

NUMERICAL ALGORITHMS FOR SOLVING PROBLEMS OF MULTIPHASE FLOWS IN POROUS MEDIA

R. ČIEGIS¹, O. ILIEV², V. STARIKOVIČIUS¹ and K. STEINER²

¹*Vilnius Gediminas Technical University*

Saulėtekio al. 11, LT-10223, Vilnius, Lithuania

E-mail: rc@fm.vtu.lt; vs@sc.vtu.lt

²*Fraunhofer ITWM*

Fraunhofer-Platz 1, D-67663 Kaiserslautern, Germany

E-mail: iliev@itwm.fhg.de

Received March 4, 2006; revised April 25, 2006; published online May 25, 2006

Abstract. In this paper we discuss numerical algorithms for solving the system of nonlinear PDEs, arising in modelling of two-phase flows in porous media, as well as the proper object oriented implementation of these algorithms. Global pressure model for isothermal two-phase immiscible flow in porous media is considered in this paper. Finite-volume method is used for the space discretization of the system of PDEs. Different time stepping discretizations and linearization approaches are discussed. The main concepts of the PDE software tool *MfsolverC++* are given. Numerical results for one realistic problem are presented.

Key words: nonlinear algorithms, finite-volume method, software tools, porous media, flows

1. Introduction

Problems involving the multiphase flow, heat transfer, and multicomponent mass transport in porous media arise in a broad spectrum of engineering disciplines. Important technological applications include the drying of porous solids and soils, subsurface contamination and remediation, thermally enhanced oil recovery, geothermal energy production, porous heat pipes, nuclear reactor safety analysis, high-level secure radioactive waste repositories, paper machines.

Mathematical models for multiphase flow in porous media usually are formulated at macroscopic level and they are obtained by volume averaging or

homogenization methods from microscopic equations. The transport phenomena are mathematically described by the basic principles of conservation for each phase separately and by appropriate interfacial conditions between various phases.

The resulting models are difficult to solve due to large number of strongly coupled nonlinear differential equations in the systems.

In this paper we solve numerically global pressure model for isothermal two-phase immiscible flow problems. The model is described in Section 2. In Section 3 we present the chosen finite volume discretization, and discuss numerical difficulties connected to different approaches for time discretization and for linearization of the governing system of PDEs. Section 4 describes the concept of our software tool for the solution of multiphase flow problems, called `MfsolverC++`, as well as the current state of the implementations. Results of computational experiments are presented in Section 5.

2. Global Pressure Model for Isothermal Two-Phase Immiscible Flow in Porous Media

Traditionally multiphase flow in porous media is described by the macroscopic models, in which various phases are considered as distinct fluids with individual thermodynamic and transport properties [6]. Macroscopic equations are obtained by averaging microscopic equations over the representative elementary volume. The microscopic equations are the equations of mass, momentum and energy conservation for all considered phases. The change of scale enables to convert the real discontinuous medium to a fictitious continuous one. Each macroscopic term is obtained by averaging microscopic ones. The averaging (integration) is done by using various closing assumptions (homogeneity, periodic cell structure etc.) [35].

So, for isothermal two-phase immiscible flow in porous media we have 2 equations for mass conservation in each phase ($k = 1, 2$):

$$\varepsilon \frac{\partial (\rho_k s_k)}{\partial t} + \nabla \cdot (\rho_k \mathbf{u}_k) = 0, \quad (2.1)$$

where ε is the porosity of the porous medium, ρ_k is the phase density, s_k is the phase saturation denoting the volumetric fraction of the void space occupied by phase k . Obviously we have

$$s_1 + s_2 = 1.$$

\mathbf{u}_k is the macroscopic velocity vector (volume rate of flow through a unit cross-sectional area of multiple fluids and porous medium). It is also called bulk velocity. It is assumed that there are no external mass sources or sinks, no mass transfer between phases, and the porous medium is not deformable.

It is also assumed that multiphase extension of Darcy's law is valid for conservation of phase momentum:

$$\mathbf{u}_k = -\mathbf{K} \frac{k_{rk}}{\mu_k} (\nabla p_k - \rho_k \mathbf{g}), \quad (2.2)$$

where \mathbf{K} is the absolute permeability tensor of the porous medium, k_{rk} is the relative permeability of phase k , μ_k is the phase dynamic viscosity, p_k is the phase pressure, and \mathbf{g} is the gravitational vector. The pressures of two phases are related through the capillary pressure, $p_c = p_2 - p_1$. Relative permeability functions and capillary pressure are assumed to be functions of the phase saturations only

$$k_{rk} = k_{rk}(s_k), \quad p_c = p_c(s_1). \quad (2.3)$$

One of the assumptions in generalized Darcy's law is that flow is slow, i.e. inertial effects can be neglected.

The governing equations (2.1)–(2.2) can be mathematically manipulated into several alternate forms with different choices of primary variables: saturation–pressure, pressure–pressure, saturation–saturation formulations. We have chosen global pressure model, sometimes also called fractional flow formulation [7] or mixture model. Comparing to other formulations, equations in this model are less coupled and entering quantities are smoother, because most of them describe mixture properties: velocity, density, etc.

First we define the properties of multiphase mixture through the properties of its components (phases). The mixture density and velocity are defined, respectively, as

$$\rho = \rho_1 s_1 + \rho_2 s_2, \quad (2.4)$$

$$\rho \mathbf{u} = \rho_1 \mathbf{u}_1 + \rho_2 \mathbf{u}_2. \quad (2.5)$$

The mixture kinematic viscosity is given by

$$\nu = \left(\frac{k_{r1}}{\nu_1} + \frac{k_{r2}}{\nu_2} \right)^{-1}, \quad (2.6)$$

where $\nu_k = \mu_k / \rho_k$ is the kinematic viscosity of phase k . Then, the mobility of each phase in the multiphase mixture is defined as

$$\lambda_k = \frac{k_{rk}}{\nu_k} \nu, \quad (2.7)$$

In the literature, λ_k is often called the fractional flow function. We have the following relation for these functions:

$$\lambda_1 + \lambda_2 = 1.$$

The main idea of the model is an introduction of the global (mixture) pressure. The mixture pressure is defined so that the following differential equation holds:

$$\nabla p = \lambda_1 \nabla p_1 + \lambda_2 \nabla p_2 = \nabla p_1 + \lambda_2 \nabla p_c.$$

It is not always possible to find such p . A necessary condition for this is the interchangeability of partial derivatives:

$$\frac{\partial}{\partial x_i} \left(\lambda_2 \frac{\partial p_c}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left(\lambda_2 \frac{\partial p_c}{\partial x_i} \right), \quad i \neq j.$$

For our assumptions this condition is satisfied, and mixture pressure can be defined explicitly as

$$\begin{aligned} p &= p_1 + \int_{S_c}^{s_1} \lambda_2(\xi) \frac{dp_c}{d\xi} d\xi = p_2 - \int_{S_c}^{s_1} \lambda_1(\xi) \frac{dp_c}{d\xi} d\xi \\ &= \frac{p_1 + p_2}{2} + \frac{1}{2} \int_{S_c}^{s_1} (\lambda_2(\xi) - \lambda_1(\xi)) \frac{dp_c}{d\xi} d\xi, \end{aligned} \quad (2.8)$$

where S_c is obtained from the equation $p_c(S_c) = 0$.

Adding the phase mass conservation equations (2.1) we obtain the continuity equation for the mixture:

$$\varepsilon \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (2.9)$$

Multiplying equations (2.2) by corresponding densities and summing them up, we obtain Darcy's law for the mixture:

$$\rho \mathbf{u} = - \frac{\mathbf{K}}{\nu} (\nabla p - (\lambda_1 \rho_1 + \lambda_2 \rho_2) \mathbf{g}). \quad (2.10)$$

Substituting (2.10) into (2.9) we get

$$\varepsilon \frac{\partial \rho}{\partial t} - \nabla \cdot \left(\frac{\mathbf{K}}{\nu} (\nabla p - (\lambda_1 \rho_1 + \lambda_2 \rho_2) \mathbf{g}) \right) = 0. \quad (2.11)$$

Obtained equation is sometimes called pressure equation and is used to find global (mixture) pressure p . The mixture kinematic viscosity ν is positive. The absolute permeability tensor \mathbf{K} of porous medium is positive-definite, so is \mathbf{K}/ν . Consequently, it follows from (2.11) that the pressure equation is *elliptic* (for known ρ).

The diffusive mass flux of phase k within the multiphase mixture is defined as

$$\mathbf{j}_k = \rho_k \mathbf{u}_k - \lambda_k \rho \mathbf{u}, \quad (2.12)$$

or alternatively,

$$\rho_k \mathbf{u}_k = \mathbf{j}_k + \lambda_k \rho \mathbf{u}. \quad (2.13)$$

As a consequence we get that

$$\mathbf{j}_1 + \mathbf{j}_2 = 0.$$

Substituting the momentum equations for the first phase (2.2) and for the mixture (2.10) into equation (2.12), the diffusive mass flux \mathbf{j}_1 can be expressed as follows:

$$\mathbf{j}_1 = \frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} (\nabla p_c + (\rho_1 - \rho_2) \mathbf{g}). \quad (2.14)$$

Substituting equations (2.13) with $k = 1$ and (2.14) into mass conservation equation of the first phase (2.1), we obtain

$$\varepsilon \frac{\partial(\rho_1 s_1)}{\partial t} + \nabla \cdot (\rho \mathbf{u} \lambda_1) = -\nabla \cdot \left(\frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} (\nabla p_c + (\rho_1 - \rho_2) \mathbf{g}) \right). \quad (2.15)$$

Equation (2.15) is called a saturation equation. It is used to find the saturation s_1 . Note that $dp_c/ds_1 < 0$ by the property of capillary pressure. Hence if \mathbf{K} is positive-definite, then (2.15) is a degenerate parabolic equation. The degeneracy is caused by the fact that fractional mobilities λ_1 and λ_2 can become zero. When the capillary forces are small, saturation equation (2.15) is advection dominated. It is purely hyperbolic in the absence of the diffusive forces.

The saturation equation (2.15) in conjunction with the pressure equation (2.11) and the Darcy law (2.10) makes up a full system of equations for unknowns p , \mathbf{u} , and s_1 :

$$\begin{cases} \varepsilon \frac{\partial \rho}{\partial t} - \nabla \cdot \left(\frac{\mathbf{K}}{\nu} (\nabla p - (\lambda_1 \rho_1 + \lambda_2 \rho_2) \mathbf{g}) \right) = 0, \\ \rho \mathbf{u} = -\frac{\mathbf{K}}{\nu} (\nabla p - (\lambda_1 \rho_1 + \lambda_2 \rho_2) \mathbf{g}), \\ \varepsilon \frac{\partial(\rho_1 s_1)}{\partial t} + \nabla \cdot (\rho \mathbf{u} \lambda_1) = -\nabla \cdot \left(\frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} (\nabla p_c + (\rho_1 - \rho_2) \mathbf{g}) \right). \end{cases} \quad (2.16)$$

The system is completed by the constitutive relations (2.3). Equations are defined in Ω , which is assumed to be a polyhedral domain. Both, Dirichlet and full flux (or Neumann) type boundary conditions can be specified on the boundary $\partial\Omega$ of the region.

This system can be used to model isothermal two-phase immiscible fluid flow in porous media, when the phase compositions are of no importance and molecular diffusion and hydraulic dispersion effects can be neglected.

3. Numerical Algorithms

The major numerical difficulties for problems arising in the simulation of processes in porous media are identified by Ewing [14] and Helmig [17] (see also the dissertation of Bastian [4]). They deal with

- solution of convection dominated parabolic problems,
- time-stepping algorithms.

Many problems describing multi-phase flows in porous media are convection dominated parabolic problems. When capillary forces are equal to zero the saturation equation in (2.16) becomes hyperbolic. Sharp fronts of solutions appear often in various applications. They should be resolved accurately by discrete schemes.

It is also very important to preserve the monotonicity of the solutions. Here we should balance between two possibilities. Upwind approximations

are monotone, but the approximation is only first order accurate. Centered differences are second order accurate but can yield oscillatory approximations if the solution is not smooth enough.

First order upwind discretizations are investigated in [4, 10, 17, 25]. A good review of upwind methods for conservation laws, including also a posteriori error estimates for upwind finite volume schemes is done by Kröner [21, 22].

Recently Tadmor *et al.* have proposed non-oscillatory central schemes for one-dimensional and multi-dimensional hyperbolic conservation laws [20, 19, 27], see also [23]. The main advantages of the proposed central schemes are the high resolution, due to the smaller amount of the numerical dissipation, and the simplicity. There are no Riemann solvers and characteristic decomposition involved. At the same time, the developed schemes have an upwind nature, since they respect the directions of wave propagation by measuring the one-sided local speeds. For some cases of schemes even the Jacobians associated with the problem are not required for the computations. The time integration is done by using the explicit predictor–corrector algorithm. Using similar ideas Balaguer and Conde proposed the fourth-order nonoscillatory upwind and central schemes for discretization of hyperbolic conservation laws [3].

In many papers splitting and decomposition methods are used to construct computationally efficient numerical algorithms:

a) Dimensional splitting is an effective method for solving multidimensional problems by constructing the integration algorithm from one-dimensional subproblems – one dimension at a time (see, e.g., [13, 25] or a classical book of Richtmayer and Morton [31]). In the context of nonlinear conservation laws this method is investigated by Crandal and Majda [12]. They proved that dimensional splitting encounters several limitations when applied for solving conservation laws.

b) Splitting in physical processes consists in solving separately a system of conservation laws (e.g. by applying an explicit scheme) and stiff diffusion and reaction problems (by using special implicit algorithms). Different time steps can be used for integration of each subproblem (see, e.g., [2]).

c) Local characteristic decomposition is used for the accurate resolution of complicated structure of solutions. The development of such methods started from the Godunov method, and now a large number of special classes of methods, e.g. ENO, WENO algorithms, are proposed (see, e.g. [25, 26, 30]). While this decomposition increases the computational cost, it controls spurious oscillations when the order of accuracy is high.

Our goal is to build a universal tool for solving a wide variety of problems of multiphase flows in porous media. Thus we have decided to start with not the most complicated but flexible and robust approximations. We try to avoid the time-consuming computation of approximate Riemann solvers and the related characteristic decompositions. Since dimensional splitting and splitting in physical processes also introduce numerical and non-physical errors, we avoid splitting also.

3.1. Finite volume discretization

We discretize equations (2.16) on structured nonuniform grids. A 2D space Ω is used as an example in this section. We have a set of vertices:

$$V_h = \{v_{ij} = (x_{1i}, x_{2j}), \quad 0 \leq i \leq N_i, \quad 0 \leq j \leq N_j, \}$$

where:

$$x_{1,i} = x_{1,i-1} + h_{1,i}, \quad x_{2,j} = x_{2,j-1} + h_{2,j}.$$

The grid V_h defines a structured mesh E_h , which covers the region Ω :

$$E_h = \{e_{ij}, \quad 1 \leq i \leq N_i, \quad 1 \leq j \leq N_j\}.$$

Elements e_{ij} of the mesh are called cells or control volumes (thus, in fact we define a dual mesh in the terminology of [4]). In 2D we get *quadrilateral* elements. These elements can be *cell-centered* or *vertex-centered*. Our scheme is based on cell-centered mesh, and these vertices make a primary grid

$$\Omega_h = \{X_{ij}, \quad 1 \leq i \leq N_i, \quad 1 \leq j \leq N_j\}.$$

For boundary elements, one or more faces of which belong to the boundary $\partial\Omega$, the vertices are taken on the boundary. Let denote:

$$h_{1,i+1/2} = X_{i+1,j} - X_{ij}, \quad h_{2,j+1/2} = X_{i,j+1} - X_{ij}.$$

Based on the given grid we can define a finite dimensional function space of piecewise linear functions:

$$W_h = \{U(\mathbf{x}) : U(\mathbf{x}) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} U_{ij} \varphi_{ij}(\mathbf{x})\}, \quad (3.1)$$

where $\varphi_{ij}(\mathbf{x})$ are the usual nodal basis functions.

We describe the finite-volume scheme for the saturation equation, since the pressure equation is approximated in a similar way. Integrating the saturation equation (2.16) over the control element $e \in E_h$ and applying the divergence theorem, we obtain

$$\int_e \varepsilon \frac{\partial(\rho_1 s_1)}{\partial t} d\mathbf{x} + \int_{\partial e} \left(\rho \mathbf{u} \lambda_1 + \frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} (\nabla p_c + (\rho_1 - \rho_2) \mathbf{g}) \right) \cdot \mathbf{n} ds = 0.$$

The numerical diffusion flux over control element face is computed by using harmonic or arithmetic average of the diffusion coefficient. For example, in 2D case taking only one part of the element interface we get the following numerical diffusion flux:

$$\int_{\partial e_{i+1/2,j}} \frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} \nabla p_c \cdot \mathbf{n} ds \approx h_{2j} \left(\frac{\mathbf{K} \lambda_1 \lambda_2}{\nu} \right)_{i+1/2,j} \frac{p_c(S_{i+1,j}) - p_c(S_{ij})}{h_{1,i+1/2}}.$$

The evaluation of the advective part of the flux is done by using the *fully upwinding* method

$$\int_{\partial e_{i+1/2,j}} \rho \mathbf{u} \lambda_1(S) \cdot \mathbf{n} ds \approx h_{2j} (\rho u_1)_{i+1/2,j} \cdot \begin{cases} \lambda_1(S_{ij}), & \text{if } \rho u_{1,i+1/2,j} \lambda_1'(S_{ij}) \geq 0, \\ \lambda(S_{i+1,j}), & \text{else.} \end{cases}$$

Fully upwinding discretization of the advective flux leads to positive diagonal and negative off-diagonal entries in the matrix obtained after linearization of nonlinear discrete equations by the Newton method and the discrete solution obeys a discrete maximum principle.

The drawback of such approximation is that artificial (numerical) diffusion is introduced. It can be split into two components: one along the streamline and a second component perpendicular to the streamline [17]. The second component exists only for multi – dimensional problems. A multidimensional-upwind method of first order is considered in [17], which try to reduce the grid orientation effects by taking into account the direction of the velocity field. Our strategy is to reduce the approximation error by decreasing the mesh step, since the obtained large systems of equations can be solved efficiently with parallel computers. For very fine mesh central differencing can be used also.

In our tool for 1D problems we have also implemented an explicit higher order TVD approximation of the advection flux, when the central difference approximation is combined with the upwind approximation by using the *minmod* limiter. Let consider the case, when $\rho u_{1,i+1/2,j} \lambda_1'(S_{ij}) \geq 0$, then the advection flux is approximated as follows

$$\int_{\partial e_{i+1/2,j}} \rho \mathbf{u} \lambda_1(S) \cdot \mathbf{n} ds \approx h_{2j} (\rho u_1)_{i+1/2,j} \lambda_1(S_{i+1/2,j}),$$

$$S_{i+1/2,j} = S_{ij} + \frac{h_{i+1/2,j}}{2} \minmod\left(\frac{S_{i+1,j} - S_{ij}}{h_{i+1/2,j}}, \frac{S_{ij} - S_{i-1,j}}{h_{i-1/2,j}}\right),$$

where the limiter is defined by

$$\minmod(a, b) = \begin{cases} a, & \text{if } ab > 0, \text{ and } |a| \leq |b|, \\ b, & \text{if } ab > 0, \text{ and } |b| \leq |a|, \\ 0, & \text{if } ab < 0. \end{cases}$$

After discretization the mass is conserved locally in each element and therefore the discrete problem satisfies the mass conservation property in the whole region Ω .

3.2. Time–stepping algorithms

After approximation of space derivatives in system (2.16) a large system of nonstationary nonlinear DEs is obtained

$$\begin{cases} L_h(S)P(t) = F_p, \\ \frac{dS(t)}{dt} = C_h(S, P) + D_h(S) + F_s. \end{cases} \quad (3.2)$$

Here the first equation approximates the global pressure equation, $C_h(S, P)$ denotes the convection part and $D_h(S, P)$ the diffusion part of the semi-discrete saturation equation. We consider incompressible case. Let denote saturation and pressure values at time t^n by S^n and P^n , respectively.

The aim of time-stepping algorithms is to balance the stability and robustness of the integration algorithm with computational efficiency. The one way is to use a some kind of segregated solution procedure. With this approach the new pressure field P^{n+1} is computed first from

$$L_h(S^n)P^{n+1} = F_p^n, \quad (3.3)$$

which is a linear elliptic equation in P for a given saturation at previous time level t^n . Next, saturation at time level t^{n+1} is computed with a velocity field obtained from the new pressure field. The most simple way is to use the forward Euler scheme:

$$\frac{S^{n+1} - S^n}{\tau} = C_h(S^n, P^{n+1}) + D_h(S^n) + F_s^n. \quad (3.4)$$

This time-stepping procedure is named IMPES (for implicit pressure and explicit saturation). It was originally developed by Sheldon *et al.* [33] and Stone and Garder [34] for solving partial differential coupled system for two-phase flow in porous medium (see also [4, 17]). Due to the explicitness of the scheme for saturation equation IMPES is stable only when time steps are sufficiently small. An improved IMPES method which utilizes an adaptive control strategy for selection of time step is proposed in [9]. The stability of IMPES scheme for solving a three-phase flow problem is investigated in [11].

For many applications the strongest restriction on time step is due the diffusion part of saturation equation. The following implicit-explicit modification of (3.4) is used in some papers (see, e.g. [2]):

$$\frac{S^{n+1} - S^n}{\tau} = C_h(S^n, P^{n+1}) + D_h(S^{n+1}) + F_s^{n+1}, \quad (3.5)$$

which consists of applying an explicit discretization for the advection term and an implicit discretization for the diffusion and source terms. We note that the approximation of different terms on different time levels leads to truncation errors similar to additive scheme case. In fact scheme (3.5) is equivalent to the following splitting scheme

$$\begin{cases} \frac{S^{n+1/2} - S^n}{\tau} = C_h(S^n, P^{n+1}), \\ \frac{S^{n+1} - S^{n+1/2}}{\tau} = D_h(S^{n+1}) + F_s^{n+1}, \end{cases}$$

but the additive scheme is more flexible, since different time steps can be used for integration of each subproblem.

The next step to increase the stability of the discrete scheme is to use backward Euler (or more generally an implicit) integration method for the saturation equation:

$$\frac{S^{n+1} - S^n}{\tau} = C_h(S^{n+1}, P^{n+1}) + D_h(S^{n+1}) + F_s^{n+1}. \quad (3.6)$$

At each time level we get a nonlinear elliptic equation with advection-dominated space operator. It can be solved by some iterative procedure, e.g. the Newton method. The whole time-stepping procedure (3.3), (3.6) is named IMPIMS.

Another way is to use the coupled schemes. In our case, the *fully implicit/fully coupled* (FIFC) scheme gives the system

$$\begin{cases} L_h(S^{n+1})P^{n+1} = F_p^{n+1}, \\ \frac{S^{n+1} - S^n}{\tau} = C_h(S^{n+1}, P^{n+1}) + D_h(S^{n+1}, P^{n+1}) + F_s^{n+1}, \end{cases} \quad (3.7)$$

when both equations are solved simultaneously. The FIFC approach is usually selected for a maximum of robustness and stability. However, it can be very expensive computationally.

Our goal is to have a tool working with both: segregated and coupled solvers, to be able to choose the best for the specific problem. Currently, the IMPIMS type iterative algorithm is implemented in our tool with additional "k"-iterations between separate equation solvers at each time level:

$$\begin{cases} L_h(S^{n+1,k})P^{n+1,k+1} = F_p^{n+1,k}, \\ \frac{S^{n+1,k+1} - S^n}{\tau} = C_h(S^{n+1,k+1}, P^{n+1,k+1}) + D_h(S^{n+1,k+1}) + F_s^{n+1,k+1}. \end{cases} \quad (3.8)$$

An quasi-Newton method is used to solve the nonlinear saturation equation. Let us write the diffusion term as

$$D_h(S) = \nabla_h \cdot (K(S)\nabla_h P_c(S)).$$

Let $S^m = S^{n+1,k+1,m}$ denote the m -th iteration of quasi-Newton method. Then the next iterative approximation is defined as

$$S^{m+1} = S^m + S_{\Delta}^m,$$

where S_{Δ}^m satisfies the following equation:

$$\begin{aligned} \frac{S_{\Delta}^m}{\tau} &= \partial C_h(S^m, P^{n+1,k+1})S_{\Delta}^m + \nabla_h \cdot (K(S^m)P'_c(S^m)\nabla_h S_{\Delta}^m) + R^m, \\ R^m &= -\frac{S^m - S^n}{\tau} + C_h(S^m, P^{n+1,k+1}) + \nabla_h \cdot (K(S^m)\nabla_h P_c(S^m)) + F_s(S^m). \end{aligned}$$

4. MfsolverC++: A Tool for Computations

Recently, it becomes very popular in the scientific computing to exploit the object-oriented programming (OOP) techniques. This allows developers to reduce the time spent on the programming and debugging and makes all implementation aspects cleaner and simpler.

We also use OOP in the design and implementation of our software tool `MfsolverC++` for computations of multiphase flows in porous media. We want to have a flexible set of efficient C++ modules to reduce the overhead for creation of different solvers for different schemes and problems.

Implementation of the solver for the solution of isothermal two-phase immiscible flow problems using global pressure model is shown in Figure 1. For this and other segregated solvers we use strategy proposed in [24] and used in *Diffpack* software library. According to it, solvers for the systems of PDEs are built by merging together the independent solvers for the alone-standing equations that enter the system.

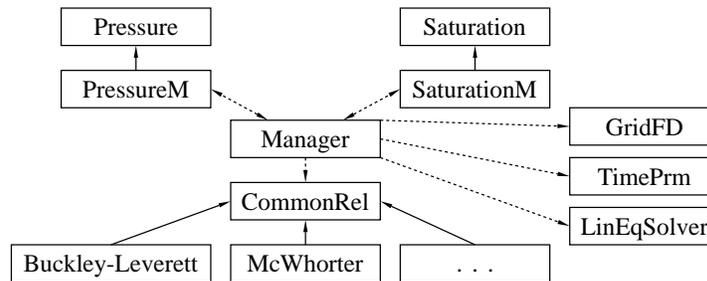


Figure 1. Design of the PDE system solver with relations between solver classes, manager and pool of common relations. Solid arrows indicate inheritance ("is-a" relationship, with arrows pointing from subclass to base class). Dashed arrows indicate pointer ("has-a" relationship).

Class `Pressure` and his children are independent solvers of pressure equation (2.11). Class `Saturation` and his children are independent solvers of saturation equation (2.15). Equations become coupled into the system through the coefficients. In our PDE solvers, these coefficients, including the source, initial and boundary conditions functions, are represented by virtual functions. Subclasses of PDE solvers override these functions and implement the physically relevant versions, when the coefficients are coupled to other unknown fields in the PDE system. All these functions are often built of a common set of relations (constitutive relationships, model definitions, etc.). Therefore, they are collected in class hierarchies and accessed from PDE solvers through a base class `CommonRel` interface (pointer). Note that constitutive relationships can be easily changed without affecting the code in PDE solvers.

A class `Manager` acts as the solver class for the whole PDEs system and manages the whole solution of the given problem. This class contains two way

pointers to the subclasses for solving the pressure and saturation equations, `PressureM` and `SaturationM`, respectively, which enable the coupling by overriding the virtual functions of base classes with the functions from common relation hierarchy. The manager is also responsible for creating a space grid and time discretization. It allocates a common linear system and solver object, and distributes all these data to the PDE solver classes.

It is clear that segregated solvers have "implementational" advantage over coupled solvers. They are easier to implement, flexible and naturally extensible if the differential model is extended by additional PDEs.

Next we review very briefly some similar projects. First we mention a general PDE software tool *Diffpack*, which is an object oriented development framework for the solution PDE. It is a commercial product and many application examples are described in [24].

The toolbox UG is a framework for unstructured grid computations. Development of UG started in 1990 at the University of Heidelberg and proceeded at the University of Stuttgart from 1994. It is a flexible software tool for the numerical solution of partial differential equations on unstructured meshes in two and three space dimensions using multigrid methods.

A number of applications of this tool for computations of complex fluid flows in porous media are described in [5, 17]. For the time discretization either backward Euler, Crank-Nicholson or BDF(2) are used. The fully implicit/fully coupled time stepping strategy is selected for a maximum of robustness and stability. In UG the Newton method is used to solve large systems of nonlinear algebraic equations at each time step. The linear systems arising within the Newton method are solved with a multigrid method.

TOUGH2 is a general-purpose numerical simulation program for multiphase fluid and heat flow in porous and fractured media [29]. It is developed at Lawrence Berkeley National Laboratory and is widely used for computations on unstructured meshes (see e.g [28]).

In TOUGH2 code various strategies are implemented. In order to take into account the hyperbolic nature of the saturation equation very accurate explicit time-stepping algorithm TOUGH2-EOS3f with front tracking is used.

5. Computational Experiments

In this section we consider the injection process of carbone dioxide (CO_2) into saline aquifers. Here an aqueous phase is displaced by a less dense and less viscous gas phase [28]. It is well known that immiscible displacement of a fluid by a less viscous one leads to the creation of fingers of low viscosity fluid penetrating the high viscosity fluid (see [18, 32]). On simulation level such phenomena arises due instabilities of macroscopic equations when the frontal mobility ratio is grater than one [4]. The appearance of finger-like structures can be triggered by

- numerical round-off errors,
- introducing a mild random variations of the permeability field.

We will note one more interesting aspect of this test problem (see [4, 15]). The phenomenon of viscous fingering comes from instabilities on *microscopic* level which are not modelled by the *macroscopic* equations, obtained after averaging process. Thus we can test a possibility to represent effects from smaller scale in a model on a larger scale. Numerical aspects of upscaling in simulation of flow porous media are investigated in [1, 8, 16].

The parameters of the model were taken from [28]. The capillary effects were assumed to be negligible. Then in fact the problem reduces to the Buckley - Leverett equation. Exactly this formulation was used in [28]. We describe the given problem in global pressure formulation and use a developed `MfsolverC++` tool. The most important coefficients are the relative permeability functions, which define the fractional flow function. The liquid is described by the van Genuchten function

$$k_{rl} = \sqrt{S^*} \left(1 - \left(1 - (S^*)^{1/m} \right)^m \right)^2, \quad S^* = \frac{S_l - S_{lr}}{1 - S_{lr}},$$

and the gas is described by the Brooks-Corey function:

$$k_{rg} = (1 - \hat{S})^2 (1 - \hat{S}^2), \quad \hat{S} = \frac{S_l - S_{lr}}{1 - S_{lr} - S_{gr}}.$$

For CO_2 and water the viscosity was taken equal to $\mu_g = 4 \cdot 10^{-5}$ and $\mu_l = 7.5 \cdot 10^{-4}$, respectively. The relative permeability parameters were defined as $S_{lr} = 0.25$, $S_{gr} = 0.05$, $m = 0.85$.

For the validation of the numerical discretizations and algorithms we solved one-dimensional case of this problem. Then a quasi-analytical solution is known (see [4, 17]) and thus it is possible to investigate the convergence of the numerical solution to the exact solution and to determine the experimental order of convergence.

The domain is discretized with N elements. The L_p -norms of the error in the saturation variable,

$$\|s_g - S_{gh}\|_{L_p} = \left(\int_{\Omega} |s_g - S_{gh}|^p \right)^{1/p},$$

are computed for $p = 1, 2$. The convergence rate r is determined as

$$r = \log \left(\frac{\|s_g - S_{gh}\|_{L_p}}{\|s_g - S_{g2h}\|_{L_p}} \right) / \log \frac{1}{2}.$$

Table 1 presents the error norms and the convergence rates for different numbers of the cells.

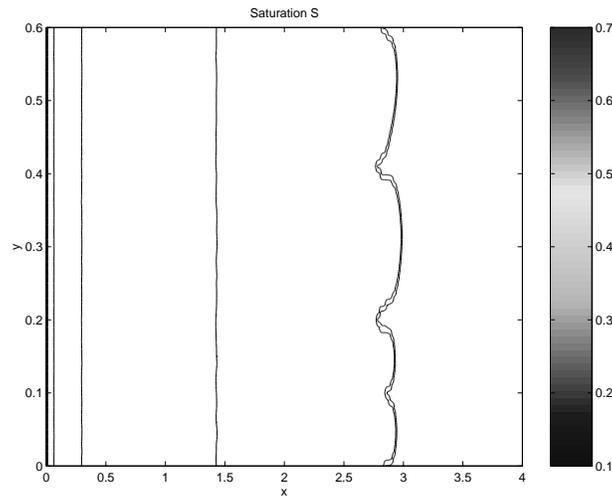
The results in table show that the experimental convergence rates are not reaching the optimal approximation orders of a step function with a linear spline, which are equal to $\mathcal{O}(h)$ in the L_1 -norm and $\mathcal{O}(h^{1/2})$ in the L_2 -norm.

The fingers were triggered by a random permeability field. We modified the permeability in the order of 5 %. It is well-known that the upwind discretization introduces a numerical diffusion, in order to reduce it the computations

Table 1. Experimental order of convergence for the 1D problem.

N	error L_1	rate L_1	error L_2	rate L_2
65	0.1366	0.707	0.1023	0.391
129	0.0854	0.676	0.0786	0.380
257	0.0554	0.626	0.0605	0.377
513	0.0377	0.554	0.0466	0.378
1025	0.0274	0.463	0.0356	0.386

were done on very fine grids. Figure 2 shows the distribution of the gas saturation for one case of parameters. We have not investigated the convergence of the solution under grid refinement.

**Figure 2.** Distribution of gas saturation. Viscous fingers are triggered by the modified permeability field.

Acknowledgements

The work has been partially supported by EC under the project INTAS-03-50-4395; by the Lithuanian Science Foundation grant T-06173 "Parallel numerical algorithms for computation of fluid flows in porous media", and by German BMBF under Grant GIN-SF-042 "Modellbasiertes Design von Brennstoffzellen und Brennstoffzellensystemen: PEMDesignNET".

References

- [1] T. Arbogast. Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems. *SIAM J. Numer. Anal.*, **42**(2), 576–598, 2005.
- [2] U. Asher, S. Ruuth and B. Wetton. Implicit–explicit methods for time dependent PDE’s. *SIAM J. Numer. Anal.*, **32**, 797–823, 1995.
- [3] A. Balaguer and C. Conde. Fourth-order nonoscillatory upwind and central schemes for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, **43**, 455–473, 2006.
- [4] P. Bastian. *Numerical Computation of Multiphase Flows in Porous Media*. Habilitation Dissertation, Kiel university, 1999.
- [5] P. Bastian, K. Birken, S. Lang, K. Johannsen, N. Neuss, H. Rentz-Reichert and C. Wieners. UG: A flexible software toolbox for solving PDE. *Computing and Visualization in Science*, **1**, 27–40, 1997.
- [6] J. Bear. *Dynamics of Fluids in Porous Media*. Elsevier, New York, 1972.
- [7] G. Chavent. A new formulation of diphasic incompressible flow in porous media. *Lecture Notes in Mathematics*, **503**, 1976.
- [8] Y. Chen, L.J. Durlofsky, M. Gerritsen and X.H. Wen. A coupled local–global upscaling approach for simulating flow in highly heterogeneous formations. *Advances in Water Resources*, **26**, 1041–1060, 2003.
- [9] Z. Chen, G. Huan and B. Li. An improved IMPES method for two–phase flow in porous media. *Transport in Porous Media*, **54**, 361–376, 2004.
- [10] R. Čiegis and V. Starikovičius. Mathematical modelling of wood drying process. *Mathematical modelling and analysis*, **7**(2), 177–190, 2002.
- [11] K.H. Coats. IMPES stability: selection of stable time steps. *SPE Journal*, **8**(2), 181–187, 2003.
- [12] M. Crandall and A. Majda. The method of fractional steps for conservation laws. *Numer. Math.*, **34**, 285–314, 1980.
- [13] B. Enquist and O. Runborg. Multi–phase computations in geometrical optics. *J. Comput. Appl. Math.*, **74**, 175–192, 1996.
- [14] R. Ewing. Problems arising in the modeling of processes for hydrocarbon recovery. In: R. Ewing(Ed.), *Research Frontiers in Applied Mathematics*, volume 1. SIAM, 3–34, 1983.
- [15] R. Ewing. Aspects of upscaling in simulation of flow in porous media. *Adv. Water Resources*, **20**(5-6), 349–358, 1997.
- [16] C.L. Farmer. Upscaling: a review. *Int. Journal Numer. Meth. Fluids*, **40**, 63–78, 2002.
- [17] R. Helmig. *Multiphase Flow and Transport Processes in the Subsurface – A Contribution to the Modelling of Hydrosystems*. Springer–Verlag, 1997.
- [18] G.M. Homsy. Viscous fingering in porous media. *Ann. Rev. Fluid Mech.*, **19**, 271–311, 1987.
- [19] G.-S. Jiang and E. Tadmor. Nonoscillatory central schemes for multidimensional hyperbolic conservation laws. *SIAM J. Sci. Comput.*, **19**(6), 1892–1917, 1998.
- [20] G.S. Jiang, D. Levy, C.-T. Lin, S. Osher and E. Tadmor. High-resolution nonoscillatory central schemes with nonstaggered grids for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, **35**(6), 2147–2168, 1998.
- [21] D. Kröner. *Numerical Schemes for Conservation Laws*. Wiley und Teubner, 1996.
- [22] D. Kröner and M. Ohlberger. A posteriori error estimates for upwind finite volume schemes in several space dimensions. *Math. Comput.*, **69**, 25–39, 2000.

- [23] A. Kurganov, S. Noelle and G. Petrova. Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton–Jacobi equations. *SIAM J. Sci. Comput.*, **23**(3), 707–740, 2001.
- [24] H.P. Langtangen. *Computational Partial Differential Equations. Numerical Methods and Diffpack Programming*. Springer, Berlin, 2002.
- [25] R.J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Verlag, 1991.
- [26] X.-D. Liu and P.D. Lax. Positive schemes for solving multi-dimensional hyperbolic systems of conservation laws. *CFD Journal*, **5**, 1–24, 1996.
- [27] H. Nessyahu and Tadmor E. Non-oscillatory central differencing for hyperbolic conservation laws. *J. Comput. Phys.*, **87**, 408–448, 1990.
- [28] K. Pruess and J.E. Garcia. Multiphase flow dynamics during CO_2 disposal into saline aquifers. *Environmental Geology*, **42**(2-3), 282–295, 2002.
- [29] K. Pruess, C. Oldenburg and G. Moridis. TOUGH2 users’s guide, version 2.0. Technical report lbnl-43134, Lawrence Berkeley National Laboratory, Berkeley, California, 1999.
- [30] J. Qiu and C.W. Shu. On the construction, comparison, and local characteristic decomposition for high-order central WENO schemes. *Journal of Computational Physics*, **183**(1), 187–209, 2002.
- [31] R.D. Richtmyer and K.W. Morton. *Difference Methods for Initial Value Problems*. Interscience, New York, 1967.
- [32] P.G. Saffman and S.G. Taylor. The penetration of a fluid into a porous medium or Hele–Shaw cell containing a more viscous liquid. In: *Proc. Roy. Soc. London Series A.*, volume 245, 225–257, 1958.
- [33] J.W. Sheldon, B. Zondek and W.T. Cardwell. One-dimensional, incompressible, non-capillary, two-phase fluid flow in a porous medium. *T. SPE AIME*, **216**, 290–296, 1959.
- [34] H.L. Stone and A.O. Garder. Analysis of gas-cap or dissolved-gas reservoirs. *T. SPE AIME*, **222**, 92–104, 1961.
- [35] S. Whitaker. Simultaneous heat, mass, and momentum transfer in porous media: A theory of drying. *Advances in Heat Transfer*, **13**, 119–203, 1977.