XVIII International Conference on Water Resources
CMWR 2010
J. Carrera (Ed)
©CIMNE, Barcelona, 2010

NUMERICAL MODELLING OF DYNAMIC GROUNDWATER TABLE USING LEVEL SET FORMULATION

Peter Frolkovič* and Petra Zacharovská*

*Slovak University of Technology
Faculty of Engineering, Radlinskeho 11, 81368 Bratislava, Slovakia
e-mail: peter.frolkovic@gmail.com, web page: http://www.math.sk.

Key words: dynamic groundwater table, level set method

Summary. We study a representative mathematical model of groundwater flow where a dynamic position of the water table is a part of unknown solution. To compute the problem on a fixed (enlarged) domain we describe the groundwater table using a level set formulation. A novel discretization method is proposed to solve the problem on a fixed grid. Numerical results confirm the applicability of the method for this type of problems.

1 INTRODUCTION

Level set methods are very popular mathematical tool for the solution of problems with moving boundaries and interfaces [9, 8, 4, 5], especially for the numerical solution of two-phase flows [3]. The idea is to describe the free boundary implicitly as a zero set of some level set function. The advantage of such formulation is a possibility to use fixed computational grids without moving grid points.

In this work we propose a level set method for the numerical simulation of groundwater flow with a free water table. In section 1 we introduce the representative mathematical model. In section 2 we propose the numerical method and in section 3 we present numerical experiments.

2 MATHEMATICAL MODEL

Let $D \subset \mathcal{R}^2$ be a unit square. Let $\Gamma(t) \subset D$ be a curve that describes in time $t \geq 0$ the dynamic position of groundwater table and $\Omega(t) \subset D$ be the subdomain "bellow" $\Gamma(t)$. Finally, $\Omega^{out}(t) := D \setminus \Omega(t)$.

The groundwater flow is considered only in $\Omega(t)$, $t \geq 0$ and is characterized by an unknown pressure $p = p(x, z, t)$ that obeys the partial differential equation

$$\nabla \cdot \vec{q} = 0\,, \quad \vec{q} = -K \nabla \left(p + \rho g z\right)\,. \tag{1}$$

All parameters in (1) are constant. The dependence of p on t is only due to the dynamic position of $\Gamma(t)$.

Furthermore, we suppose that $\partial\Omega(t) = \overline{\Gamma(t) \cup \Gamma^D \cup \Gamma^N}$ and

$$p(x,z,t) = 0\,,\ (x,z) \in \Gamma(t)\,, \tag{2}$$

$$p(x,z,t) = p^D(x,z)\,,\ (x,z) \in \Gamma^D\,, \tag{3}$$

$$\vec{n}(x,z) \cdot \nabla p(x,z,t) = p^N(x,z)\,,\ (x,z) \in \Gamma^N\,, \tag{4}$$

where $\vec{n}$ is the normal vector with respect to Γ^N.

Finally, we suppose that the movement of $\Gamma(t)$ is prescribed by the speed $\bar{f} = \bar{f}(x,z,t)$, $(x,z) \in \Gamma(t)$ that is defined in normal direction $\vec{N} = \vec{N}(x,z,t)$ with respect to $\Gamma(t)$ (pointing from $\Omega(t)$ to outside) and that is equal to

$$\bar{f} = \frac{1}{\theta}\vec{N} \cdot (\vec{q} - A\vec{e}_z)\,. \tag{5}$$

Following [1], θ is a given effective porosity and $A = A(x,z,t)$, $(x,z) \in \Gamma(t)$ is a given velocity of accretion.

The system (1) - (5) constitutes our mathematical model to introduce a representative example of groundwater flow with dynamic water table.

Next we continue to describe the dynamic position $\Gamma(t)$ of groundwater table using a level set formulation. Let the initial position $\Gamma(0)$ be given implicitly as the zero level set of some function $\varphi^0 = \varphi^0(x,z)$, i.e., $\Gamma(0) = \{(x,z) \in D\,,\ \varphi^0(x,z) = 0\}$. Moreover, let $\Omega(0)$ be given by $\Omega(0) = \{(x,z) \in D\,,\ \varphi^0(x,z) < 0\}$.

An important (nontrivial) step of level set formulation is to find a (smooth) velocity function $\vec{V} = \vec{V}(x,z,t)$ such that $\vec{V} = \bar{f}\vec{N}$ for $(x,z) \in \Gamma(t)$, see later. Once $\vec{V}$ is given, we can search for the solution $\varphi = \varphi(x,z,t)$, $(x,z) \in D$, $t > 0$ of advection equation

$$\partial_t \varphi + \vec{V} \cdot \nabla\varphi = 0\,, \quad \varphi(x,z,0) = \varphi^0(x,z)\,, \tag{6}$$

that describes implicitly the time dependant position of the interface, i.e., $\Gamma(t) = \{(x,z) \in D\,,\ \varphi(x,z,t) = 0\}$ and $\Omega(t) = \{(x,z) \in D\,,\ \varphi(x,z,t) < 0\}$. Some standard, e.g., Dirichlet or outflow boundary conditions, can be considered with (6).

Before introducing our choice of $\vec{V}$ in (6), we need to define for a fixed t the so called signed distance function $\phi(x,z,t)$ for the interface $\Gamma(t)$ that is a (weak) solution of the so called eikonal equation [9, 8],

$$|\nabla\phi(x,z,t)| = 1\,, \quad (x,z) \in D\,, \quad \phi(x,z,t) = 0\,, \quad (x,z) \in \Gamma(t)\,. \tag{7}$$

To find ϕ, we search for the stationary solution $\Phi = \Phi(x,z,s)$ of two equations

$$\partial_s\Phi(x,z,s) + |\nabla\Phi(x,z,s)| = 1\,, \quad (x,z) \in \Omega^{out}(t)\,,\ s > 0\,, \tag{8}$$

$$\partial_s\Phi(x,z,s) - |\nabla\Phi(x,z,s)| = -1\,, \quad (x,z) \in \Omega(t)\,,\ s > 0\,. \tag{9}$$

The initial condition for (8) - (9) are given by $\Phi(x,z,0) = \phi(x,z,t)$, $(x,z) \in D$, and the boundary conditions for $(x,z) \in \Gamma(t)$, $s > 0$ by $\Phi(x,z,s) = 0$. The treatment of boundary conditions on ∂D can be found, e.g., in [6].

Let the stationary solution of (8) - (9) be reached for some finite time $S > 0$, then $\phi(x,z,t) := \Phi(x,z,S)$. Note that $\Gamma(t)$ is the zero level set of $\phi(x,z,t)$ and $\Phi(x,z,s)$.

Having $\phi(x,z,t)$ for a fixed t, we search for $f = f(x,z,t)$ such that $f(x,z,t) = \bar{f}(x,z,t)$, $(x,z) \in \Gamma(t)$, and

$$\nabla\phi \cdot \nabla f = 0\,. \tag{10}$$

To find more easily the function f for a fixed t, we can again search for the stationary solution $F = F(x,z,s)$ of following advection equations

$$\partial_s F + \nabla\phi \cdot \nabla F = 0\,, \tag{11}$$

that are solved for $s > 0$ independently in two subdomains $\Omega(t)$ and $\Omega^{out}(t)$. The boundary condition on $\Gamma(t) = \partial\Omega(t) \cap \partial\Omega^{out}(t)$ is given by $F(x,z,s) = \bar{f}(x,z,t)$. Again, such stationary solution is obtained at some finite time $S > 0$ and $f(x,z,t) = F(x,z,S)$.

Once the function f is found, the velocity $\vec{V}$ used in (6) is defined by $\vec{V} = f\nabla\phi$.

3 DISCRETIZATION METHOD

We describe our discretization method using standard notation of finite differences. To do so, let us discretize D by a grid made of points (x_i, z_j), $0 \le i,j \le I$, where I is given and $h := x_{i+1} - x_i = z_{j+1} - z_j$.

Let $\varphi^0_{ij} := \varphi^0(x_i, z_j)$. The values φ^n_{ij} will approximate $\varphi(x_i, z_j, t^n)$ for some discrete time points $0 = t^0 < t^1 < \ldots < t^n < \ldots$ and will be determined in our algorithm.

To find a polygonal approximation Γ^n_h of the interface $\Gamma(t^n)$, we will assume a linear interpolation between φ^n_{ij} and its (at most four) neighbouring values.

Throughout this paper we say that $(k,l) \in \Lambda^n_{ij}$, if $\varphi^n_{ij} < 0$, and (k,l) is the index of one of existing neighbours $\varphi^n_{i\pm1j}$ or $\varphi^n_{ij\pm1}$, and, moreover, $\varphi^n_{ij}\varphi^n_{kl} < 0$. Clearly, if $(k,l) \in \Lambda^n_{ij}$, there exists a zero point of the linear interpolation on the edge between (x_i, z_j) and (x_k, z_l). To determine such point, one can find $\bar{\alpha} \in (0,1)$ such that

$$0 = \bar{\alpha}\varphi^n_{ij} + (1-\bar{\alpha})\varphi^n_{kl} \;\Rightarrow\; \bar{\alpha} = \frac{\varphi^n_{kl}}{\varphi^n_{kl} - \varphi^n_{ij}}\,, \tag{12}$$

and the zero point $(\bar{x}, \bar{z})$ of linear interpolation between φ^n_{ij} and φ^n_{kl} is given by

$$(\bar{x}, \bar{z}) = \frac{\varphi^n_{kl}}{\varphi^n_{kl} - \varphi^n_{ij}}(x_i, z_j) + \frac{\varphi^n_{ij}}{\varphi^n_{ij} - \varphi^n_{kl}}(x_k, z_l)\,. \tag{13}$$

Therefore, Γ^n_h can be represented by a polygonal that connects all such zero points. Analogously, we can define $\Omega^n_h \approx \Omega(t^n)$. Due to our assumptions we have that $(x_i, z_j) \in \Omega^n_h$ if $\varphi^n_{ij} < 0$. If $\varphi^0_{ij} = 0$, one has $(x_i, z_j) \in \Gamma^n_h$, but in general Γ^n_h does not cross the grid points.

3.1 NUMERICAL SOLUTION OF GROUNDWATER FLOW

Let the values φ_{ij}^n be know. We discretize (1) by standard finite differences

$$4p_{ij}^n - \hat{p}_{i+1j}^n - \hat{p}_{i-1j}^n - \hat{p}_{ij+1}^n - p_{ij-1}^n = 0 \,. \tag{14}$$

The discrete equations (14) are constructed only for grid points $(x_i, z_j) \in \Omega_h^n \cup \Gamma^N$. For the values $\hat{p}_{kl}^n$ in (14) with corresponding indices one has $\hat{p}_{kl}^n = p_{kl}^n$ if $(k,l) \notin \Lambda_{ij}^n$. Standard treatment of Neumann and Dirichlet boundary conditions shall be used, including the case $p_{kl}^n = 0$ if $\varphi_{kl}^n = 0$.

To define $\hat{p}_{kl}^n$, $(k,l) \in \Lambda_{ij}^n$, we extrapolate linearly the non-existing discrete value of p in the grid point (x_k, z_l) using (12) and (13). By exploiting that $p(\bar{x}, \bar{z}) = 0$, we obtain

$$\hat{p}_{kl}^n = \frac{\varphi_{kl}^n}{\varphi_{ij}^n} p_{ij}^n \,, \quad (k,l) \in \Lambda_{ij}^n \,. \tag{15}$$

A caution is necessary for very small values of φ_{ij}^n, see [7].

To determine the values $p_{ij}^n \approx p(x_i, z_j, t^n)$ for the grid nodes $(x_i, z_j) \in \Omega_h^n \cup \Gamma^N$, one has to solve a linear system of algebraic equations. When done, the values $\vec{q}_{ij}^n \approx \vec{q}(x_i, z_j, t^n)$ can be computed for $(x_i, z_j) \in \Omega_h^n$ by

$$\vec{q}_{ij}^n = \frac{1}{2h} \left(\hat{p}_{i+1j}^n - \hat{p}_{i-1j}^n \,, \hat{p}_{ij+1}^n - p_{ij-1}^n \right) \,. \tag{16}$$

To proceed with (14) from n to $n+1$, we need to compute the values φ_{ij}^{n+1} using some approximation of advection equation (6). In next section we describe how to obtain the discrete values $\vec{V}_{ij}^n \approx \vec{V}(x_i, z_j, t^n)$. Once such values are available, we use the standard first order accurate upwind method (explicit in time, see, e.g., [4]) to compute φ_{ij}^{n+1}.

3.2 NUMERICAL SOLUTION OF EIKONAL EQUATION

Let the index n be fixed. To discretize (8) and (9), we follow [2] and introduce a numerical scheme valid for $0 \le i, j \le I$ and $m = 0, 1, \ldots$

$$\Phi_{ij}^{m+1} = \Phi_{ij}^m \pm \Delta s^m \left(1 - \frac{1}{h} \sqrt{(\Delta_x \Phi_{ij}^m)^2 + (\Delta_z \Phi_{ij}^m)^2} \right) \,, \tag{17}$$

where a particular sign of $\pm$ has to be chosen analogously to (8) or (9). Furthermore,

$$\Delta_x \Phi_{ij}^m = \max\{|M_{i+1\,j}|, |M_{i-1\,j}|\} \,, \quad \Delta_z \Phi_{ij}^m = \max\{|M_{i\,j+1}|, |M_{i\,j-1}|\} \,, \tag{18}$$

and

$$M_{k\,l} = \begin{cases} \min\{\hat{\Phi}_{k\,l}^m - \Phi_{ij}^m, 0\}, & \varphi_{ij}^n > 0 \,, \\ \max\{\hat{\Phi}_{k\,l}^m - \Phi_{ij}^m, 0\}, & \varphi_{ij}^n < 0 \,. \end{cases} \tag{19}$$

The definition (18) is valid without changes if $0 < i, j < I$, otherwise the values of M with the indices of non-existent grid points are simply skipped in (18).

Analogously to previous section, one chooses in (19) that $\hat{\Phi}^m_{kl} = \Phi^m_{kl}$ if $\varphi^n_{ij}\varphi^n_{kl} \geq 0$ and $\hat{\Phi}^m_{kl} = \frac{\varphi^n_{kl}}{\varphi^n_{ij}}\Phi^m_{ij}$ if $\varphi^n_{ij}\varphi^n_{kl} < 0$, see also the related discussion in section 3.1.

The time step in (17) is chosen typically to be $\Delta s^m = h/2$ that insures a stability in the case of uniform grids [2]. Unfortunately, this stability can be disturbed for the grid points (x_i, z_j) near Γ^n_h.

For such grid points we slightly modify the scheme (17) to allow larger time steps. We are inspired by similar approach used in [4] and by exploiting a close relation between the signed distance function and the so called first arrival time function [9, 6].

By simplifying the topic and without going into much details, one can consider for the grid points (x_i, z_j) near to Γ^n_h the following numerical scheme

$$\tilde{\Phi}^{m+1}_{ij} = \Phi^m_{ij} \mp \Delta s^m \frac{1}{h}\sqrt{(\Delta_x \Phi^m_{ij})^2 + (\Delta_z \Phi^m_{ij})^2}\,, \tag{20}$$

that can be seen as analogous to (17) if applied to equation $\partial_s \Phi = \mp|\nabla\Phi|$. Using (20), one can define a special time step $\Delta^{crit} s^m_{ij}$ such that $\tilde{\Phi}^{m+1}_{ij} = 0$ if $\Delta s^m = \Delta^{crit} s^m_{ij}$ in (20), i.e.,

$$\Delta^{crit} s^m_{ij} = \frac{|\Phi^m_{ij}| h}{\sqrt{(\Delta_x \Phi^m_{ij})^2 + (\Delta_z \Phi^m_{ij})^2}} \tag{21}$$

that can be viewed as an approximation of the first arrival time function at (x_i, z_j).

Applying these ideas in the context of (17), we replace (17) for the grid points (x_i, z_j) near Γ^n_h by the scheme

$$\Phi^{m+1}_{ij} = \Phi^m_{ij} \pm \Delta s^m_{ij}\left(1 - \frac{1}{h}\sqrt{(\Delta_x \Phi^m_{ij})^2 + (\Delta_z \Phi^m_{ij})^2}\right)\,, \tag{22}$$

where $\Delta s^m_{ij} = \min\{\Delta s^m, \Delta^{crit} s^m_{ij}\}$. In such a way, the modified scheme (22) has no stability restriction on the choice of Δs^m.

In theory, one has to apply (17) with so many time steps m, until a steady state is reached. In practice, only some fixed number of time steps might be used, say $m = M$. Once finished, $\phi^n_{ij} = \Phi^M_{ij}$.

It is important to note that although φ and ϕ has an identical zero level set, this is not necessary the case for their approximations. Therefore, the approximation of φ is used only to represent implicitly Γ^n_h and Ω^n_h, see section 3, and the approximation of ϕ is used to approximate $\nabla\phi$, see the following section.

3.3 NUMERICAL SOLUTION OF VELOCITY EXTRAPOLATION

To approximate $\bar{f}$ in (5), for $(x_i, z_j) \in \Omega^n_h$ such that $\Lambda^n_{ij} \neq \emptyset$ we define

$$\bar{f}^n_{ij} := \frac{1}{\theta_{ij}} \frac{\nabla\phi^n_{ij}}{|\nabla\phi^n_{ij}|} \cdot \left(\vec{q}^n_{ij} - A^n_{ij}\vec{e}_z\right) \tag{23}$$

where

$$\nabla \phi_{ij}^n = \left(\frac{\phi_{i+1j}^n - \phi_{i-1j}^n}{2h}, \frac{\phi_{ij+1}^n - \phi_{ij-1}^n}{2h} \right) . \tag{24}$$

To compute the approximative values F_{ij}^m for the solution F of advection equation (11), we apply again the standard upwind method.

In the exact case of (11), F is known and fixed on $\Gamma(t)$. In our numerical approximation, we use a simple approximation by fixing the values of $F_{ij}^m = \bar{f}_{ij}^n$ for $(x_i, z_j) \in \Omega_h^n$ such that $\Lambda_{ij}^n \neq \emptyset$. We are aware of the fact that such an approximation is rather rough, see our numerical experiments, and in future we plan to improve this part of the algorithm.

Concerning the initial condition, for $n = 0$ no straightforward definition of F_{ij}^0 is possible, and it can be chosen rather arbitrary. One has then to insure that enough time steps are computed, say $m = M$, to obtain F_{ij}^M being in steady state. For $n > 0$, similarly to section 3.2, one can compute F_{ij}^m only for a fixed number of time steps, say $m = M$, using $F_{ij}^0 = f_{ij}^{n-1}$. Once done, $f_{ij}^n = F_{ij}^M$.

4 NUMERICAL EXPERIMENTS

To test our algorithm, we compute two examples where the exact solution is known. In all examples, Dirichlet boundary conditions are chosen for (1) always on the left and right side of D and the Neumann boundary condition on the bottom of D such that specified stationary pressure fulfils them exactly. We choose a rather coarse mesh with $h = 0.125$ to illustrate visibly the numerical approximations used in our algorithm. The time step is chosen $\Delta s = 0.05$, the time interval is $(0, 0.3)$, and $K = \rho = 1$.

Firstly, we test if a straight horizontal groundwater table is reached when its initial position is disturbed. To do so, $g = 1$, $\varphi^0 = \Psi$, and

$$\Psi(x,z) = r(0, 0.75) - r(x,z) \, , \quad r(x,z) := \sqrt{(x-0.5)^2 + (z-1.5)^2} \, . \tag{25}$$

The numerical results for $n = 0$ can be seen in Figure 1. The numerical solution approximates the steady state at $t = 0.3$ very well, but we do not present the results here.

The second example is proposed in such a way that the stationary solution is given by $P(x,z) = \ln\left(r(0,0.75)^{-1}\right) - \ln\left(r(x,z)^{-1}\right)$ if no gravity is present, i.e., $g = 0$. The distance function to the zero level set of P is given by Ψ in (25). The speed $A(x,z)$ in (5) is chosen such that $\bar{f}(x,z) = 0$ in (5) for $(x,z) \in D$ and $\Psi(x,z) = 0$, and $\theta = 1$, i.e.,

$$A = -\frac{1}{\partial_z \Psi} \left(\nabla P \cdot \nabla \Psi \right) . \tag{26}$$

The exact pressure P and the velocity $\vec{q} - A\vec{e}_z$ can be seen on the left picture in Figure 2.

We start the simulation with the horizontal groundwater table, i.e., $\varphi^0(x,z) = z - 0.75$. The corresponding numerical results are given in Figure 2.

For an illustration of other properties of the method (including its convergence), the numerical steady state results for the grid 12×12 at $t = 0.3$ are presented in Figure 3.

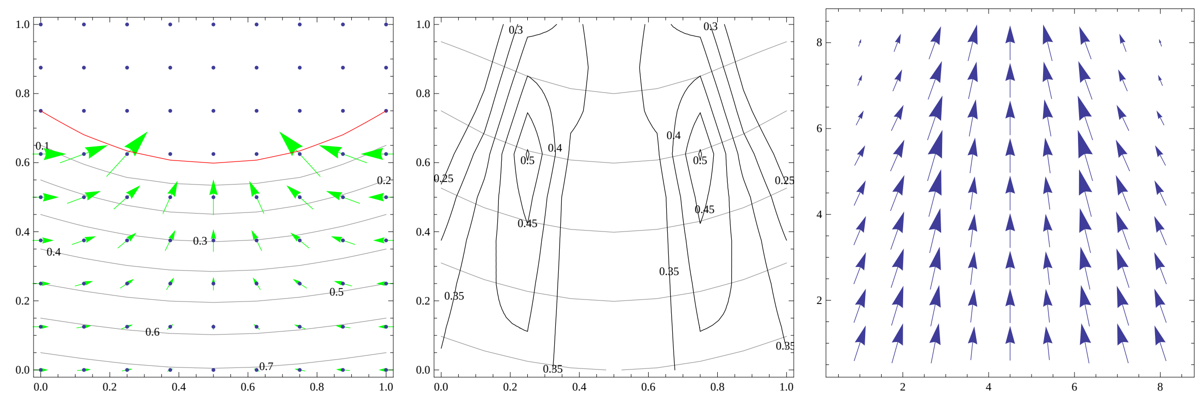

Figure 1: Numerical solution of Example 1, the contours of pressure p^0, the groundwater velocity $\vec{q}^0$, and the grid points (left picture), the contours of level set function ϕ^0 and of extrapolated velocity f^0 (middle picture), and the advection velocity $\vec{V}^0$ (right picture).

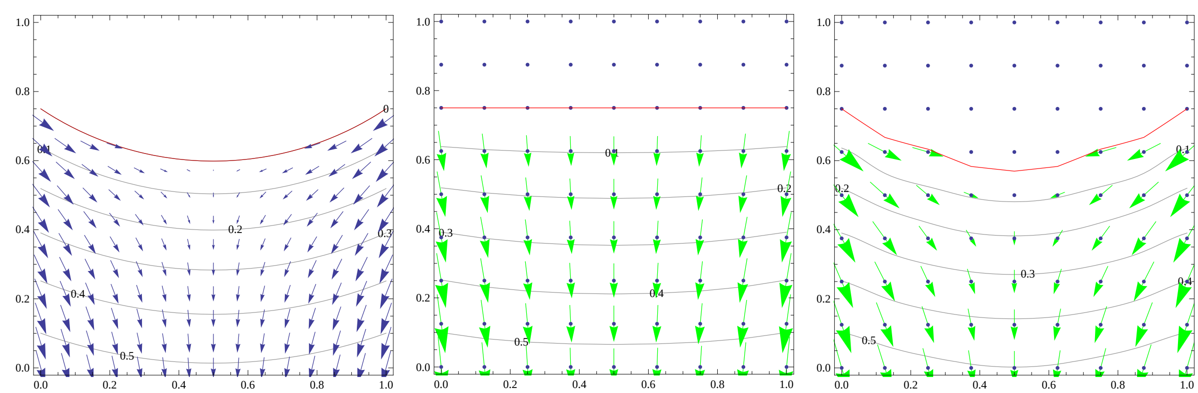

Figure 2: Example 2, the exact stationary pressure P and the velocity $\vec{q} - A\vec{e}_z$ (left picture), the numerical initial pressure p^0 and the velocity $\vec{q}^0 - A\vec{e}_z$ at $t = 0$ (middle picture) and at $t = 0.3$ (right picture).

5 CONCLUSIONS

The first author was supported by the grants APVV-0351-07, VEGA 1/0269/09, the LOEWE-Program of HIC4FAIR and the E-DuR project of BMWi in Germany.

REFERENCES

[1] J. Bear. *Hydraulics of Groundwater.* McGraw-Hill, New York, 1979.

[2] P. Bourgine, P. Frolkovič, K. Mikula, and M.Remešíková N. Peyriéras. Extraction of the intercellular skeleton from 2d microscope images of early embryogenesis. In *Lecture Notes in Computer Science 5567*, pages 38–49, 2009.

[3] P. Frolkovič, D. Logashenko, and G. Wittum. Flux-based level set method for two-

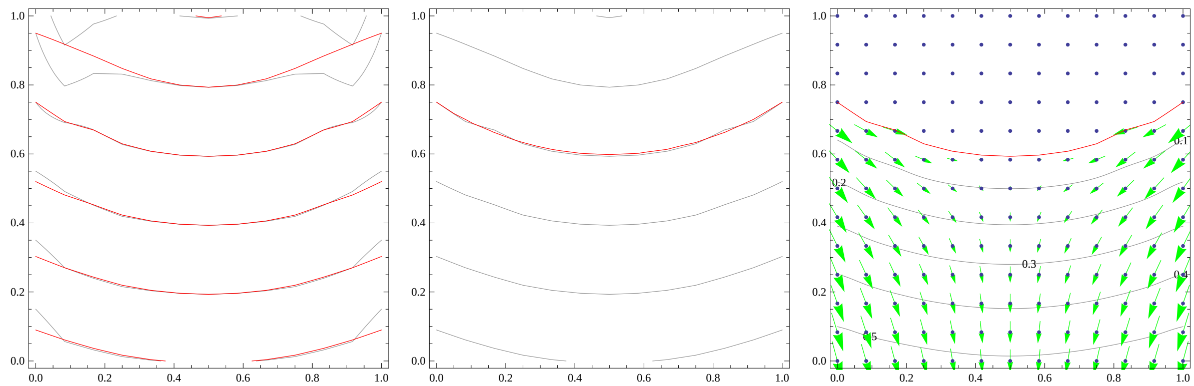

Figure 3: Numerical solution of Example 2, the distance function ϕ (red contours) and the level set function φ (grey contours) (left picture), the exact groundwater table (red) and ϕ (middle picture), and the pressure and the velocity $\vec{q} - A\vec{e}_z$ at $t = 0.3$ (right picture).

phase flows. In R. Eymard and J.-M. Herard, editors, *Finite Volumes for Complex Applications*, pages 415–422. ISTE and Wiley, 2008.

[4] P. Frolkovič and K. Mikula. Flux-based level set method: A finite volume method for evolving interfaces. *Applied Numerical Mathematics*, 57(4):436–454, 2007.

[5] P. Frolkovič and K. Mikula. High-resolution flux-based level set method. *SIAM J. Sci. Comp.*, 29(2):579–597, 2007.

[6] P. Frolkovič and C. Wehner. Flux-based level set method on rectangular grids and computations of first arrival time functions. *Computing and Visualization in Science*, 12(5):297–306, 2009.

[7] F. Gibou, R. Fedkiw, L.-T. Cheng, and M. Kang. A second order accurate symetric discretization of the Poisson equation on irregular domains. *J. Comput. Phys.*, 176:205–227, 2002.

[8] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003.

[9] J. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.