Adaptive Moving Mesh Modeling for Two Dimensional Groundwater Flow and Transport

Weizhang Huang and Xiaoyong Zhan

Abstract. An adaptive moving mesh method is presented for numerical simulation of two dimensional groundwater flow and transport problems. A selection of problems are considered, including advection dominated chemical transport and reaction, solute transport from contamination sources, transport of nonaqueous phase liquids (NAPLs) in an aquifer, and coupling of groundwater flow with NAPL transport. Numerical results show that the adaptive moving mesh method is able to capture sharp moving fronts and detect the emerging of new fronts.

1. Introduction

Groundwater resources are vital to the natural environment, social welfare, and economic health. In the past two decades, numerical simulation has proved to be an effective tool for simulating and predicting behaviors of chemical contamination in the subsurface that can degrade groundwater quality. Indeed, a variety of algorithms and software have been developed for the numerical simulation of groundwater flow and transport, e.g., see [25, 31] and references therein. Meanwhile, the quest for more accurate and efficient numerical methods remains active. This is especially true for problems exhibiting sharp moving fronts, for which conventional methods tend to produce oscillatory solutions and excessive numerical dispersion in regions around sharp fronts.

It has been amply demonstrated that, by placing more mesh nodes in the regions around sharp fronts than in the rest of the domain, mesh adaptation provides an effective tool in reducing numerical dispersion and oscillation while enhancing the efficiency and accuracy of numerical simulation. Roughly speaking, mesh adaptation can be cast into two main categories, the refinement method and the dynamic or moving mesh method. The first type of method is conceptually simpler. It achieves mesh adaptivity by adding or removing mesh nodes locally. The refinement method has been successfully applied to a number of groundwater problems in one and two dimensions. For example, Yeh et al. [29] combine the method of characteristics with local mesh refinement and effectively avoid the numerical dispersion and

1991 *Mathematics Subject Classification.* 65M50, 65M60, 86A05.

oscillation problem in solving advection-dispersion equations. Trompert [26] applies local-uniform-mesh refinement to modeling transport in heterogeneous media.

On the other hand, the moving mesh method [11, 14, 23] achieves mesh adaptivity by relocating the node positions while keeping the number of mesh nodes and the mesh connectivity fixed throughout the entire solution process. Since the size of computation and data structure are kept fixed, it is much easier to implement the moving mesh method than the refinement method. Moreover, the moving mesh method often works better because the relocation of node positions improves the alignment of mesh elements with the solution and thus the simulation accuracy. Unfortunately, it has been known that it is difficult to formulate a reliable moving mesh method in multi-dimensions. So far, the moving mesh method has been successfully applied to groundwater modeling only in one dimension; see [8–10, 17, 30].

Progress has been made recently in developing more reliable moving mesh methods in multi-dimensions. The development has mainly focused on the variational approach where the node positions are obtained by minimizing a functional measuring some difficulty in approximating the physical solution. Such a functional can be formulated as the energy functional of harmonic mappings [3,7,20,28] or based on a Jabobian weighting [18], interpolation error estimates [12, 16], or a combination of various mesh properties [1, 2]. The reader is also referred to [4] for brief comparison of variational-type moving mesh methods with those based on mesh velocity.

In this work, we focus on a particular variational-type moving mesh method, the so-called moving mesh PDE (MMPDE) approach developed by Huang et al. [14, 15]. With this approach, adaptive moving meshes are generated as images of a fixed computational mesh in the auxiliary domain through a time dependent coordinate transformation. The transformation is obtained as the solution of the gradient flow equation of an adaptation functional which is related to the well-known equidistribution principle [6] and measures the difficulty in approximating the solution. Since the mesh nodes are continuously relocated and dynamically adapted to solution behavior, the MMPDE moving mesh method provides an ideal adaptive strategy to capture evolving sharp fronts without using a large number of grid points. In [17], the method has been used to model a range of one dimensional groundwater problems, including of advection dominated chemical transport and reaction, non-linear infiltration in soil, and the coupling of density dependent flow and transport.

The objective of this paper is to investigate the applicability of the MMPDE moving mesh method to two dimensional groundwater flow and transport problems. Although the moving mesh method has been successfully applied to a number of one dimensional groundwater problems [8–10, 17, 30], there lacks in published work in two dimensions. Moreover, it is far from clear that the method is necessarily able to capture sharp two dimensional moving fronts. This is because two dimensional moving fronts have far more complicated structures and thus more difficult to simulate and track. For demonstration purpose, a range of problems common in groundwater flow and transport are considered. They include advection dominated chemical transport and reaction, solute transport from contamination sources, transport of nonaqueous phase liquids (NAPLs) in an aquifer, and coupling of groundwater flow with NAPL transport.

The paper is organized as follows. The MMPDE moving mesh method and its implementation are presented in section 2. In section 3, the groundwater problems are described. The simulation results using the adaptive moving mesh method are also given in the same section. Finally, the summary and comments are given in Section 4.

2. The MMPDE moving mesh method

The MMPDE moving mesh method is developed in [14] in one dimension and in [15] for multi-dimensional problems. The basic idea is to define the time-dependent coordinate transformation needed for generating adaptive meshes as the solution of an MMPDE (moving mesh PDE), which in turn is defined as the gradient flow equation of an adaptation functional measuring the difficulty in approximating the physical solution. The physical PDE is discretized and integrated with the quasi-Lagrange approach where the effect of mesh movement is reflected by extra terms involving mesh velocity in the transformed physical PDE.

For the clarity of presentation, the MMPDE moving mesh method is described for the following simplified version of the advection-dispersion-reaction equation (ADRE), one of typical mathematical models describing chemical transport in the subsurface:

$$R\frac{\partial C}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial C}{\partial x}\right) + \frac{\partial}{\partial y}\left(D\frac{\partial C}{\partial y}\right) - V_1\frac{\partial C}{\partial x} - V_2\frac{\partial C}{\partial y} - \lambda RC, \tag{2.1}$$

where $C = C(t,x,y)$ is the concentration of a certain type of chemical, $D = D(t,x,y)$ is the dispersivity, $V = (V_1(t,x,y), V_2(t,x,y))$ is the Darcy velocity, R is the retardation factor, and λ is the reaction factor. It should be emphasized that the method can be straightforwardly applied to other systems.

2.1. Spatial discretization on a moving mesh. In a moving mesh method, the mesh is generated as the image of a uniform logical mesh under the coordinate transformation (denoted by $x = x(t,\xi,\eta)$, $y = y(t,\xi,\eta)$) from the logical domain (Ω_c) to the physical domain (Ω). As a common practice, we choose the logical domain as the the unit square, i.e., $\Omega_c = (0,1)\times(0,1)$. Given two positive integers J and K, let $\Delta\xi = 1/J$ and $\Delta\eta = 1/K$. With the uniform logical mesh being denoted by

$$\xi_j = j\Delta\xi, \quad \eta_k = k\Delta\eta \qquad j = 0,...,J,\ k = 0,...,K \tag{2.2}$$

the moving mesh can be written as

$$x_{j,k}(t) = x(t,\xi_j,\eta_k), \quad y_{j,k}(t) = y(t,\xi_j,\eta_k) \qquad j = 0,...,J,\ k = 0,...,K. \tag{2.3}$$

Finite differences are used for the spatial discretization of equation (2.1). To this end, (2.1) is first transformed from the physical coordinates to the logical ones. Let

$$c = c(t,\xi,\eta) = C(t, x(t,\xi,\eta), y(t,\xi,\eta)).$$

Then, (2.1) can be written in the new coordinates as

$$R\frac{\partial c}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial c}{\partial x}\right) + \frac{\partial}{\partial y}\left(D\frac{\partial c}{\partial y}\right) - (V_1 - x_t)\frac{\partial c}{\partial x} - (V_2 - y_t)\frac{\partial c}{\partial y} - \lambda Rc, \tag{2.4}$$

where (x_t, y_t) is the mesh velocity. Note that the partial derivatives with respect to x and y in (2.4) should be understood through the chain-rule, viz.,

$$\frac{\partial}{\partial x} = \xi_x\frac{\partial}{\partial \xi} + \eta_x\frac{\partial}{\partial \eta}, \quad \frac{\partial}{\partial y} = \xi_y\frac{\partial}{\partial \xi} + \eta_y\frac{\partial}{\partial \eta}, \tag{2.5}$$

with the transformation relations being given by $J = x_\xi y_\eta - x_\eta y_\xi$ and

$$\begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix}^{-1} = \frac{1}{J}\begin{bmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{bmatrix}. \tag{2.6}$$

The approximation of the partial derivatives using central finite differences on the uniform logical mesh is standard. For example,

$$\begin{aligned} \left.\frac{\partial c}{\partial x}\right|_{j,k} &= \left.\left(\xi_x \frac{\partial c}{\partial \xi} + \eta_x \frac{\partial c}{\partial \eta}\right)\right|_{j,k} \\ &\approx \xi_x|_{j,k} \frac{c_{j+1,k} - c_{j-1,k}}{2\Delta\xi} + \eta_x|_{j,k} \frac{c_{j,k+1} - c_{j,k-1}}{2\Delta\eta}, \end{aligned}$$

where $\xi_x|_{j,k}$ and $\eta_x|_{j,k}$ are calculated through (2.6). Some cautions may be needed when discretizing the second order derivatives. Consider the first term on the right-hand side of (2.4). In the new coordinates it reads as

$$\frac{\partial}{\partial x}\left(D\frac{\partial c}{\partial x}\right) = \xi_x \frac{\partial}{\partial \xi}\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right) + \eta_x \frac{\partial}{\partial \eta}\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right).$$

To obtain a compact and stable scheme, we use half-point differences for approximating the outer differentiation,

$$\begin{aligned} \left.\frac{\partial}{\partial x}\left(D\frac{\partial c}{\partial x}\right)\right|_{j,k} \approx{}& \frac{\xi_x|_{j,k}}{\Delta\xi}\left[\left.\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right)\right|_{j+\frac{1}{2},k} - \left.\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right)\right|_{j-\frac{1}{2},k}\right] \\ &+ \frac{\eta_x|_{j,k}}{\Delta\eta}\left[\left.\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right)\right|_{j,k+\frac{1}{2}} - \left.\left(D\xi_x \frac{\partial c}{\partial \xi} + D\eta_x \frac{\partial c}{\partial \eta}\right)\right|_{j,k-\frac{1}{2}}\right]. \end{aligned}$$

The remaining differentiation can be approximated by the standard central differencing and averaging, e.g.,

$$\begin{aligned} \left.\frac{\partial c}{\partial \xi}\right|_{j,k\pm\frac{1}{2}} &\approx \frac{1}{2}\left[\frac{c_{j+1,k} - c_{j-1,k}}{2\Delta\xi} + \frac{c_{j+1,k\pm1} - c_{j-1,k\pm1}}{2\Delta\xi}\right], \\ \left.\frac{\partial c}{\partial \eta}\right|_{j\pm\frac{1}{2},k} &\approx \frac{1}{2}\left[\frac{c_{j,k+1} - c_{j,k-1}}{2\Delta\eta} + \frac{c_{j\pm1,k+1} - c_{j\pm1,k-1}}{2\Delta\eta}\right]. \end{aligned}$$

It is noted that the effect of mesh movement is reflected in the transformed equation (2.4) via the terms containing mesh velocity (x_t, y_t). For this reason, (2.4) is often said to be of the quasi-Lagrange form. With this quasi-Lagrange approach, there is no need for interpolation of the physical variable from the old mesh to the new mesh. This is in contrast to the so-called mesh rezoning approach, where the physical variable is interpolated from the old mesh to the new mesh and the physical PDE is discretized and integrated on the new mesh (that is considered fixed in the current time step); see [11, 20] for more discussion on this approach.

2.2. The MMPDE approach of mesh movement. The idea of the MMPDE approach is to determine the time-dependent coordinate transformation needed for generating adaptive meshes as the solution of an MMPDE, which in turn is defined as the gradient flow equation of an adaptation functional measuring the difficulty in approximating the physical solution. To measure the difficulty in the numerical approximation of the solution, we define the so-called arc-length monitor function as

$$M(x,y) = I + \nabla C (\nabla C)^T = \begin{bmatrix} 1+C_x^2 & C_x C_y \\ C_x C_y & 1+C_y^2 \end{bmatrix}. \tag{2.7}$$

Generally we expect that the mesh is generated such that it closely satisfies the so-called equidistribution principle or the mesh density is approximately proportional to the square root of the determinant of the monitor function (e.g., see [16]). In this sense, the monitor function (2.7) places more mesh nodes in the regions of large gradient of the solution C. It is known that the arc-length monitor function is the simplest but not the best for all cases. The reader is referred to [16] for a more sophisticated definition of the monitor function based on interpolation error.

Given the monitor function, the adaptation functional is defined for the inverse coordinate transformation as

$$I[\xi,\eta] = \frac{1}{2}\int_\Omega \left((\nabla\xi)^T M^{-1} \nabla\xi + (\nabla\eta)^T M^{-1}\nabla\eta \right) dxdy, \tag{2.8}$$

where $\nabla = (\partial/\partial x, \partial/\partial y)$ is the gradient operator. The MMPDE is then defined [13] as the gradient flow equation of $I[\xi,\eta]$,

$$\frac{\partial \xi}{\partial t} = \frac{1}{\tau p}\nabla\cdot\left(M^{-1}\nabla\xi\right), \qquad \frac{\partial \eta}{\partial t} = \frac{1}{\tau p}\nabla\cdot\left(M^{-1}\nabla\eta\right), \tag{2.9}$$

where $p = p(t,x,y)$ is a positive function (cf. its definition in (2.11)) and $\tau > 0$ is a user-prescribed parameter for adjusting the time scale of mesh movement. The final form of the MMPDE is obtained by interchanging the roles of the independent and dependent variables in (2.9),

$$\begin{aligned} \frac{\partial x}{\partial t} &= \frac{1}{\tau p}\left[a_{11}\frac{\partial^2 x}{\partial \xi^2} + 2a_{12}\frac{\partial^2 x}{\partial\xi\partial\eta} + a_{22}\frac{\partial^2 x}{\partial\eta^2} + b_1\frac{\partial x}{\partial \xi} + b_2\frac{\partial x}{\partial\eta}\right], \\ \frac{\partial y}{\partial t} &= \frac{1}{\tau p}\left[a_{11}\frac{\partial^2 y}{\partial \xi^2} + 2a_{12}\frac{\partial^2 y}{\partial\xi\partial\eta} + a_{22}\frac{\partial^2 y}{\partial\eta^2} + b_1\frac{\partial y}{\partial \xi} + b_2\frac{\partial y}{\partial\eta}\right], \end{aligned} \tag{2.10}$$

where

$$\begin{aligned} &\boldsymbol{u} = \frac{1}{J}\begin{bmatrix} y_\eta \\ -x_\eta \end{bmatrix}, \quad \boldsymbol{v} = \frac{1}{J}\begin{bmatrix} -y_\xi \\ x_\xi \end{bmatrix}, \quad J = x_\xi y_\eta - x_\eta y_\xi, \\ &a_{11} = \boldsymbol{u}^T M^{-1}\boldsymbol{u}, \quad a_{12} = \boldsymbol{u}^T M^{-1}\boldsymbol{v}, \quad a_{22} = \boldsymbol{v}^T M^{-1}\boldsymbol{v}, \\ &b_1 = -\boldsymbol{u}^T\left(\boldsymbol{u}\frac{\partial M^{-1}}{\partial\xi} + \boldsymbol{v}\frac{\partial M^{-1}}{\partial\eta}\right), \quad b_2 = -\boldsymbol{v}^T\left(\boldsymbol{u}\frac{\partial M^{-1}}{\partial\xi} + \boldsymbol{v}\frac{\partial M^{-1}}{\partial\eta}\right), \end{aligned}$$

$$p = \sqrt{a_{11}^2 + a_{22}^2 + b_1^2 + b_2^2}. \tag{2.11}$$

Central finite differences are used to discretize the mesh equation (2.10) on the uniform logical mesh. In our computation, τ is taken as $\tau = 0.01$, and this value seems to work well for all the tested cases .

2.3. Time integration and solution procedure. In principle, the physical PDE (2.4) and the MMPDE (2.10) can be integrated in time either simultaneously or alternately. However, alternating solution seems more realistic in multi-dimensions since it voids the highly nonlinear coupling of the mesh and physical solution and preserves many structures such as ellipticity and sparsity in each of the mesh and physical PDEs. The following alternating procedure is used in our computations. Here, Δt is the time step size associated with the physical PDE and Δt_{mesh} for that related to the MMPDE.

Alternating Procedure: Assume that the physical solution c^n, the mesh (x^n, y^n), and a time step size Δt_n are given at time $t = t_n$.

(i) Compute the monitor function $M^n(x, y) = M(t_n, x, y)$ using c^n and (x^n, y^n). The solution derivatives used in M are calculated using a gradient recovery technique similar to that developed by Zienkiewicz and Zhu [32, 33] based on the nodal values of the computed solution. M^n is understood as a continuous function in the sense of interpolation.

(ii) Integrate the MMPDE (2.10) over the time period $[t_n, t_n + \Delta t_n]$ using variable step size $\Delta t_{mesh,n}$ and monitor function $M(x, y) = M^n(x, y)$. The MMPDE is discretized in time using the backward Euler scheme with the coefficients a_{11}, a_{12}, a_{22}, b_1, and b_2 being calculated at t_n. More than one sub-step may be needed for the integration to reach $t = t_n + \Delta t_n$. When this happens, the monitor function is updated for each sub-step via linear interpolation. The obtained mesh is denoted by (x^{n+1}, y^{n+1}).

(iii) Integrate the physical PDE (2.4) with a fixed or variable step size. The equation is discretized in time using the Singly Diagonally Implicit Runge-Kutta scheme [5]. The mesh and mesh velocity are calculated using linear interpolation:

$$x(t) = \frac{t - t_n}{\Delta t_n} x^{n+1} + \frac{t_n + \Delta t_n - t}{\Delta t_n} x^n, \quad y(t) = \frac{t - t_n}{\Delta t_n} y^{n+1} + \frac{t_n + \Delta t_n - t}{\Delta t_n} y^n.$$

(iv) When a variable step size is used in step (iii), the physical PDE may actually be integrated over a smaller step $\hat{\Delta} t_n < \Delta t_n$. In this case, the mesh at the actual new time level $t_{n+1} = t_n + \hat{\Delta} t_n$ should be updated as $(x^{n+1}, y^{n+1}) := (x(t_{n+1}), y(t_{n+1}))$.

(v) Go to the next step with the step size predicted by the physical PDE solver.

3. Applications

Example 3.1: Advection dominated chemical transport and reaction. Contaminant in form of ions, molecules, or particles undergoes multiple and complicated processes including transport, chemical, and biological ones in water environment. The transport process is mainly due to advection and dispersion. The chemical process covers acid-base reactions, solution, volatilization, precipitation, solute reactions, oxidation-reduction reactions, hydrolysis reaction, isotropic reactions, adsorption, and desorption. The biological process covers bacterias and virus activities. A quantitative description of these processes is prerequisite to the successful protection of water quality. The advection-dispersion-reaction equation (ADRE), Eq. (2.1), is one of most commonly used mathematical models describing these processes in geohydrology [19]. The equation can become advection-dominated or dispersion-dominated depending on the magnitude of the aquifer permeability and Darcy velocity. The challenge in the numerical solution of this equation is to capture the chemical transport and reaction front without introducing erroneous instability and numerical dispersion while maintaining reasonable computational efficiency. The physical domain is taken as $\Omega = (0, 1) \times (0, 1)$ in our computations.

We now consider a case which essentially is one-dimensional. We choose this test example because it has the exact solution and can be used to verify the computer code. Moreover, the one dimensional version of the ADRE with the same exact solution has been used as an example for testing the one dimensional MM-PDE moving mesh method in [17].

For this case, the physical parameters are taken as $D(t,x,y) = D = 10^{-5}$, $V_1(t,x,y) = V = 1$, $V_2(t,x,y) = 0$, and $R = 1.1$, and $\lambda = 1.1$. The initial and boundary conditions are defined as

(3.1)
$$C(0,x,y) = 0, \quad C(t,0,y) = 1, \quad C(t,1,y) = C^*, \quad \frac{\partial C}{\partial y}(t,x,0) = \frac{\partial C}{\partial y}(t,x,1) = 0,$$

where C^* is a constant taken as the value at $x = 1$ of the exact solution [19, 27]

$$\begin{aligned} C(x,t) = \frac{1}{2}\exp\left(\frac{Vx}{2D}\right) & \left[\exp\left(-\frac{\sqrt{V^2+4D\lambda Rx}}{2D}\right)\operatorname{erfc}\left(\frac{x-\sqrt{V^2+4D\lambda R}t/R}{\sqrt{4Dt/R}}\right)\right. \\ & + \left.\exp\left(\frac{\sqrt{V^2+4D\lambda Rx}}{2D}\right)\operatorname{erfc}\left(\frac{x+\sqrt{V^2+4D\lambda R}t/R}{\sqrt{4Dt/R}}\right)\right]. \end{aligned}$$

A typical adaptive mesh and the solution obtained thereon are shown in Fig. 1. The chemical reaction front is captured with the adaptive moving mesh. The results are in good agreement with the one dimensional ones obtained in [17].

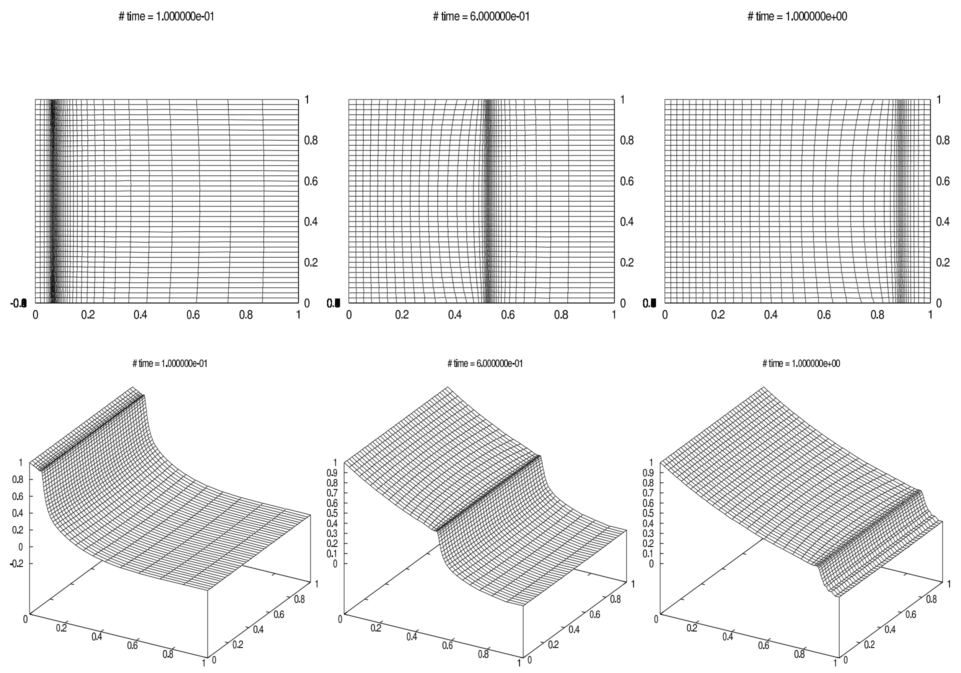

FIGURE 1. Example 3.1. Adaptive moving mesh (first row) and computed solution (second row) for ADRE.

Example 3.2: Solute transport from contamination sources in a heterogeneous aquifer. Solute transport from contamination sources in a heterogeneous aquifer is studied in this example. The governing equation is given by ADRE (2.1). The physical parameters are taken as $D(t,x,y) = D = 10^{-5}$, $R = 1$, $\lambda = 0$, and

$$V_1(t,x,y) = 1 - 0.1(1 + x^2 - y^2), \qquad V_2(t,x,y) = 0.$$

We first consider the case with a single source. The boundary conditions on the south, north, and east sides of the domain are flux free and the initial condition is zero everywhere. The source is placed on the west side of the domain. The adaptive mesh and the computed concentration of the contaminant are presented in Fig. 2. It can be seen that the concentration of the mesh nodes follows correctly with that of the contaminant, which is moving southeast due to the non-uniform Darcy velocity in the aquifer.

We now consider a case where three contamination sources, one on the west boundary and the other two inside the domain, are disposed in different time intervals. The results are shown in Fig. 3. The mesh adapts well to the concentration variations and automatically captures the emerging of new peaks. This example demonstrates that the adaptive moving mesh method provides a useful tool to detect unknown or unexpected pollution sources in water environment.

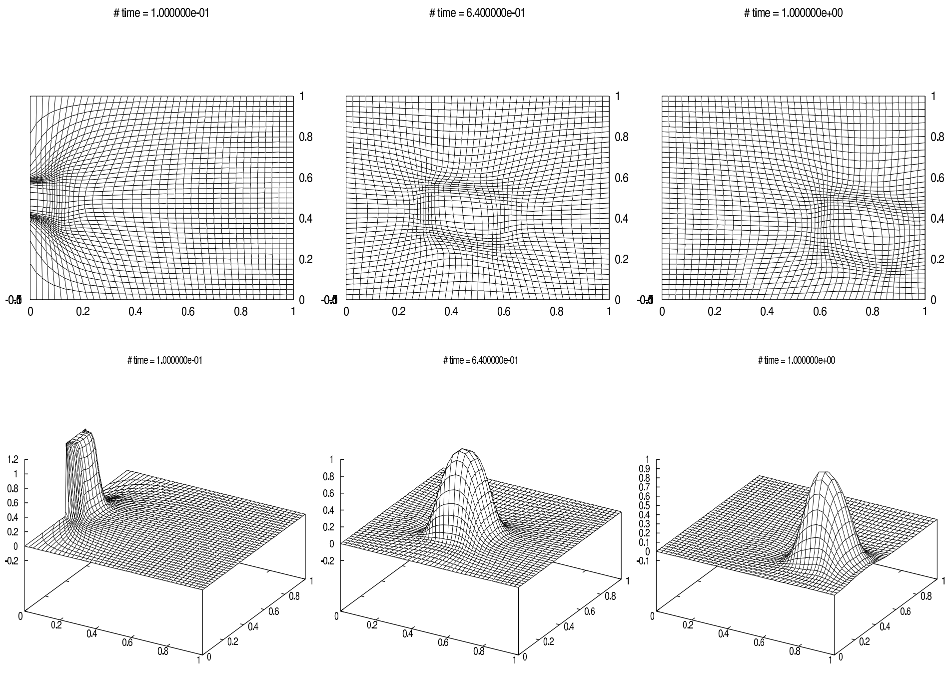

Figure 2. Example 3.2. Adaptive moving mesh (first row) and computed solution (second row) for ADRE with a single contamination source.

Example 3.3. Transport of nonaqueous phase liquids (NAPLs) in an aquifer. The system of governing equations becomes more complicated for

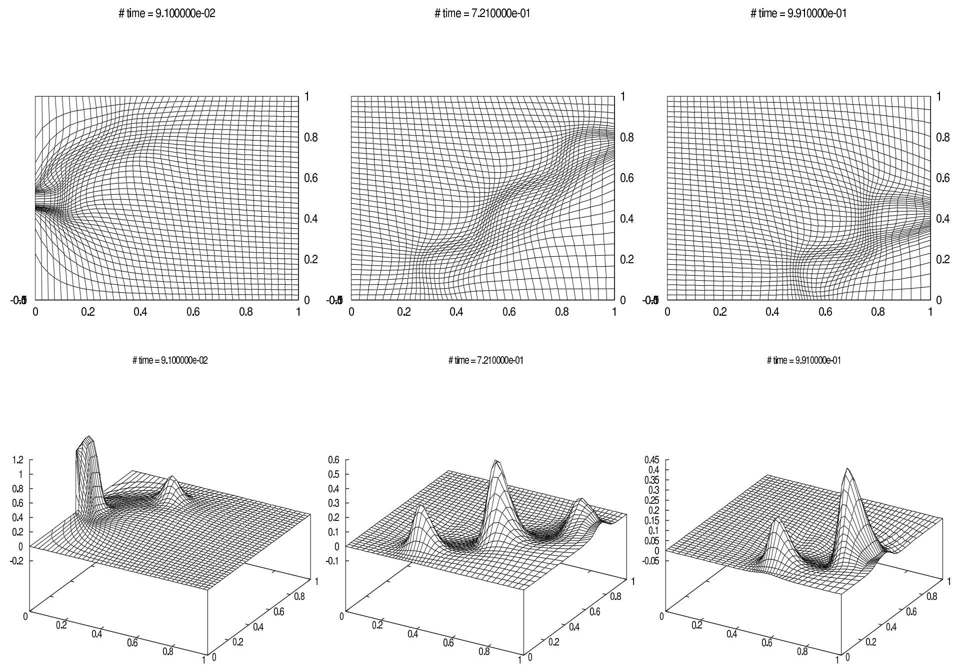

FIGURE 3. Example 3.2. Adaptive moving mesh (first row) and computed solution (second row) for ADRE with multiple contamination sources.

multiphase flow and transport in groundwater environment; e.g. see [21, 24] and references therein. Here we consider a specific case where nonaqueous phase liquids (NAPLs) are dissolved into the aqueous phase [22]. The physical process is described by two PDEs, one for the volumetric fraction of NAPL or NAPL content,

$$\frac{\partial \theta_n}{\partial t} = -\frac{k_{na}\ (C_a^* - C_a)}{\rho_n}, \tag{3.2}$$

and the other for the NAPL dissolved in water,

$$\frac{\partial(\theta_a C_a)}{\partial t} = \nabla \cdot (D\nabla C_a - q_a C_a) + k_{na}\ (C_a^* - C_a), \tag{3.3}$$

where the subscripts "a" and "n" represent the aqueous and nonaqueous phases, respectively, the superscript "$*$" indicates an equilibrium condition with the companion phase involved in the mass transfer, $\theta = \theta(t, x, y)$ is the volumetric fraction, $C_a = C_a(t, x, y)$ is the concentration of the NAPL dissolved in water, ρ is density, k_{na} is a mass transfer coefficient representing a mass transfer process referenced to a loss by the nonaqueous phase and a gain by the aqueous phase, q_a is water flux, and D is the dispersivity. It is noted that $\theta_n + \theta_a = n$, where n is the porosity considered to be constant here. The reader is referred to [22] for the derivations of the governing equations and the corresponding initial and boundary conditions. The values of the physical parameters are taken as those given in Table 15.1 of [22]. In particular, the water flux is taken as a constant vector in the x-direction. The parameter values are representative of conditions encountered in the two-dimensional physical experiements. The physical scenario of this case is that the aqueous phase is being flushed from the left boundary and the dissolved NAPL is being eluted from the right boundary. The left and right boundary conditions are a specified flux for the aqueous phase, while the top and bottom boundary conditions are no-flow. The initial residual NAPL saturation and other parameters are homogeneous, a typical laboratory condition, except that a perturbation in the residual NAPL saturation near the left boundary where a portion of the boundary is NAPL free, indicating that a clean water is flushing in. The development of this perturbation into a dissolution profile is then observed.

The obtained results are shown in Fig. 4 for mesh evolution, NAPL, and dissolved NAPL in water. It can be observed that a clean inflow from the west boundary washes out NAPL and reduces dissolved NAPL in water in a channel zone with time. The movements of the front and the boundary of the channel are captured correctly with the adaptive mesh. Once again the results show that the adaptive moving mesh method holds a great promise for solving multiphase flow and transport problems such as NAPL dissolution.

Example 3.4. Coupling of groundwater flow and NAPL transport. A more realistic modeling of the NAPL transport requires consideration of the coupling with the water flow. In this situation, in addition to equations (3.2) and (3.3), another equation is needed for the aqueous phase pressure $p_a = p_a(t, x, y)$, viz.,

$$0 = \nabla \cdot \left(\frac{k\, k_{ra}}{\mu} \left(\nabla p_a - \rho_a g \nabla x \right) \right) + \left(\frac{1}{\rho_a} - \frac{1}{\rho_n} \right) k_{na} (C_a^* - C_a), \tag{3.4}$$

where k is permeability, k_{ra} is relative permeability, μ is viscosity, g is the gravity constant, and x is the vertical coordinate. Unlike the previous example, the water flux in the equation (3.3) is now given by Darcy's law,

$$q_a = - \frac{k\, k_{ra}}{\mu} \left(\nabla p_a - \rho_a g \nabla x \right).$$

The values of other physical parameters are taken from Table 15.1 of [22]. Similar phenomena can be observed from the computational results (cf. Fig. 5). This seems reasonable since the water flux remains stable and has a similar profile as the one used in Example 3.3 as far as the water pressure gradient is maintained.

4. Summary and comments

The MMPDE (moving mesh PDE) moving mesh method developed in [14, 15] has been presented and applied to the numerical simulation of groundwater flow and transport problems. A range of problems have been considered, including advection dominated chemical transport and reaction, solute transport from contamination sources, transport of nonaqueous phase liquids (NAPLs) in an aquifer, and coupling of groundwater flow with NAPL transport. Numerical results have demonstrated that the MMPDE moving mesh method is able to capture sharp moving fronts and detect the emerging of new fronts. This consolidates the findings in a previous work [17] for one dimensional groundwater problems.

It should be pointed out that the investigations presented in this work are still at a preliminary stage and more work has yet to be done. For example, we have used a simple adaptation functional (2.8) and the arc-length monitor function (2.7) for mesh movement. For more precise control of mesh adaptation and for better

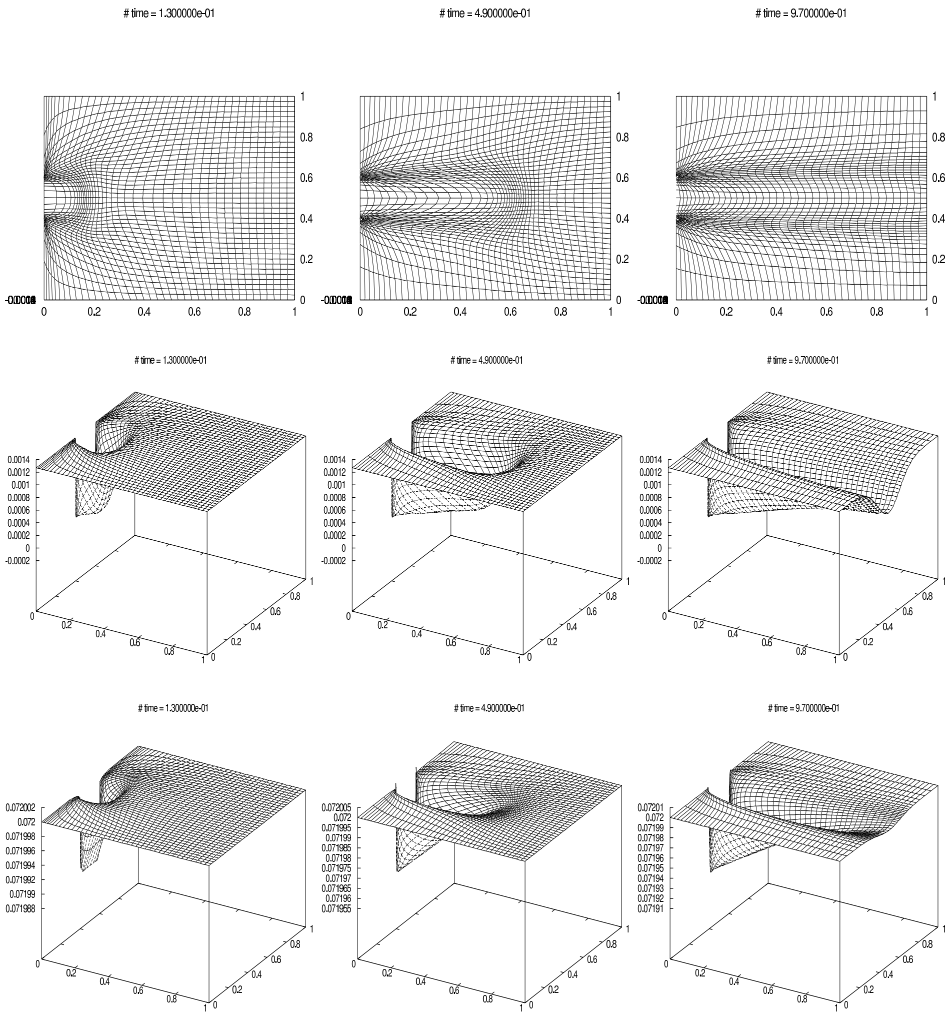

FIGURE 4. Example 3.3. Adaptive moving mesh (first row), NAPL (second row), and dissolved NAPL in water (third row) for the NAPL problem.

simulation accuracy, it would be better to use more sophisticated functionals and monitor functions such as those developed in [12, 16] based on interpolation error.

Acknowledgements

The work was partially supported by the National Science Foundation through grants DMS-0074240 and DMS-0410545 and by the Kansas Geological Survey. The simulations were performed with a SGI-Origin 2000 multi-processor computer in the Kansas Center for Advanced Scientific Computing, The University of Kansas. It is acknowledged that L. Zheng participated in this investigation at the early stage.

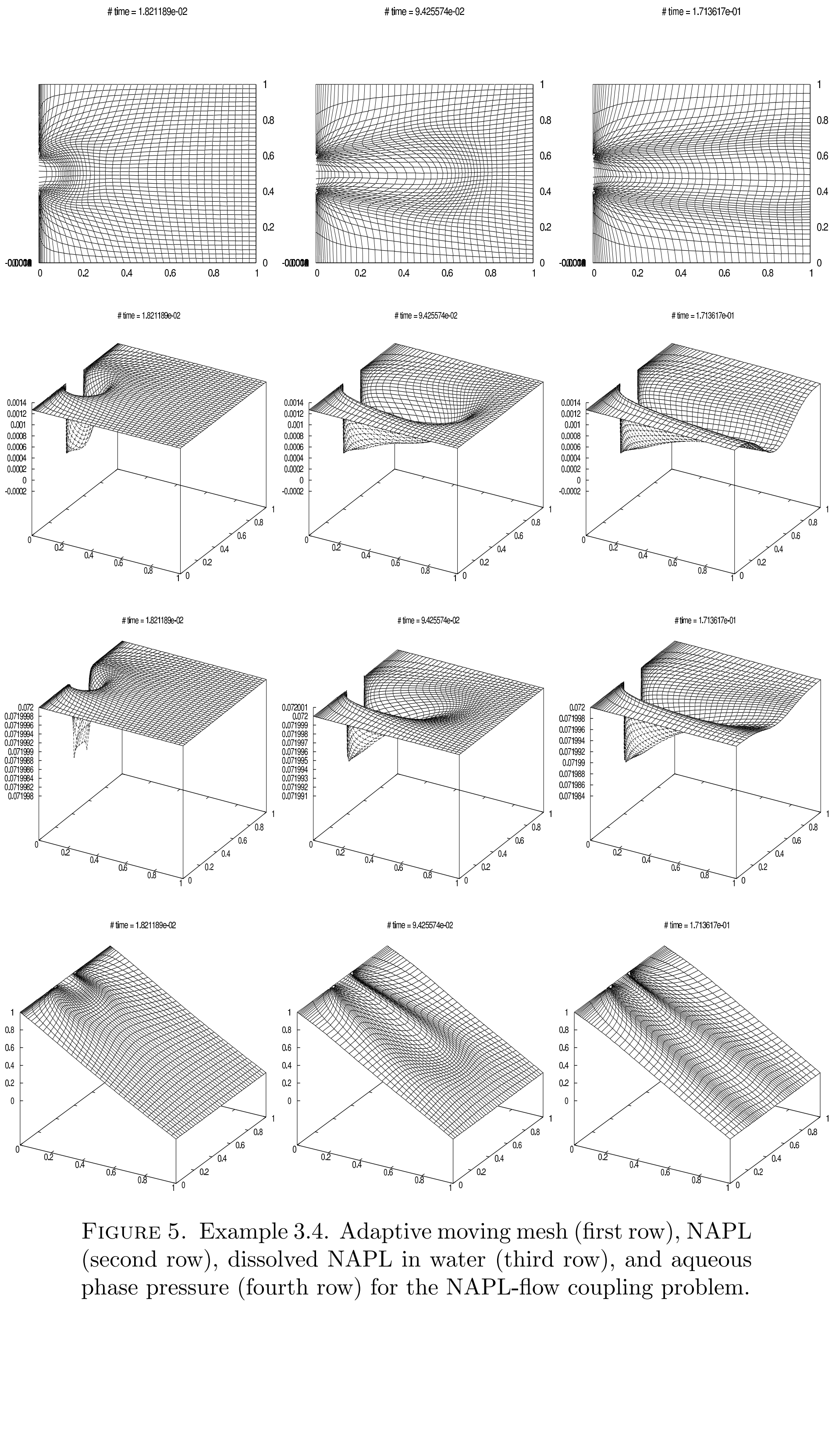

FIGURE 5. Example 3.4. Adaptive moving mesh (first row), NAPL (second row), dissolved NAPL in water (third row), and aqueous phase pressure (fourth row) for the NAPL-flow coupling problem.

References

1. J. U. Brackbill, *An adaptive grid with directional control*, J. Comput. Phys. **108** (1993), 38 – 50.
2. J. U. Brackbill and J. S. Saltzman, *Adaptive zoning for singular problems in two dimensions*, J. Comput. Phys. **46** (1982), 342 – 368.
3. W. Cao, W. Huang, and R. D. Russell, *An r-adaptive finite element method based upon moving mesh pdes*, J. Comp. Phys. **149** (1999), 221 – 244.
4. ______, *Approaches for generating moving adaptive meshes: location versus velocity*, Appl. Numer. Math. **47** (2003), 121 – 138.
5. J. R. Cash, *Diagonally implicit runge-kutta formulae with error estimates*, J. Inst. Math. Appl. **24** (1979), 293 – 301.
6. C. de Boor, *Good approximation by splines with variable knots*, Spline Functions and Approximation Theory (A. Meir and A. Sharma, eds.), Birkhäuser Verlag, Basel und Stuttgart, 1973, pp. 57 – 73.
7. A. S. Dvinsky, *Adaptive grid generation from harmonic maps on riemannian manifolds*, J. Comput. Phys. **95** (1991), 450 – 476.
8. A. Gamliel and L. M. Abriola, *A one-dimensional moving grid solution for the coupled nonlinear equations governing multi-phase flow in porous media. 1: Model development*, Int. J. Numer. Meth. Fluids **14** (1992), 25 – 45.
9. ______, *A one-dimensional moving grid solution for the coupled nonlinear equations governing multi-phase flow in porous media. 2: Example simulations and sensitivity analysis*, Int. J. Numer. Meth. Fluids **14** (1992), 47 – 69.
10. G. Gottardi and M. Venutelli, *Moving finite element model for one-dimensional infiltration in unsaturated soil*, Water Resour. Res. **28** (1992), 3259 – 3267.
11. D. F. Hawken, J. J. Gottlieb, and J. S. Hansen, *Review of some adaptive node-movement techniques in finite element and finite difference solutions of PDEs*, J. Comput. Phys. **95** (1991), 254 – 302.
12. W. Huang, *Measuring mesh qualities and application to variational mesh adaptation*, to appear in SIAM J. Sci. Comput.
13. ______, *Practical aspects of formulation and solution of moving mesh partial differential equations*, J. Comput. Phys. **171** (2001), 753 – 775.
14. W. Huang, Y. Ren, and R. D. Russell, *Moving mesh partial differential equations (MMPDEs) based upon the equidistribution principle*, SIAM J. Numer. Anal. **31** (1994), 709 – 730.
15. W. Huang and R. D. Russell, *A high dimensional moving mesh strategy*, Appl. Numer. Math. **26** (1997), 63 – 76.
16. W. Huang and W. Sun, *Variational mesh adaptation II: error estimates and monitor functions*, J. Comput. Phys. **184** (2003), 619 – 648.
17. W. Huang, L. Zheng, and X. Zhan, *Adaptive moving mesh methods for simulating one-dimensional groundwater problems with sharp moving fronts*, Int. J. Numer. Meth. Engng. **54** (2002), 1579 – 1603.
18. P. M. Knupp, *Jacobian-weighted elliptic grid generation*, SIAM J. Sci. Comput. **17** (1996), 1475 – 1490.
19. T. Lee, *Applied mathematics in hydrology*, Lewis Publishers, 1999.
20. R. Li, T. Tang, and P. W. Zhang, *Moving mesh methods in multiple dimensions based on harmonic maps*, J. Comput. Phys. **170** (2001), 562 – 588.
21. C. T. Miller, G. Christakos, P. T. Imhoff, J. F. McBride, J. A. Pedit, and J. A. Trangenstein, *Multiphase flow and transport modeling in heterogeneous porous media: Challenges and approaches*, Adv. Water Resour. **21** (1998), 77 – 120.
22. C. T. Miller, S. N. Gleyzer, and P. T. Imhoff, *Numerical modeling of napl dissolution fingering in porous media*, Physical nonequilibrium in soils modeling and application (H. M. Selim and L. Ma, eds.), Ann Arbor Press, Chelsea, Michigan, 1998.
23. K. Miller and R. N. Miller, *Moving finite elements I*, SIAM J. Numer. Anal. **18** (1981), 1019 – 1032.
24. T. F. Russell, *Modeling of multiphase multicontaminant transport in the subsurface*, Rev. Geophys. **33** (1995), 1035 – 1047, Supplement 2.
25. D. R. Shier and K. T. Wallenius, *Applied mathematical modeling: A multidisciplinary approach*, Chapman & Hall/CRC, London, Boca Raton, 2000.

26. R. Trompert, *Local-uniform grid refinement and transport in heterogeneous porous media*, Adv. Water Resour. **16** (1993), 293 – 304.
27. E. J. Wexler, *Analytical solutions for one-, two-, and three-dimensional solute transport in ground-water systems with uniform flow*, U.S. Geological Survey, Book 3, Techniques of Water-Resources Investigations, 1992, p. 190.
28. A. M. Winslow, *Adaptive mesh zoning by the equipotential method*, Tech. Report UCID-19062, Lawrence Livemore Laboratory, 1981.
29. G. T. Yeh, J. Chang, and T. E. Short, *An exact peak capturing and oscillation-free scheme to solve advection-dispersion transport equations*, Water Resour. Res. **28** (1992), 2937 – 2951.
30. P. A. Zegeling, J. G. Verwer, and J. C. H. Van Eijkeren, *Application of a moving grid method to a class of 1D brine transport problems in porous media*, Int. J. Numer. Meth. Fluids **15** (1992), 175 – 191.
31. C. Zheng and G. D. Bennett, *Applied contaminant transport modeling: Theory and practice*, Van Nostrand Reinhold, New York, 1995.
32. O. C. Zienkiewicz and J. Z. Zhu, *The superconvergence patch recovery and a posteriori error estimates. part 1: The recovery technique*, Int. J. Numer. Methods Engrg. **33** (1992), 1331 – 1364.
33. ______, *The superconvergence patch recovery and a posteriori error estimates. part 2: Error esimates and adaptivity*, Int. J. Numer. Methods Engrg. **33** (1992), 1365 – 1382.

Department of Mathematics, The University of Kansas, Lawrence, KS 66045, USA
E-mail address: `huang@math.ku.edu`

Kansas Geological Survey, The University of Kansas, Lawrence, KS 66047, USA
E-mail address: `xyz@ku.edu`