 the recirculation shape functions associated with one of the edges of the triangular and quadrilateral reference elements.

3.4.2. BDM₁ on triangular elements

On triangular elements, the three recirculation shape functions are:

$$\hat{\mathbf{M}}_1^{\hat{u}} = \begin{pmatrix} \hat{x} \\ 1 - 2\hat{x} - \hat{y} \end{pmatrix}, \quad \hat{\mathbf{M}}_2^{\hat{u}} = \begin{pmatrix} -1 + \hat{x} + 2\hat{y} \\ -\hat{y} \end{pmatrix}, \quad \hat{\mathbf{M}}_3^{\hat{u}} = \frac{1}{\sqrt{2}} \begin{pmatrix} -\hat{x} \\ \hat{y} \end{pmatrix}. \tag{38}$$

They enrich the RT₀ velocity field to obtain the following form of the BDM₁ velocity:

$$\hat{\mathbf{u}}(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} + c_1\hat{y} \\ a_2 + b_2\hat{x} + c_2\hat{y} \end{pmatrix}, \quad (a_1, a_2, b_1, b_2, c_1, c_2) \in \mathbb{R}^6. \tag{39}$$

The additional divergence-free condition leads to the restriction:

$$\nabla \cdot \hat{\mathbf{u}} = b_1 + c_2 = 0 \Rightarrow c_2 = -b_1. \tag{40}$$

Using this condition, the BDM₁⁰ velocity field takes the form:

$$\hat{\mathbf{u}}_0(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} + c_1\hat{y} \\ a_2 + b_2\hat{x} - b_1\hat{y} \end{pmatrix}, \quad (a_1, a_2, b_1, b_2, c_1) \in \mathbb{R}^5, \tag{41}$$

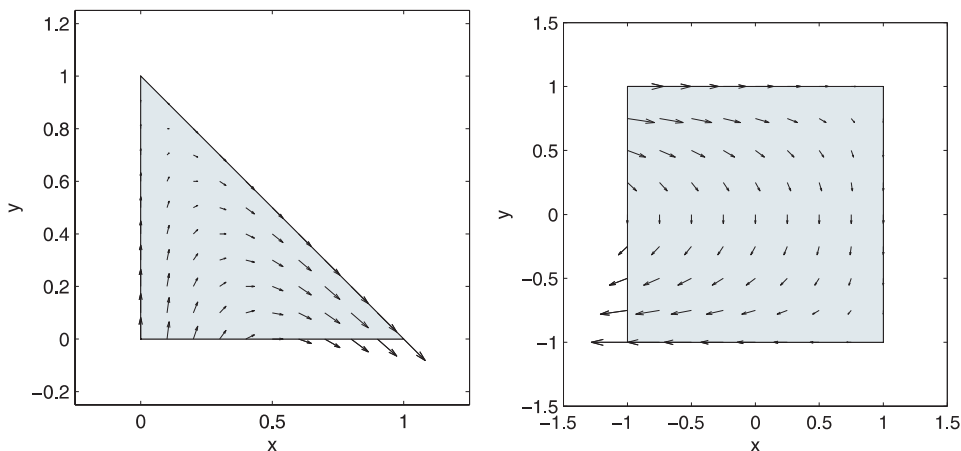


Fig. 7. BDM₁⁰ recirculation shape function for one of the edges of the triangular and quadrilateral reference elements.

and the associated stream function is:

$$\Psi(\hat{x}, \hat{y}) = -a_2\hat{x} + a_1\hat{y} + b_1\hat{x}\hat{y} - \frac{b_2}{2}\hat{x}^2 + \frac{c_1}{2}\hat{y}^2. \quad (42)$$

3.4.3. BDM₁ on quadrilateral elements

The recirculation shape functions on the reference quadrilateral element are:

$$\begin{aligned} \hat{\mathbf{M}}_1^{\hat{u}} &= (1 - \hat{x})(2\hat{y}) \begin{pmatrix} 1 \\ 0 \end{pmatrix} - (1 - \hat{y}^2) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\ \hat{\mathbf{M}}_2^{\hat{u}} &= (1 + \hat{x})(2\hat{y}) \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (1 - \hat{y}^2) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\ \hat{\mathbf{M}}_3^{\hat{u}} &= (1 - \hat{x}^2) \begin{pmatrix} 1 \\ 0 \end{pmatrix} - (2\hat{x})(1 - \hat{y}) \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\ \hat{\mathbf{M}}_4^{\hat{u}} &= -(1 - \hat{x}^2) \begin{pmatrix} 1 \\ 0 \end{pmatrix} - (2\hat{x})(1 + \hat{y}) \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \end{aligned} \quad (43)$$

The BDM₁ velocity field is given by:

$$\hat{\mathbf{u}}(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} + c_1\hat{y} - r\hat{x}^2 - 2s\hat{x}\hat{y} \\ a_2 + b_2\hat{x} + c_2\hat{y} + 2r\hat{x}\hat{y} + s\hat{y}^2 \end{pmatrix}, \quad (a_1, a_2, b_1, b_2, c_1, c_2, r, s) \in \mathbb{R}^8. \quad (44)$$

The divergence-free condition,

$$\nabla \cdot \hat{\mathbf{u}} = b_1 + c_2 = 0 \Rightarrow c_2 = -b_1, \quad (45)$$

yields the following form for the velocity field in BDM₁⁰:

$$\hat{\mathbf{u}}_0(\hat{\mathbf{x}}) = \begin{pmatrix} a_1 + b_1\hat{x} + c_1\hat{y} - r\hat{x}^2 - 2s\hat{x}\hat{y} \\ a_2 + b_2\hat{x} - b_1\hat{y} + 2r\hat{x}\hat{y} + s\hat{y}^2 \end{pmatrix}, \quad (a_1, a_2, b_1, b_2, c_1, r, s) \in \mathbb{R}^7, \quad (46)$$

and the corresponding stream function:

$$\Psi(\hat{x}, \hat{y}) = -a_2\hat{x} + a_1\hat{y} + b_1\hat{x}\hat{y} - \frac{b_2}{2}\hat{x}^2 + \frac{c_1}{2}\hat{y}^2 - r\hat{x}^2\hat{y} - s\hat{x}\hat{y}^2. \quad (47)$$

4. Streamline tracing approach

4.1. General strategy

Our streamline tracing algorithm is based on a particle-tracking concept. A streamline is traced by following the movement of a fluid particle in time. Since the mixed finite element method provides a velocity field defined elementwise and not globally, it is therefore natural to trace streamlines by segments – each segment corresponding to an underlying element of the simulation grid. The streamline tracing procedure may be summarized as follows:

- Start at a *launching* point in Ω . The launching point (x_0, y_0, t_0) defines the location (in space and time) of the fluid particle that will be followed to trace the streamline.
- Determine in what element the launching point lies. The tracing will start within this element.
- Trace the streamline downstream towards a sink. The fluid particle is followed forward in time. In each element crossed by the streamline:
 - Trace the streamline downstream from the point of entry in the element.
 - Store the exit point and corresponding time-of-flight.
 - Move on to the next downstream element until a sink (element with a production well) or an outflow boundary is reached.

- Trace the streamline upstream, towards a source. The fluid particle is followed backwards in time. In each element crossed:
 - Trace the streamline upstream from the entry point in the element.
 - Store the exit point and time-of-flight.
 - Move on to the next upstream element until a source (element with an injection well) or an inflow boundary is reached.

4.2. Tracing streamlines in the reference space

Most streamline tracing methods share the particle tracking approach described above. They usually differ in the procedure chosen to integrate the streamline within each element. For example, for low-order accurate methods on general triangular or rectangular elements, an analytical integration of the streamline path is possible [7,8]. When rectangular elements are distorted into general quadrilaterals, the Jacobian of the isoparametric mapping is not constant over the element and the analytical integration of the streamline is not possible in physical space. Moreover, in the case of higher-order discretizations, the enrichment of the velocity fields prevents the decoupling between the x - and y -components of the velocity used by Pollock to obtain an analytical integration [7].

Because we are interested in tracing streamlines on general unstructured grids, we propose to perform the tracing in the reference space and then map the solution to the physical space. Two main reasons drive this choice. First, the velocity field and stream function are known analytically on the reference element. Second, working on the reference element permits a more elegant and efficient implementation of the tracing algorithm.

To trace streamlines in the reference element, we recall that two mappings need to be used: the isoparametric mapping in Eq. (17) for the coordinates, and the Piola transform in Eq. (20) for the velocity. The general procedure to trace a streamline within an element starts with the mapping of the entry point from the physical space to the reference space. Then, the streamline is integrated and the exit point found in the reference space. The exit point is finally mapped back to the physical space and stored.

Two approaches are possible to obtain the exit point of a streamline in an element: an algebraic formulation using the stream function, and a numerical integration using a Runge–Kutta-type method.

4.2.1. Stream function approach

In [22], we showed that a very large class of mixed finite element velocity fields induce a stream function. In particular, this is true for the choice of velocity spaces described in Sections 3.3 and 3.4. Using the fundamental property that the stream function is constant along a streamline, we can write an equation for the path of the streamline that passes through a point (\hat{x}_0, \hat{y}_0) :

$$\Psi(\hat{x}, \hat{y}) = \Psi(\hat{x}_0, \hat{y}_0), \quad \forall (\hat{x}, \hat{y}) \in \text{streamline}. \quad (48)$$

This approach simplifies the tracing from the solution of an ordinary differential equation to that of an algebraic equation. Since the stream function has an analytical expression, the streamline path is known analytically and an efficient Newton method can be used to solve the algebraic equation for the exit point up to machine precision.

Depending on the functional form of the stream function, Eq. (48) may lead to multiple solutions on the boundary of a given element. This situation cannot be encountered when using an RT_0^0 discretization, but it is not uncommon for BDM_1^0 velocity fields. This situation occurs when a streamline enters and exit an element several times. To find the real exit point among the solutions of the streamline equation, the time-of-flight is computed (see Section 4.3) for each potential exit location. The correct exit point is the solution that yields the smallest positive time-of-flight.

4.2.2. Numerical integration approach

A more common approach to obtain the streamline path is to use numerical integration in time. Because the velocity field is known analytically, an efficient numerical integration is possible by solving an initial value problem for the streamline location $\hat{\mathbf{x}}$ on the reference element:

$$\frac{d\hat{\mathbf{x}}}{d\hat{t}} = \hat{\mathbf{u}}(\hat{\mathbf{x}}), \quad \hat{\mathbf{x}}(\hat{t} = \hat{t}_0) = \hat{\mathbf{x}}_0. \quad (49)$$

In our implementation, we used an explicit Runge–Kutta method of fourth order [29], which proved to be very efficient and robust.

4.3. Time-of-flight computation

The time-of-flight τ along a streamline is given by:

$$\tau := \int_{\mathcal{L}} \frac{1}{|\mathbf{u}(s)|} ds, \quad (50)$$

where s represents the arc length along the streamline \mathcal{L} .

In two cases only, the time-of-flight variable can be computed analytically. First, for the RT_0 discretization on triangles, the velocity field is constant over each element; therefore, the streamline is a straight line and the time-of-flight computation is trivial. The second case is that of Pollock's method [7], where an RT_0 discretization is used on rectangular grids. In this case, each component of the velocity field depends only on its own coordinate: $u_x(x,y) = u_x(x)$, and $u_y(x,y) = u_y(y)$. This decoupling of the coordinates permits an analytical integration of the time-of-flight.

For general triangular or quadrilateral grids, and for higher-order velocity approximations, a numerical integration of the time-of-flight is necessary. For consistency with the rest of the streamline tracing framework, one must be able to evaluate the integral in Eq. (50) on the reference space. Using the Piola transform, we can express the reference velocity in terms of the physical one and recast the time-of-flight integration in terms of the reference coordinates only [22]:

$$\tau = \int_{\hat{\mathcal{L}}} \frac{1}{|\hat{\mathbf{u}}(\hat{s})|} J(\hat{x}) d\hat{s}. \quad (51)$$

It is worthwhile noting that the isoparametric mapping between the reference and physical spaces is affine for triangular and rectangular elements. In this case, the Jacobian is constant over the reference element and can be taken out of the integral in Eq. (51). For general quadrilateral elements, however, the Jacobian varies inside the element. Prévost et al. [9] used the value of the Jacobian at the center of the element as an approximation. Hægland [10] and Jimenez et al. [11] showed, however, that this choice may lead to erroneous results and recommends keeping the Jacobian inside the integral. This explains why the time-of-flight cannot be integrated analytically on general quadrilaterals even when a low-order approximation is employed.

In our implementation, the time-of-flight integral of Eq. (51) is computed using a quadrature rule when the stream function approach is used for the integration of the streamline path. When the streamline path is integrated numerically, the time-of-flight is computed within the Runge–Kutta stepping.

5. Results

5.1. Validation examples

In this section, we test the performance of the proposed streamline tracing approach on cases of increasing complexity. In particular, we systematically compare the behavior of low-order and high-order tracing, both in terms of accuracy and grid sensitivity. All examples are defined in terms of dimensionless quantities.

5.1.1. Uniform flow field

This first test case is designed for validation purposes. It tests the ability of the tracing method to produce exact results for a constant velocity field. The domain considered is a square of dimensions 1×1 . The permeability is assumed to be homogeneous, isotropic and equal to 1. Fixed pressures are imposed on the left and right boundaries and they take values of 1 and 0, respectively. The top and bottom boundaries are impermeable. In this case, the exact streamlines are horizontal straight lines. The time required for a fluid particle to travel through the entire domain is identically equal to 1.

To test the robustness and accuracy of the tracing algorithm on this simple problem, the domain was discretized with a variety of grids (Fig. 8):

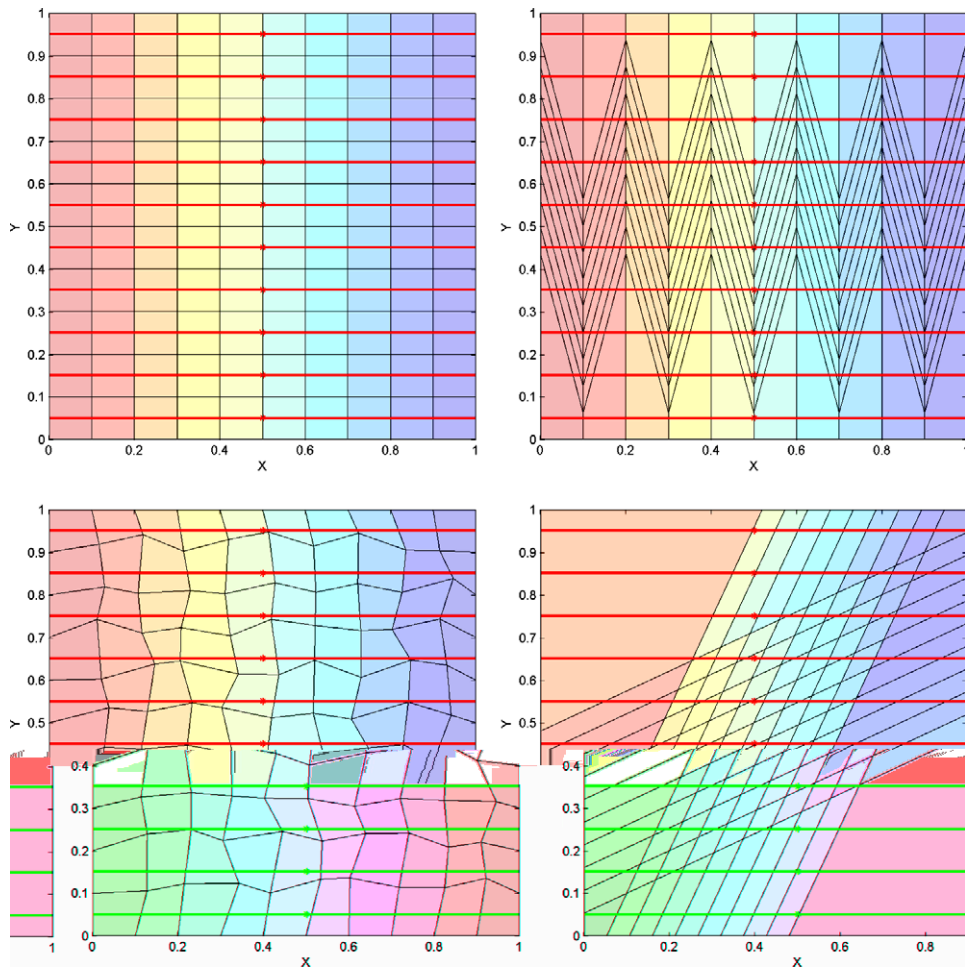


Fig. 8. RT_0^0 streamlines for the uniform flow problem on a Cartesian (top-left), chevron (top-right), random (bottom-left) and skewed (bottom-right) grids.

- A 10×10 *Cartesian* grid serves as the base grid for our comparisons.
- The *chevron* grid is formed by keeping the vertical lines of the Cartesian grid and reorienting the horizontal edges to obtain a chevron-like pattern.
- A random movement of the Cartesian grid nodes creates the *random* grid. The node movement is bounded to ensure element convexity.
- The *skewed* grid is obtained from a diagonal distortion of the Cartesian grid.

All four grids have 100 elements. Four triangular grids were created from these quadrilateral grids by splitting each quadrilateral into two triangles. By construction, the triangular grids have twice as many elements as the quadrilateral grids: they are all composed of 200 elements.

Ten streamlines were traced on each grid. Fig. 8 shows the streamlines traced on the four quadrilateral grids using the low-order RT_0^0 approximation. The streamlines are drawn with thick red lines, and the grid with thin black lines. The background colors represent the pressure field.

The results obtained in this first test case indicate that regardless of the type of grid used (Cartesian, chevron, random or skewed), the type of building elements (triangles or quadrilaterals), or the type of discretization (RT_0^0 or BDM_1^0), all streamlines traced are perfect straight lines. Unit time-of-flight were invariably obtained and, therefore, the tracing algorithm produces an exact time-of-flight in this case. This is to be

expected given that both RT_0^0 and BDM_1^0 elements satisfy the patch-test [30], that is, they can reproduce constant velocity fields in two dimensions.

5.1.2. Convergence study

This test case is designed to analyze the convergence of our streamline tracing method and compare the accuracy of the RT_0^0 and BDM_1^0 streamlines.

A base model is defined on a 4×4 Cartesian grid, shown in Fig. 9. The boundary edges of the bottom-left and top-right elements – drawn with a thicker line – are set to constant pressure values of 1 and 0, respectively. The remaining boundaries are impermeable.

To study the convergence of the streamline tracing method, we track the evolution of the error on the time-of-flight as the grid is refined. We do so for a streamline launched from the point of coordinates $(1/8, 7/8)$, which is at the center of the top-left element in the base grid. This individual streamline is represented in Fig. 9 by a thick red line.

Fig. 10 shows the results of the refinement study. The time-of-flight error is plotted as a function of the inverse of the typical length scale of the grid used, h , which in our case is the length of an element edge. For a grid of typical length scale h , the time-of-flight error ϵ_h is computed through:

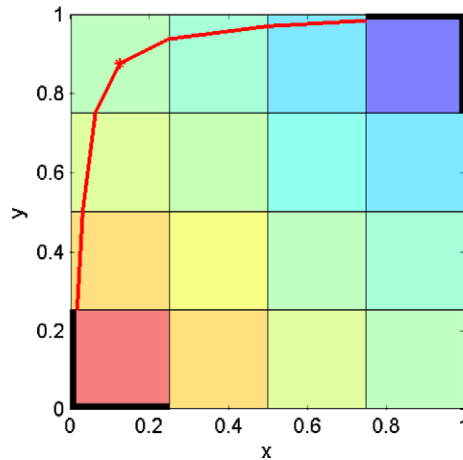


Fig. 9. Base grid, boundary conditions and streamline analyzed in the convergence study.

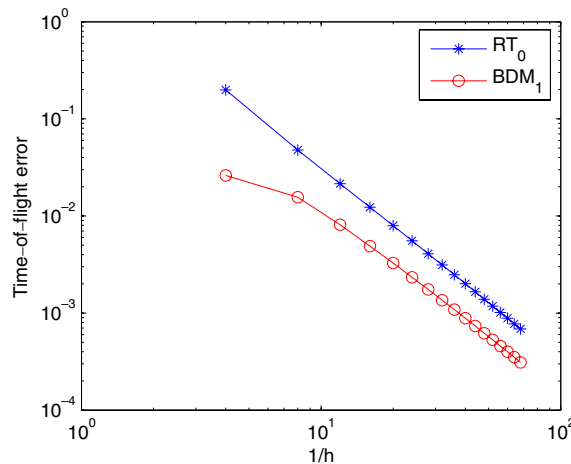


Fig. 10. Convergence of the RT_0^0 and BDM_1^0 streamlines, in terms of time-of-flight.

$$\varepsilon_h = \frac{|\tau_{h'} - \tau_h|}{\tau_{h'}}, \quad \text{with } h' < h, \quad (52)$$

where h' defines a finer grid than does h .

We clearly see that both discretizations converge and that they do so at the same rate. The asymptotic rate of convergence is of order h^2 in both cases. However, for a given grid size, the streamline traced using a BDM_1^0 discretization always displays a lower error than the one using an RT_0^0 discretization.

5.1.3. Robustness to grid distortion

To quantify the robustness of the low-order and high-order versions of the streamline tracing algorithm, we study how much grid distortion affects the accuracy of the streamlines in terms of location and time-of-flight.

Our base case is a 10×10 Cartesian grid discretizing a unit square domain of unit permeability. Pressures of 1 and 0 are imposed at the boundary edges of the bottom-left and top-right elements, respectively. The rest of the domain boundary is impermeable. The grids presented in Section 5.1.1 are used here again, the only difference being that elements where pressure boundary conditions are set are not distorted, so that identical boundary conditions are applied to all grids.

Seven streamlines are traced on every grid and for each type of discretization. These streamlines are launched from equally spaced points along the diagonal of the domain. As an illustration, Figs. 11 and 12 compare the streamlines traced using the RT_0^0 and BDM_1^0 discretizations for the quadrilateral chevron and the triangular Cartesian grids, respectively. These two examples clearly show the strong influence of grid distortion on streamline accuracy for an RT_0^0 discretization as well as the improved robustness demonstrated by BDM_1^0 . The higher-order method yields streamlines that are much smoother and much less sensitive to the distortion of the underlying grid. Notice, for example, the severe degree of nonsymmetry present in the streamlines computed using the low-order method on the chevron grid (Fig. 11): the center streamline displays a tortuous path, when it should be a perfect straight line. This behavior improves dramatically when the higher-order approximation is used.

To quantify the apparent increased accuracy and robustness of the streamlines based on the higher-order velocity approximation, we compute the error in the time-of-flight for each of the seven streamlines. Reference values are obtained using an 80×80 Cartesian grid with a BDM_1^0 discretization. In Table 1 we report the average (arithmetic mean) time-of-flight error for each grid and type of discretization. We note the following observations:

- (1) For all grids, the time-of-flight error is lower – sometimes much lower – if the high-order BDM_1^0 approximation is used, rather than the low-order RT_0^0 approximation. BDM_1^0 is more accurate than RT_0^0 by a factor of 5–15 in the case of quadrilateral grids, and by a factor of 2–4 for triangular grids.

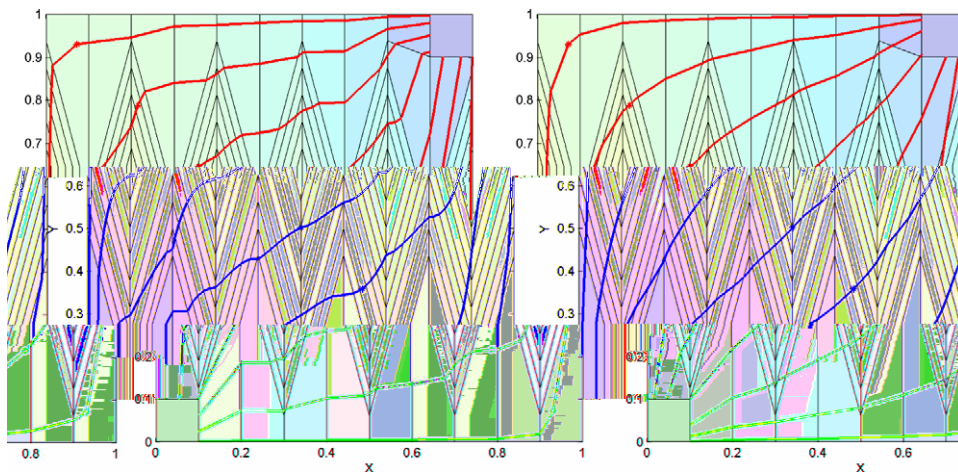


Fig. 11. RT_0^0 (left) and BDM_1^0 (right) streamlines on a quadrilateral chevron grid.

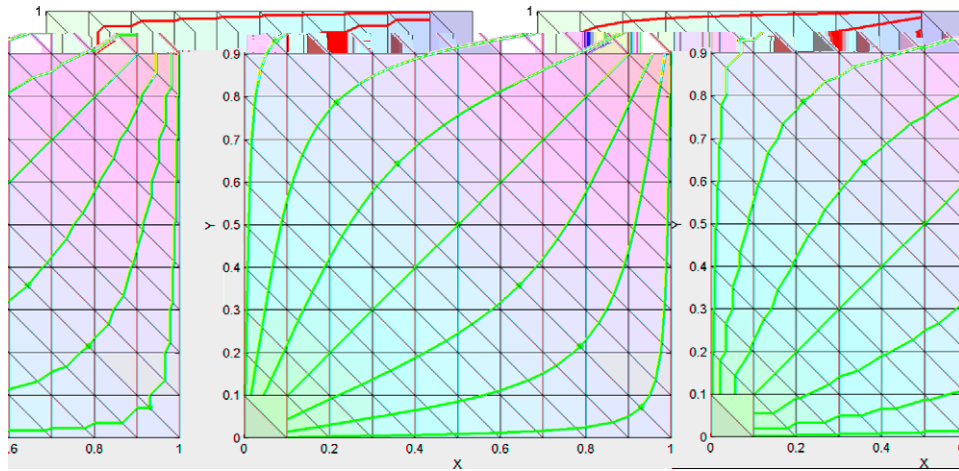


Fig. 12. RT_0^0 (left) and BDM_1^0 (right) streamlines on a triangular Cartesian grid.

Table 1

Average time-of-flight error for the diagonal flow problem

Element type	Discretization	Cartesian (%)	Chevron (%)	Random (%)	Skewed (%)
Triangle	RT_0^0	11.39	16.09	13.05	9.14
	BDM_1^0	4.06	4.62	4.12	4.02
Quadrilateral	RT_0^0	12.21	41.29	13.09	44.58
	BDM_1^0	2.45	5.51	2.13	2.85

- (2) The variability of the BDM_1^0 time-of-flight error is much smaller than that of RT_0^0 , confirming the robustness of the BDM_1^0 -based streamline tracing with respect to grid distortion.

5.2. Representative simulations

5.2.1. Accuracy in the presence of heterogeneity

Heterogeneity of the medium – reflected in a discontinuous permeability field that may vary several orders of magnitude – is an essential characteristic of petroleum reservoirs. For this reason, we test the accuracy of our tracing method in heterogeneous domains. We employ a test case from an interesting study by Mosé et al. [16] that compares the performance of several discretization methods in the presence of heterogeneity. The permeability field is shown in Fig. 13. This permeability field turns out to be challenging for some discretization methods, and we use it here to assess the quality of the RT_0^0 and BDM_1^0 streamlines. In this test case, pressures of 1 and 0 are set at the top and bottom boundaries of the domain, respectively. The left and right boundaries are impermeable. A total of 19 streamlines are launched from equally spaced points located on the top boundary.

The 10×10 quadrilateral Cartesian grid that constitutes the base case shown in Fig. 13 was refined into 20×20 and 40×40 Cartesian grids. Each of these grids was also transformed into a triangular grid by subdividing each quadrilateral element into two triangles.

Figs. 14 and 15 compare the RT_0^0 and BDM_1^0 streamlines on the 20×10 and 80×40 triangular grids, respectively. It is important to note that the streamlines obtained with all grids and discretization orders are physical: they avoid entirely the low permeability regions, in agreement with the findings of Mosé et al. [16]. The streamlines computed with the higher-order BDM_1^0 discretization, however, are smoother and less sensitive to the level of refinement. We quantify these observations by computing the time-of-flight errors on the streamlines using a 80×80 Cartesian grid with a BDM_1^0 discretization as a reference. The results are presented in Table 2. As expected, the BDM_1^0 errors are always much lower than those of RT_0^0 .

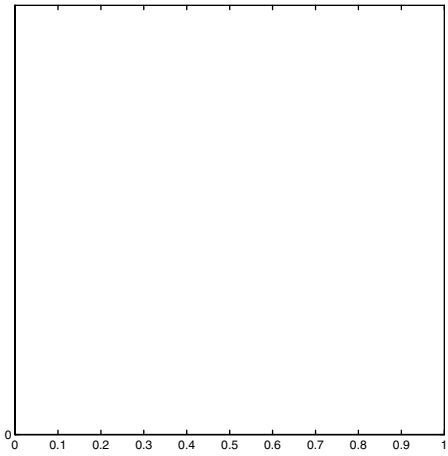


Table 2
Average time-of-flight error in the presence of heterogeneity

Element type	Discretization	10 × 10 (%)	20 × 20 (%)	40 × 40 (%)
Triangle	RT ₀ ⁰	13.39	6.55	3.49
	BDM ₁ ⁰	3.57	1.61	0.23
Quadrilateral	RT ₀ ⁰	8.58	4.19	2.20
	BDM ₁ ⁰	5.64	2.00	0.02

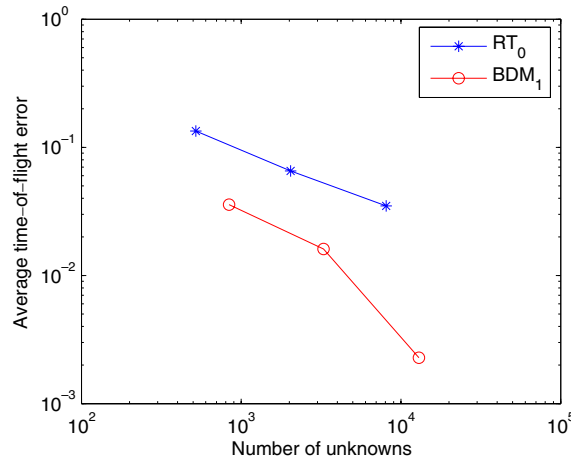


Fig. 16. Average time-of-flight error as a function of the number of unknowns for a sequence of triangular grids.

A legitimate question to ask is what is the incremental cost associated with the improved accuracy of BDM₁⁰. By now, we have established that for a given grid, BDM₁⁰ streamlines are more accurate than RT₀⁰ ones. To obtain this accuracy, however, one must solve for roughly twice as many velocity unknowns. In Fig. 16, we plot the average time-of-flight errors for triangular grids as a function of the number of unknowns. Clearly, for a given number of unknowns, BDM₁⁰ is more accurate than RT₀⁰. Refining the grid with an RT₀⁰ discretization is not as efficient as increasing the order of accuracy of the method by using the BDM₁⁰ space. Moreover, the slopes of the two curves are different: BDM₁⁰ seems to converge faster than RT₀⁰ as the number of unknowns is increased, although this could be an artifact of not having yet reached the asymptotic convergence regime.

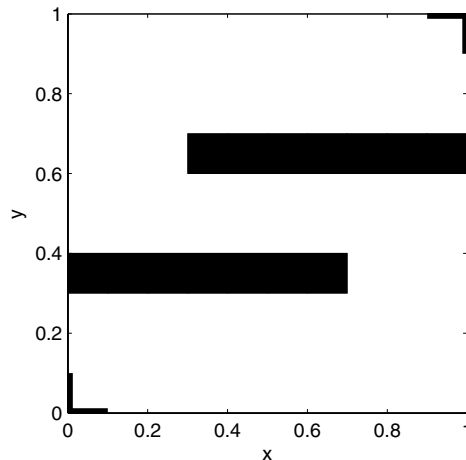


Fig. 17. Base case for the unstructured grid discretization.

5.2.2. Heterogeneous unstructured grids

Our last experiment tests the ability of the tracing method to deal with more realistic reservoir simulation grids such as unstructured grids and heterogeneous permeability fields. The domain, presented in Fig. 17, is a 1×1 square of unit permeability except for two flow barriers of low (10^{-3}) permeability represented by the black elements. Once again, pressures of 1 and 0 are set at the bottom-left and top-right corners (marked by thicker boundary lines) and the rest of the boundaries are impermeable.

Streamlines are launched from ten points located in the center of the domain. Fig. 18 shows a comparison of the RT_0^0 and BDM_1^0 streamlines for this problem using the same unstructured grid of 148 triangular elements. As before, the grid is shown in thin black lines, the streamlines with thick red lines, and their launching points are marked by a red star. The streamlines computed by the low-order and high-order approximations on a refined grid of 350 elements are shown in Fig. 19. In agreement with previous observations, the BDM_1^0 streamlines are smoother and much less sensitive to the grid refinement level.

To compute the time-of-flight errors committed on this unstructured grid, similar streamlines were traced on an 80×80 Cartesian grid with a BDM_1^0 discretization. Table 3 presents the time-of-flight errors committed for both the RT_0^0 and BDM_1^0 discretizations on the 148-element and 350-element grids. For both grids, the BDM_1^0 streamlines are about five times more accurate than the RT_0^0 ones.

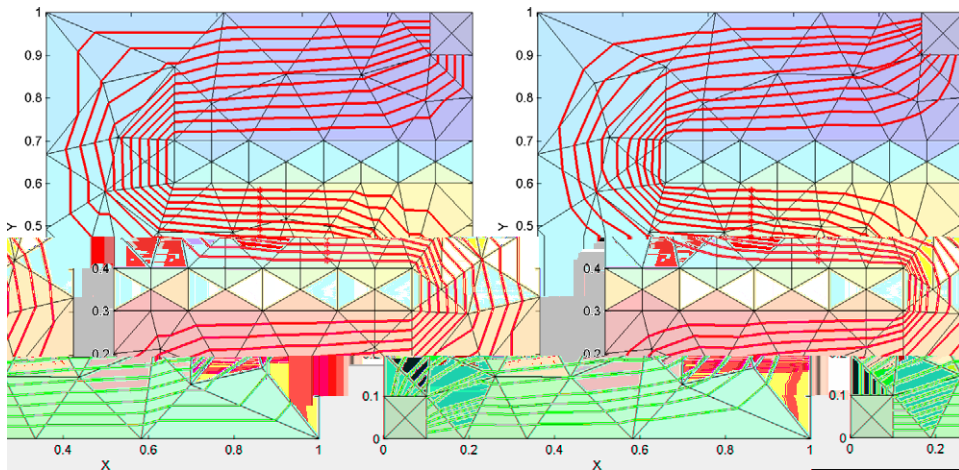


Fig. 18. RT_0^0 (left) and BDM_1^0 (right) streamlines for a 148-element grid.

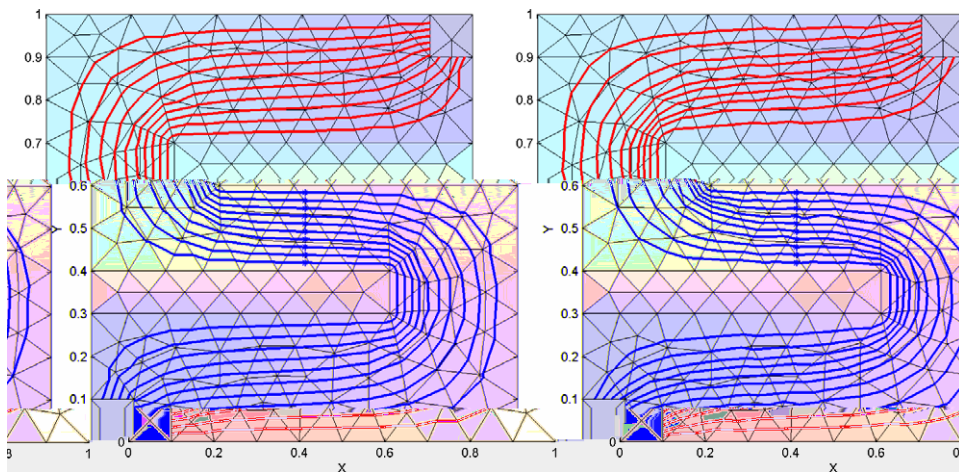


Fig. 19. RT_0^0 (left) and BDM_1^0 (right) streamlines for a 350-element grid.

Table 3
Average time-of-flight error for the heterogeneous unstructured grid test case

Discretization	148 elements (%)	350 elements (%)
RT_0^0	15.20	7.25
BDM_1^0	3.17	1.71

6. Conclusions and future work

In this paper, we presented a rather general approach for streamline tracing on two-dimensional unstructured grids, motivated by the need for accurate streamlines – in terms of streamline location and time-of-flight – in groundwater transport and petroleum reservoir simulation. The proposed streamline tracing approach is strongly linked to a mixed finite element discretization of the governing pressure equation, that is, a method that solves for pressure and Darcy velocity simultaneously. Our approach has three distinctive ingredients: (1) the use of a proper mapping of the velocity (the Piola transform) from physical to reference space; (2) the identification of suitable functional spaces for the approximation of the velocity field; and (3) the use of the stream function – induced by the choice of the particular velocity spaces – to determine the streamline location.

In a separate publication [22] we identify a large class of velocity spaces satisfying the condition that their restriction to divergence-free fields induces a stream function. Here, we concentrate on a small subset of the mixed velocity–pressure approximations identified in [22]: the lowest-order Raviart–Thomas space (RT_0) and the Brezzi–Douglas–Marini space of order one (BDM_1), both on triangles and quadrilaterals. We term these two choices as low-order and high-order tracing, respectively.

We analyzed the performance of the proposed tracing strategy by means of carefully designed test cases, and we assessed the potential benefit of tracing streamlines based on high-order accurate velocity spaces. The main result of this investigation can be summarized as follows: high-order BDM_1 -based tracing is much more accurate (smaller error in the time-of-flight) and robust (less sensitive to grid distortion) than low-order RT_0 -based tracing. In particular, we found that high-order tracing is more accurate than low-order tracing for a given number of unknowns, which suggests that refining the grid with an RT_0 discretization is not as efficient as increasing the order of accuracy of the method by using the BDM_1 space.

The approach described in this paper is limited to two-dimensional problems. Conceptually, the streamline tracing framework proposed here can be extended to three-dimensional problems. However, to obtain the analytical streamline path in three dimensions (Section 4.2.1), one needs to derive dual stream functions [31]. This can be achieved for RT_0^0 but the increased complexity of the BDM_1^0 velocity fields makes the problem challenging. We are currently extending the proposed streamline tracing approach to three dimensions using a numerical integration of the streamlines (Section 4.2.2), to provide a thorough comparison of low-order and high-order tracing. We are interested in assessing whether BDM_1 -based tracing is more effective than RT_0 -based tracing, especially given the fact that – unlike in 2D – both RT_0 and BDM_1 elements cannot reproduce uniform flow on general hexahedral grids [32]. We also plan to compare the results with an alternative approach recently proposed by Hægland et al. [12], which is based on a corner velocity interpolation that preserves uniform flow in 3D.

Acknowledgement

We gratefully acknowledge financial support from the industrial affiliates of the Stanford University Petroleum Research Institute for Reservoir Simulation (SUPRI-B).

References

- [1] K. Aziz, A. Settari, *Petroleum Reservoir Simulation*, Elsevier, London, 1979.
- [2] G. Chavent, J. Jaffré, *Mathematical Models and Finite Elements for Reservoir Simulation*, Elsevier, North-Holland, 1986.
- [3] Z. Chen, R.E. Ewing, Comparison of various formulations of three-phase flow in porous media, *J. Comput. Phys.* 132 (1997) 362–373.

- [4] F. Bratvedt, K. Bratvedt, C.F. Buchholz, T. Gimse, H. Holden, L. Holden, N.H. Risebro, Frontline and Frontsim, two full scale, two-phase, black oil reservoir simulators based on front tracking, *Surv. Math. Ind.* 3 (1993) 185–215.
- [5] R.P. Batycky, M.J. Blunt, M.R. Thiele, A 3D field-scale streamline-based reservoir simulator, *SPE Reserv. Eng.* 11 (4) (1997) 246–254.
- [6] M.J. King, A. Datta-Gupta, Streamline simulation: A current perspective, *In Situ* 22 (1) (1998) 91–140.
- [7] D.W. Pollock, Semianalytical computation of path lines for finite difference models, *Ground Water* 26 (1988) 743–750.
- [8] C. Cordes, W. Kinzelbach, Continuous groundwater velocity fields and path lines in linear, bilinear, and trilinear finite elements, *Water Resour. Res.* 28 (11) (1992) 2903–2911.
- [9] M. Prevost, M.G. Edwards, M.J. Blunt, Streamline tracing on curvilinear structured and unstructured grids, *Soc. Pet. Eng. J.* 7 (2) (2002) 139–148.
- [10] H. Hægland, Streamline tracing on irregular grids, Cand. Scient Thesis, University of Bergen, Dept. of Mathematics (Dec.) (2003).
- [11] E. Jimenez, K. Sabir, A. Datta-Gupta, M.J. King, Spatial error and convergence in streamline simulation, in: *SPE Reservoir Simulation Symposium*, Houston, TX, 2005, (SPE 92873).
- [12] H. Hægland, H.K. Dahle, G.T. Eigestad, K.-A. Lie, I. Aavatsmark, Improved streamlines and time-of-flight for streamline simulation on irregular grids, *Adv. Water Resour.*, submitted for publication.
- [13] J.E. Marsden, T.J.R. Hughes, *Mathematical Foundations of Elasticity*, Prentice-Hall, Englewood Cliffs, NJ, 1983, reprinted with corrections, Dover, New York, 1994.
- [14] F. Brezzi, M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer, Verlag, New York, 1991.
- [15] L.J. Durlofsky, Accuracy of mixed and control volume finite element approximations to Darcy velocity and related quantities, *Water Resour. Res.* 30 (4) (1994) 965–973.
- [16] R. Mosé, P. Siegel, P. Ackerer, G. Chavent, Application of the mixed hybrid finite element approximation in a groundwater flow model: Luxury or necessity? *Water Resour. Res.* 30 (11) (1994) 3001–3012.
- [17] C. Cordes, W. Kinzelbach, Comment on “Application of the mixed hybrid finite element approximation in a groundwater flow model: Luxury or necessity?” by R. Mosé, P. Siegel, P. Ackerer, and G. Chavent, *Water Resour. Res.* 32 (6) (1996) 1905–1909.
- [18] P. Ackerer, R. Mosé, P. Siegel, G. Chavent, Application of the mixed hybrid finite element approximation in a groundwater flow model: Luxury or necessity? Reply to the Comment by C. Cordes and W. Kinzelbach, *Water Resour. Res.* 32 (6) (1996) 1911–1913.
- [19] E.F. Kaasschieter, Mixed finite elements for accurate particle tracking in saturated groundwater flow, *Adv. Water Resour.* 18 (5) (1995) 277–294.
- [20] S.F. Matringe, M.G. Gerritsen, On accurate tracing of streamlines, in: *SPE Annual Technical Conference and Exhibition*, Houston, TX, 2004, (SPE 89920).
- [21] S.F. Matringe, R. Juanes, H.A. Tchelepi, Streamline tracing on general triangular or quadrilateral grids, in: *SPE Annual Technical Conference and Exhibition*, Dallas, TX, 2005, (SPE 96411).
- [22] R. Juanes, S.F. Matringe, Unified formulation of velocity fields for streamline tracing on two-dimensional unstructured grids, *Comput. Methods Appl. Mech. Eng.*, submitted for publication.
- [23] S.C. Brenner, L.R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer Verlag, New York, 1994.
- [24] I. Babuška, The finite element method with Lagrangian multipliers, *Numer. Math.* 20 (1973) 179–192.
- [25] F. Brezzi, On the existence, uniqueness and approximation of saddle point problems arising from Lagrange multipliers, *RAIRO Anal. Numér.* 8 (1974) 129–151.
- [26] R. Glowinsky, P. Le Tallec, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, SIAM, Philadelphia, PA, 1989.
- [27] P.A. Raviart, J.M. Thomas, A mixed finite element method for second order elliptic problems, in: I. Galligani, E. Magenes (Eds.), *Mathematical Aspects of the Finite Element Method*, Lecture Notes in Mathematics, vol. 606, Springer-Verlag, New York, 1977, pp. 292–315.
- [28] F. Brezzi, J. Douglas Jr., L.D. Marini, Two families of mixed finite elements for second order elliptic problems, *Numer. Math.* 47 (1985) 217–235.
- [29] J.R. Dormand, P.J. Prince, A family of embedded Runge–Kutta formulae, *J. Comput. Appl. Math.* 6 (1980) 19–26.
- [30] O.C. Zienkiewicz, R.L. Taylor, *The Finite Element Method* (3 vols), 1: The Basis, 2: Solid Mechanics, 3: Fluid Dynamics, 5th ed., Butterworth-Heinemann, Oxford, 2000.
- [31] J. Bear, *Dynamics of Fluids in Porous Media* Environmental Science Series, Elsevier, New York, 1972, reprinted with corrections, Dover, New York, 1988.
- [32] R.L. Naff, T.F. Russell, J.D. Wilson, Shape functions for velocity interpolation in general hexahedral cells, *Comput. Geosci.* 6 (3–4) (2002) 285–314.