Russ. J. Numer. Anal. Math. Modelling, Vol. 28, No. 3, pp. 267–288 (2013)
DOI 10.1515/rnam-2013-0016

Two-phase water flooding simulations on dynamic adaptive octree grids with two-point nonlinear fluxes

K. TEREKHOV[*][†] and Yu. VASSILEVSKI[*][‡]

Abstract — We present a method for numerical simulation of the two-phase water flooding problem on general polyhedral grids not aligned with permeability tensor $\mathbb{K}$ ($\mathbb{K}$-non-orthogonal grids) and dynamic octree grids adapted to the front between the phases. The discretization is based on the cell-centered monotone finite volume (FV) method with the nonlinear two-point flux approximation (TPFA) applicable to general $\mathbb{K}$-non-orthogonal polyhedral grids. We use fully implicit discretization in time to avoid the restriction on the time step caused by the minimal mesh size. In our numerical experiments we demonstrate the superiority of the nonlinear TPFA on a $\mathbb{K}$-non-orthogonal grid over linear TPFA and considerable speed-up of the simulation on a dynamically adapted octree grid with minimal loss in accuracy compared to the simulation on a fine regular grid.

Two-phase water flooding is the secondary stage of oil recovery represented by the two-phase black oil model equations. At this stage, water is injected into injector wells, while oil is produced through producer wells. In two-phase water flooding simulation, appropriate recovery of the moving water front is very important. The presence of the full essentially anisotropic permeability tensor $\mathbb{K}$ and grids not aligned with the tensor $\mathbb{K}$ ($\mathbb{K}$-non-orthogonal grids) and the computer memory restriction on the minimal grid size transform the numerical simulation into a challenge. We use two approaches to this challenge: the nonlinear finite volume (FV) method and the

[*] Institute of Numerical Mathematics, Russian Academy of Sciences, Moscow 119333, Russia

[†] Nuclear Safety Institute, Russian Academy of Sciences, Moscow 113191, Russia

[‡] Moscow Institute of Physics and Technology, Dolgoprudny 141700, Moscow Region, Russia. Corresponding author. E-mail: yuri.vassilevski@gmail.com

This work was carried out under the auspices of ExxonMobil Upstream Research Company and has been partly supported by RFBR grants 11-01-00971, 12-01-31223, 12-01-31275, 12-01-33084, Federal contracts 14.740.11.1389, 14.514.11.4057, and Rosatom project ‘Breakthrough’.

dynamically adapted octree grid where the sizes of neighbouring cells differ at most by factor 2. The simple Cartesian structure of the octree and embedded hierarchy [8] make grid adaptation fast and easy. The use of dynamically adapted octree grids rises several issues: time step restrictions, criterion for refinement, data interpolation, and grid non-conformity.

Two main approaches are used for time discretization of two-phase black oil equations: IMPES [2, 25, 27] (implicit in pressure, explicit in saturation) and FI [3] (fully implicit) schemes. The use of the IMPES scheme imposes restrictions on the time step caused by the minimal mesh size, which can be small for adaptively refined grids. In this article we use fully implicit discretization due to its unconditional stability. The control of adaptive refinement in two-phase water flooding simulations may be based on *a posteriori* error estimators [4, 5] or physical considerations [6, 24]. We refine the mesh in the regions of the high gradients of water saturation and oil pressure. We treat high gradients of water saturation as the sharp front between two phases and high gradients of oil pressure as peculiarities of Darcy's velocities. For interpolation of data between two nested grids we use local conservative interpolation [9].

In our discretization we treat the octree grid as a general conformal polyhedral mesh where each cell may have at most 24 neighbours and every face have at most 2 cells. Following [7,10,12–14,18,23,26,28] we use the nonlinear FV method based on nonlinear two-point flux approximation (nonlinear TPFA) pioneered by Le Potier [23]. Its compact discretization is second-order accurate even on $\mathbb{K}$-non-orthogonal grids and preserves the positivity of the differential solution. The discretization has been successfully used in two-phase water flooding simulations using the IMPES scheme [16] and the FI scheme with approximate Jacobian calculation [19]. Although unused here, the recent development of the nonlinear FV method [15] is worth mentioning. It provides the discrete maximum principle at a sacrifice in the two-point flux stencil, yet keeping the stencil of the discrete operator the most compact.

Another approach to discretization on $\mathbb{K}$-non-orthogonal grids is the multi-point flux approximation (MPFA) scheme that uses more than two points in the flux stencil [1] and generates a matrix of transmissibility coefficients. The MPFA scheme is also second-order accurate, but is often only conditionally stable [11] and conditionally monotone [20], in contrast to the nonlinear monotone FV methods. For the comparison of the nonlinear TPFA and MPFA performance we refer to [17].

Since we use the fully implicit scheme in time, we solve nonlinear problems by the Newton method that requires a Jacobian matrix of the nonlinear operator. We consider three approaches to the computation of the Jacobian matrix. First, we consider approximate Jacobian, which ignores nonlinearity in the discrete flux coefficients and preserves the stencil of the conventional FV method with the linear TPFA. Second, we consider a true Jacobian matrix due to differentiation of all nonlinearities of the mesh operators; the matrix contains more non-zero entries than the approximate Jacobian. In the third approach we drop off small values according to a predefined threshold and thus sparsify the true Jacobian matrix. This method turns to be the most efficient, according to our experience.

We prove the efficiency of our approaches by several numerical experiments. In particular, we show the superiority of the nonlinear TPFA calculation over the conventional TPFA calculation on $\mathbb{K}$-non-orthogonal grids and demonstrate essential speed-up of water flooding simulations using dynamically adapted octree grids.

The novelty of our work is the first application of the monotone FV method with nonlinear TPFA to a fully implicit solution of two-phase black oil equations, including the derivation of Jacobian matrix and the use of adaptively refined octrees to track the water fronts in water flooding simulations.

The paper outline is as follows. In Section 2, we formulate the equations of the two-phase black oil model. Discretizations in time and space are described in Sections 3 and 4, respectively. In Section 5, we present formulas for the calculation of Jacobian matrices. Comparisons of nonlinear and linear TPFAs for a water flooding benchmark are presented in Section 6. The efficiency of using dynamically adapted octrees in water flooding simulation is discussed in Section 7.

1. Two-phase black oil model

We consider a two-phase flow of immiscible fluids in a porous medium [3, 21]. The phase that wets the medium more than the other, is called the wetting phase and is indicated by subscript w. The other phase is the non-wetting phase and is indicated by o. Here S_α and p_α stand for the saturation and pressure of phase $\alpha = w, o$.

The basic equations for the two-phase flow are the following:

- Mass conservation for each phase:

$$\frac{\partial}{\partial t}\frac{\varphi S_\alpha}{B_\alpha} = -\mathrm{div}\mathbf{u}_\alpha + q_\alpha, \quad \alpha = w, o. \tag{1.1}$$

- Darcy's law:

$$\mathbf{u}_\alpha = -\lambda_\alpha \mathbb{K}\left(\nabla p_\alpha - \rho_\alpha g \nabla z\right), \quad \alpha = w, o. \tag{1.2}$$

- Two fluids fill the voids:

$$S_w + S_o = 1. \tag{1.3}$$

- Pressure difference between the phases is given by capillary pressure $p_c = p_c(S_w)$:

$$p_o - p_w = p_c \tag{1.4}$$

where $\mathbb{K}$ is the absolute permeability tensor, φ is porosity, g is the gravity term, z is depth, in the phase α: p_α is unknown pressure, S_α is unknown saturation, $\mathbf{u}_\alpha$ is unknown Darcy's velocity, ρ_α is unknown density, $B_\alpha = \rho_{\alpha,0}/\rho_\alpha$ is the formation volume factor, μ_α is the viscosity, $k_{r\alpha}$ is the relative phase permeability, $\lambda_\alpha = k_{r\alpha}/(\mu_\alpha B_\alpha)$ is the mobility, q_α is the source/sink well term.

We choose oil pressure p_o and water saturation S_w as the primary unknowns. In the sequel we also take into account the following dependences: $k_{r\alpha} = k_{r\alpha}(S_w)$, $\mu_\alpha = \mu_\alpha(p_o)$, $B_\alpha = B_\alpha(p_o)$ and $\varphi = \varphi_0\left(1 + c_R(p_o - p_o^0)\right)$. Here c_R is the rock matrix compressibility constant.

We have the no-flow (homogeneous Neumann) boundary condition on the reservoir boundary. The wells are incorporated through the well terms in (1.1). Each well is assumed to be vertical and connected to the center of a cell. It is assumed that there is no capillary pressure in the wells, so both water and oil fluxes depend on the same (oil) pressure. The formula for the well term has been suggested by Peaceman [22]. For a cell T with a center $\mathbf{x}_T$ connected to the well we have:

$$q_\alpha = \frac{\rho_\alpha k_{r\alpha}}{\mu_\alpha} WI\left(p_{\mathrm{bh}} - p_o - \rho_\alpha(z_{\mathrm{bh}} - z)\right)\delta(\mathbf{x} - \mathbf{x}_T) \tag{1.5}$$

where p_{bh} is a given bottom hole pressure, $\delta(\mathbf{x}-\mathbf{x}_T)$ is the Dirac function, WI is the well index, which does not depend on the properties of the fluids, but depends on the properties of the media and the cell sizes h_x, h_y, h_z. For a $\mathbb{K}$-orthogonal hexahedral cell, $\mathbb{K} = \operatorname{diag}\{K_x, K_y, K_z\}$, one has

$$WI = \frac{2\pi h_z \sqrt{K_x K_y}}{\log(r/r_w)+s}, \quad r = 0.28 \frac{\left((K_y/K_x)^{1/2} h_x^2 + (K_x/K_y)^{1/2} h_y^2\right)^{1/2}}{(K_y/K_x)^{1/4} + (K_x/K_y)^{1/4}} \tag{1.6}$$

where r_w is the well radius and s is the skin factor.

2. Fully-implicit discretization

The mass conservation equations (1.1) are discretized fully implicitly in time:

$$\frac{\left(\dfrac{\varphi S_\alpha}{B_\alpha}\right)^{n+1} - \left(\dfrac{\varphi S_\alpha}{B_\alpha}\right)^{n}}{\Delta t^{n+1}} = -\operatorname{div}\mathbf{u}_\alpha^{n+1} + q_\alpha^{n+1}, \qquad \alpha = w, o. \tag{2.1}$$

Now we can define the nonlinear residual for the lth approximation to a quantity evaluated at the time step $n+1$ inside the grid cell T_i:

$$R_{\alpha,i} = \int_{T_i} \left[\left(\frac{\varphi S_\alpha}{B_\alpha}\right)^{l,i} - \left(\frac{\varphi S_\alpha}{B_\alpha}\right)^{n,i} + \Delta t^{n+1} \left(\operatorname{div}\mathbf{u}_\alpha - q_\alpha\right)^{l,i} \right] \mathrm{d}x, \quad \alpha = w, o. \tag{2.2}$$

The discrete counterpart of (2.1) can be written as:

$$R_{\alpha,i} = 0, \quad \alpha = w, o \tag{2.3}$$

for all grid cells T_i at every time step.

The combination of (1.2), (2.2), (2.3) generates a nonlinear system, which is usually solved by the Newton method:

$$J(x^l)\delta x^l = -R(x^l) \tag{2.4}$$

$$x^{l+1} = x^l + \delta x^l \tag{2.5}$$

where l is the lth Newton step, $x = (p_o S_w)^T$ is a vector of primary unknowns in all grid cells, $R(x) = (R_w(x) R_o(x))^T$ is a vector of nonlinear residuals in all grid cells, and J is the Jacobian matrix:

$$J(x) = \begin{pmatrix} \dfrac{\partial R_w}{\partial p_o}(x) & \dfrac{\partial R_w}{\partial S_w}(x) \\ \dfrac{\partial R_o}{\partial p_o}(x) & \dfrac{\partial R_o}{\partial S_w}(x) \end{pmatrix}.$$

We terminate Newton iterations when the norm of the residual vector drops below ε_{nwt}.

3. Finite-volume method

Let $\Omega \in \mathbb{R}^3$ be a polyhedral domain and $\mathscr{T}$ be a conformal polyhedral mesh in Ω composed of $N_{\mathscr{T}}$ shape-regular cells with planar faces. We assume that each cell T is a star-shaped 3D domain with respect to its barycenter $\mathbf{x}_T$, and each face f is a star-shaped 2D domain with respect to the face's barycenter $\mathbf{x}_f$. We also assume that $\mathscr{T}$ is face-connected, i.e. it cannot be split into two meshes having no common faces.

Let $\mathbf{q}$ denote the total flux of a conservative unknown c which satisfies the conservation equation with a source term g:

$$\operatorname{div} \mathbf{q} = g \quad \text{in} \ \ \Omega. \tag{3.1}$$

We derive the cell-centered FV scheme with TPFA. Integrating 3.1 over a polyhedral cell T and using the Green's formula we get:

$$\int_{\mathrm{d}T} \mathbf{q} \cdot \mathbf{n}_T \, \mathrm{d}s = \int_T g \, \mathrm{d}x \tag{3.2}$$

where $\mathbf{n}_T$ denotes the outer unit normal to $\mathrm{d}T$. Let f denotes the face of the cell T and $\mathbf{n}_f$ be the corresponding normal vector. For a single cell T, we always assume that $\mathbf{n}_f$ is the outward normal vector. In all other cases, we specify the orientation of $\mathbf{n}_f$. It will be convenient to assume that $|\mathbf{n}_f| = |f|$ where $|f|$ denotes the area of the face f. Equation (3.2) becomes

$$\sum_{f \in \mathrm{d}T} \mathbf{q}_f \cdot \mathbf{n}_f = \int_T g \, \mathrm{d}x \tag{3.3}$$

where $\mathbf{q}_f$ is the average flux density for the face f.

For each cell T, we assign one degree of freedom, C_T, for the conservative unknown c. Let C be the vector of all degrees of freedom. If two cells T_+

and T_- have a common face f, our flux approximation with the two-point support, or the two-point flux approximation, is as follows:

$$\mathbf{q}_f^h \cdot \mathbf{n}_f = D^+ C_{T_+} - D^- C_{T_-} \tag{3.4}$$

where D^+ and D^- are some coefficients. In linear TPFA, these coefficients are equal and fixed. In nonlinear TPFA, they may be different and depend on conservative unknowns in surrounding cells. On the Neumann boundary the face $\mathbf{q}_f^h \cdot \mathbf{n}_f$ is set to the prescribed flux. Substituting (3.4) into (3.3), we obtain a system of $N_{\mathscr{T}}$ equations with $N_{\mathscr{T}}$ unknowns C_T. The cornerstone of the cell-centered FV method is the definition of discrete flux (3.4).

We restrict ourselves to the case of a continuous permeability tensor field $\mathbb{K}$. We denote by T_+ and T_- the cells that share f and assume that $\mathbf{n}_f$ is outward for T_+. Let $\mathbf{x}_\pm$ (or $\mathbf{x}_{T_\pm}$), the collocation points of $T_\pm$, coincide with barycenters of $T_\pm$. Let $C_\pm$ (or $C_{T_\pm}$) be the discrete conservative unknowns collocated in $T_\pm$.

The conventional linear TPFA reads

$$\mathbf{q}_f^h \cdot \mathbf{n}_f = \frac{\mathbb{K}\mathbf{n}_f \cdot \mathbf{t}_f}{|\mathbf{t}_f|^2}(C_{T_+} - C_{T_-}) \tag{3.5}$$

where $\mathbf{t}_f = \mathbf{x}_{T_+} - \mathbf{x}_{T_-}$. In the case of a $\mathbb{K}$-orthogonal grid $\mathbb{K}\mathbf{n}_f$ and $\mathbf{t}_f$ are colinear, and expression (3.5) takes the form of the central finite difference and approximates the flux with at least first-order accuracy. In the general case, the linear scheme may not provide approximation at all.

The detailed description of the nonlinear TPFA in the 3D case can be found in [7, 18]. Here we sketch the method presentation for interior faces and diffusive fluxes.

Let $\mathscr{F}_T$ denote the set of faces f of the polyhedral cell T. For every cell T we define a set Σ_T of nearby collocation points as follows. First, we add to Σ_T the collocation point $\mathbf{x}_T$. Then, for every interior face $f \in \mathscr{F}_T$, we add the collocation point $\mathbf{x}_{T'_f}$, where T'_f is the cell other than T, that has face f. Finally, for any boundary face $f \in \mathscr{F}_T$, we add the point $\mathbf{x}_f$ (see Fig. 1 a). Let $N(\Sigma_T)$ denote the cardinality of Σ_T.

We assume that for every cell-face pair $T \to f$, $T \in \mathscr{T}$, $f \in \mathscr{F}_T$, there exist three points $\mathbf{x}_{f,1}$, $\mathbf{x}_{f,2}$, and $\mathbf{x}_{f,3}$ in the set Σ_T such that the following condition holds (see Fig. 1 b): The co-normal vector $\ell_f = \mathbb{K}(\mathbf{x}_f)\mathbf{n}_f$ started from $\mathbf{x}_T$ belongs to the trihedral corner formed by the vectors

$$\mathbf{t}_{f,1} = \mathbf{x}_{f,1} - \mathbf{x}_T, \quad \mathbf{t}_{f,2} = \mathbf{x}_{f,2} - \mathbf{x}_T, \quad \mathbf{t}_{f,3} = \mathbf{x}_{f,3} - \mathbf{x}_T \tag{3.6}$$

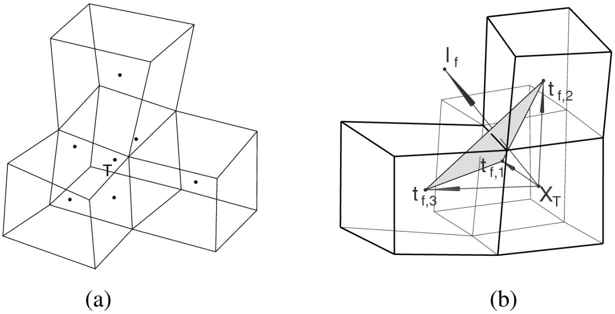

Figure 1. (a) Example of sets Σ_T. (b) Co-normal vector and vector triplet.

and

$$\frac{1}{|\ell_f|}\ell_f = \frac{\alpha_f}{|\mathbf{t}_{f,1}|}\mathbf{t}_{f,1} + \frac{\beta_f}{|\mathbf{t}_{f,2}|}\mathbf{t}_{f,2} + \frac{\gamma_f}{|\mathbf{t}_{f,3}|}\mathbf{t}_{f,3} \tag{3.7}$$

where $\alpha_f \geqslant 0$, $\beta_f \geqslant 0$, $\gamma_f \geqslant 0$. In [7] a simple and efficient algorithm for searching a triplet satisfying (3.7) with non-negative coefficients is presented.

The coefficients α_f, β_f, γ_f are computed as follows:

$$\alpha_f = \frac{A_{f,1}}{A_f}, \qquad \beta_f = \frac{A_{f,2}}{A_f}, \qquad \gamma_f = \frac{A_{f,3}}{A_f} \tag{3.8}$$

where

$$A_f = \frac{|\mathbf{t}_{f,1} \quad \mathbf{t}_{f,2} \quad \mathbf{t}_{f,3}|}{|\mathbf{t}_{f,1}||\mathbf{t}_{f,2}||\mathbf{t}_{f,3}|}, \qquad A_{f,1} = \frac{|\ell_f \quad \mathbf{t}_{f,2} \quad \mathbf{t}_{f,3}|}{|\ell_f||\mathbf{t}_{f,2}||\mathbf{t}_{f,3}|}$$

$$A_{f,2} = \frac{|\mathbf{t}_{f,1} \quad \ell_f \quad \mathbf{t}_{f,3}|}{|\mathbf{t}_{f,1}||\ell_f||\mathbf{t}_{f,3}|}, \qquad A_{f,3} = \frac{|\mathbf{t}_{f,1} \quad \mathbf{t}_{f,2} \quad \ell_f|}{|\mathbf{t}_{f,1}||\mathbf{t}_{f,2}||\ell_f|}$$

and $|\mathbf{a}\ \mathbf{b}\ \mathbf{c}| = |(\mathbf{a}\times\mathbf{b})\cdot\mathbf{c}|$.

In the case of the diffusive flux $\mathbb{K}_f \nabla c \cdot \mathbf{n}_f$, the non-negative coefficients in (3.4) are

$$D^{\pm} = \mu_{\pm}|\ell_f|(\alpha_{\pm}/|\mathbf{t}_{\pm,1}| + \beta_{\pm}/|\mathbf{t}_{\pm,2}| + \gamma_{\pm}/|\mathbf{t}_{\pm,3}|). \tag{3.9}$$

Coefficients $\mu_{\pm}$ depend on the neighbouring conservative unknowns:

$$\mu_+ = \frac{d_-}{d_- + d_+}, \qquad \mu_- = \frac{d_+}{d_- + d_+}$$

where

$$d_{\pm} = |\ell_f|\left(\frac{\alpha_{\pm}}{|\mathbf{t}_{\pm,1}|}C_{\pm,1} + \frac{\beta_{\pm}}{|\mathbf{t}_{\pm,2}|}C_{\pm,2} + \frac{\gamma_{\pm}}{|\mathbf{t}_{\pm,3}|}C_{\pm,3}\right). \tag{3.10}$$

If $d_\pm = 0$, we set $\mu_+ = \mu_- = 1/2$.

The important feature of nonlinear TPFA is that it reduces to linear TPFA on $\mathbb{K}$-orthogonal grids.

In the next section we shall need the variation of coefficients $D^\pm$ in (3.9) to calculate the Jacobian matrix. First we write variations for $d_\pm$ and $\mu_\pm$:

$$\Delta d_\pm = |\ell_f| \left(\frac{\alpha_\pm}{|\mathbf{t}_{\pm,1}|} \Delta C_{\pm,1} + \frac{\beta_\pm}{|\mathbf{t}_{\pm,2}|} \Delta C_{\pm,2} + \frac{\gamma_\pm}{|\mathbf{t}_{\pm,3}|} \Delta C_{\pm,3} \right) \tag{3.11}$$

$$\Delta \mu_\pm = \frac{\Delta d_\mp}{d_\mp + d_\pm} - (\Delta d_\mp + \Delta d_\pm) \frac{d_\mp}{(d_\mp + d_\pm)^2}. \tag{3.12}$$

Then for the variation of $D^\pm$ we have the linear combination:

$$\Delta D^\pm = \Delta \mu_\pm \big(\alpha_\pm / |\mathbf{t}_{\pm,1}| + \beta_\pm / |\mathbf{t}_{\pm,2}| + \gamma_\pm / |\mathbf{t}_{\pm,3}| \big) = \sum_{T_i \in \Sigma_{T_*}} L_i^\pm \Delta C_i \tag{3.13}$$

where $\Sigma_{T_*} := \Sigma_{T_+} \cup \Sigma_{T_-}$ and $L_i^\pm = L_i^\pm(C)$ are the coefficients of the linear combination obtained by substituting (3.11) and (3.12) into (3.9).

4. Jacobian matrix

The nonlinear TPFA (3.4)–(3.9) presumes that flux discretization introduces extra nonlinearity in the discrete system (2.1). This distinction from the conventional linear TPFA requires care in deriving the Jacobian matrix that appears in (2.4).

The construction of the Jacobian matrix is as follows. We divide the residual into two parts: accumulation (including well terms) and transport, $R_{\alpha,i} = R_{\alpha,i}^{\text{acc}} + R_{\alpha,i}^{\text{transp}}$, where:

$$R_{\alpha,i}^{\text{acc}} = V_i \left[\left(\frac{\varphi S_\alpha}{B_\alpha} \right)^{l,i} - \left(\frac{\varphi S_\alpha}{B_\alpha} \right)^{n,i} \right] - \Delta t^{n+1} Q_\alpha^{l,i}$$

$$Q_\alpha^{l,i} = \int_{T_i} q_\alpha^{l,i} \mathrm{d}x$$

$$R_{\alpha,i}^{\text{transp}} = \Delta t^{n+1} \int_{T_i} \operatorname{div} \mathbf{u}_\alpha^l \, \mathrm{d}x, \quad \alpha = w, o.$$

We omit indexes l and i. Firstly, we consider the variation of the accumulation term:

$$\Delta R_{w,i}^{\text{acc}} = V_i \Delta \left(\frac{\varphi S_w}{B_w} \right) - \Delta t^{n+1} \Delta Q_w$$

$$\Delta R^{\text{acc}}_{o,i} = V_i \Delta\left(\frac{\varphi S_o}{B_o}\right) - \Delta t^{n+1}\Delta Q_o$$

where

$$\Delta\left(\frac{\varphi S_w}{B_w}\right) = \frac{\varphi}{B_w}\Delta S_w + S_w\left(\frac{\varphi_0 c_R}{B_w} - \frac{\varphi}{B_w^2}\frac{dB_w}{dp_o}\right)\Delta p_o$$

$$\Delta\left(\frac{\varphi S_o}{B_o}\right) = -\frac{\varphi}{B_o}\Delta S_w + (1-S_w)\left(\frac{\varphi_0 c_R}{B_o} - \frac{\varphi}{B_o^2}\frac{dB_o}{dp_o}\right)\Delta p_o$$

and V_i is the volume of the grid cell T_i.

For the wells terms we introduce auxiliary variables and derivatives:

$$\mathscr{D}_\alpha = p_{bh} - p_o - \frac{\rho_{\alpha,0}}{B_\alpha} g(z_{bh} - z)$$

$$\frac{\mathrm{d}\mathscr{D}_\alpha}{\mathrm{d}p_o} = -1 + \frac{\rho_{\alpha,0}}{B_\alpha^2}\frac{\mathrm{d}B_\alpha}{\mathrm{d}p_o} g(z_{bh} - z), \qquad \frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}S_w} = \frac{\mathrm{d}k_{r\alpha}}{\mathrm{d}S_w}\frac{1}{B_\alpha \mu_\alpha}$$

$$\frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}p_o} = -k_{r\alpha}\left(B_\alpha \frac{\mathrm{d}\mu_\alpha}{\mathrm{d}p_o} + \mu_\alpha \frac{\mathrm{d}B_\alpha}{\mathrm{d}p_o}\right)/(B_\alpha \mu_\alpha)^2, \quad \alpha = w, o.$$

Then the variation of the well term is: for the producer well

$$\Delta Q_\alpha = WI\left[\mathscr{D}_\alpha \frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}S_w}\Delta S_w + \left(\lambda_\alpha \frac{\mathrm{d}\mathscr{D}_\alpha}{\mathrm{d}p_o} + \frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}p_o}\mathscr{D}_\alpha\right)\Delta p_o\right]$$

and for the injector well

$$\begin{aligned}\Delta Q_w = WI\Bigg[&\mathscr{D}_w\left(\frac{\mathrm{d}\lambda_w}{\mathrm{d}S_w} + \frac{\mathrm{d}\lambda_o}{\mathrm{d}S_w}\right)\Delta S_w \\ &+ \left((\lambda_w + \lambda_o)\frac{\mathrm{d}\mathscr{D}_w}{\mathrm{d}p_o} + \mathscr{D}_w\left(\frac{\mathrm{d}\lambda_w}{\mathrm{d}p_o} + \frac{\mathrm{d}\lambda_o}{\mathrm{d}p_o}\right)\right)\Delta p_o\Bigg] \\ \Delta Q_o = 0.&\end{aligned}$$

Secondly, we consider the transport term composed of Darcy fluxes

$$R^{\text{transp}}_{\alpha,i} = \Delta t^{n+1}\int_{\partial T_i} (\mathbf{u}_\alpha \cdot \mathbf{n})\,\mathrm{d}s \approx \Delta t^{n+1}\sum_{f\in\partial T_i} \mathbf{u}^h_{\alpha,f}\cdot \mathbf{n}_f. \tag{4.1}$$

We apply TPFA for the flux of each field: p_o, p_c, z and denote the respective flux coefficients by $D^\pm_{p_o}$, $D^\pm_{p_c}$, $D^\pm_z$ and the collocated field values at

$\mathbf{x}_{T_\pm}$ by $p_o^\pm$, $p_c^\pm$, $S_w^\pm$, $z^\pm$. In this case

$$\begin{aligned}\mathbf{u}_{w,f}^h \cdot \mathbf{n}_f = &-\left(\frac{k_{rw}}{\mu_w B_w}\right)_f \left(D_{p_o}^+ p_o^+ - D_{p_o}^- p_o^-\right)\\ &+\left(\frac{k_{rw}}{\mu_w B_w^2}\right)_f \left[\rho_{w,0}\, g\left(D_z^+ z^+ - D_z^- z^-\right)\right]\\ &+\left(\frac{k_{rw}}{\mu_w B_w}\right)_f \left(D_{p_c}^+ p_c^+ - D_{p_c}^- p_c^-\right) \end{aligned} \tag{4.2}$$

$$\begin{aligned}\mathbf{u}_{o,f}^h \cdot \mathbf{n}_f = &-\left(\frac{k_{ro}}{\mu_o B_o}\right)_f \left(D_{p_o}^+ p_o^+ - D_{p_o}^- p_o^-\right)\\ &+\left(\frac{k_{ro}}{\mu_o B_o^2}\right)_f \left[\rho_{o,0}\, g\left(D_z^+ z^+ - D_z^- z^-\right)\right]. \end{aligned} \tag{4.3}$$

Here $k_{r\alpha} = k_{r\alpha}(\widetilde{S}_w)$, $\widetilde{S}_w$ is the upwinded value of water saturation on the face f and $B_\alpha = B_\alpha(\widetilde{p}_o)$, $\mu_\alpha = \mu_\alpha(\widetilde{p}_o)$, $\widetilde{p}_o$ is the upwinded value of oil pressure on the face f.

We define the auxiliary variables and derivatives:

$$\lambda_{g,\alpha} = \frac{k_{r\alpha}}{\mu_w B_w^2}, \qquad \frac{\mathrm{d}\lambda_{g,\alpha}}{\mathrm{d}\widetilde{S}_w} = \frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}\widetilde{S}_w} / B_w$$

$$\frac{\mathrm{d}\lambda_{g,\alpha}}{\mathrm{d}\widetilde{p}_o} = \left(\frac{\mathrm{d}\lambda_\alpha}{\mathrm{d}\widetilde{p}_o} B_w - \lambda_\alpha \frac{\mathrm{d}B_w}{\mathrm{d}\widetilde{p}_o}\right) / B_w^2, \quad \alpha = w, o$$

$$\mathscr{D}_1 = D_{p_o}^+ p_o^+ - D_{p_o}^- p_o^-$$

$$\mathscr{D}_2 = D_{p_c}^+ p_c^+ - D_{p_c}^- p_c^-$$

$$\mathscr{D}_{3,\alpha} = \rho_{\alpha,0}\, g\left(D_z^+ z^+ - D_z^- z^-\right).$$

Using (4.2) and (4.3) we get the following representation for the flux variation for each of two phases:

$$\begin{aligned}
\Delta(\mathbf{u}^h_{w,f}\cdot\mathbf{n}_f) = & \left[\left(\frac{\mathrm{d}\lambda_w}{\mathrm{d}\widetilde{S}_w}\right)(-\mathscr{D}_1+\mathscr{D}_2)+\frac{\mathrm{d}\lambda_{g,w}}{\mathrm{d}\widetilde{S}_w}\mathscr{D}_{3,w}\right]\Delta\widetilde{S}_w \\
& + \left[\left(\frac{\mathrm{d}\lambda_w}{\mathrm{d}\widetilde{p}_o}\right)(-\mathscr{D}_1+\mathscr{D}_2)+\frac{\mathrm{d}\lambda_{g,w}}{\mathrm{d}\widetilde{p}_o}\mathscr{D}_{3,w}\right]\Delta\widetilde{p}_o \\
& - \lambda_w\left(D^+_{p_o}\Delta p^+_o - D^-_{p_o}\Delta p^-_o\right) \\
& + \lambda_w\left(D^+_{p_c}\left(\frac{\mathrm{d}p_c}{\mathrm{d}S_w}\right)^+\Delta S^+_w - D^-_{p_c}\left(\frac{\mathrm{d}p_c}{\mathrm{d}S_w}\right)^-\Delta S^-_w\right) \\
& - \lambda_w\left(\Delta D^+_{p_o}p^+_o - \Delta D^-_{p_o}p^-_o\right) + \lambda_w\left(\Delta D^+_{p_c}p^+_c - \Delta D^-_{p_c}p^-_c\right)
\end{aligned} \tag{4.4}$$

$$\begin{aligned}
\Delta(\mathbf{u}^h_{o,f}\cdot\mathbf{n}_f) = & \left[(\frac{\mathrm{d}\lambda_o}{\mathrm{d}\widetilde{S}_w})(-\mathscr{D}_1+\mathscr{D}_2)+\frac{\mathrm{d}\lambda_{g,o}}{\mathrm{d}\widetilde{S}_w}\mathscr{D}_{3,o}\right]\Delta\widetilde{S}_w \\
& + \left[(\frac{\mathrm{d}\lambda_o}{\mathrm{d}\widetilde{p}_o})(-\mathscr{D}_1+\mathscr{D}_2)+\frac{\mathrm{d}\lambda_{g,o}}{\mathrm{d}\widetilde{p}_o}\mathscr{D}_{3,o}\right]\Delta\widetilde{p}_o \\
& - \lambda_o\left(D^+_{p_o}\Delta p^+_o - D^-_{p_o}\Delta p^-_o\right) + \lambda_o\left(\Delta D^+_{p_o}p^+_o - \Delta D^-_{p_o}p^-_o\right).
\end{aligned} \tag{4.5}$$

One can use two possible approaches for computing the variation of the transport terms (4.4) and (4.5): coefficients $D^\pm_{p_o}, D^\pm_{p_c}, D^\pm_z$ may be assumed to be frozen for each Newton step [19] or they may be differentiated as dependent on pressure and saturation in a few neighbouring cells. In the first case $\Delta D^\pm_{p_o} = \Delta D^\pm_{p_c} = \Delta D^\pm_z = 0$ and the difference between the linear and nonlinear TPFAs is only in the way we calculate $D^\pm_{p_o}, D^\pm_{p_c}, D^\pm_z$, but not in the sparsity of the Jacobian matrix. The cost of each Jacobian-vector multiplication will remain the same for both linear and nonlinear TPFAs. If no coefficients are frozen in differentiation, then

$$\Delta D^\pm_{p_o} = \sum_{T_j\in\Sigma_{T_*}} L^\pm_{p,j}\Delta p^j_o \tag{4.6}$$

$$\Delta D^\pm_{p_c} = \sum_{T_j\in\Sigma_{T_*}} L^\pm_{p_c,j}\left(\frac{\mathrm{d}p_c}{\mathrm{d}S_w}\right)^j\Delta S^j_w \tag{4.7}$$

$$\Delta D^\pm_z = 0 \tag{4.8}$$

Table 1.
Fluid compressibility properties.

p (psia)	B_o (bbl/STB)	B_w (bbl/STB)	μ_o (cp)	μ_w (cp)
3900	1.003029	1.013174	90.58	0.515
4000	1.001967	1.012908	96.02	0.518
4100	1.000903	1.012638	101.72	0.521

where $L^{\pm}_{p_o,j}$ and $L^{\pm}_{p_c,j}$ are the coefficients calculated in (3.13) for the fields p_o and $p_c(S_w)$, respectively. This results in a more dense Jacobian matrix and more expensive Jacobian-vector multiplication and preconditioning in the linear solver. In order to reduce the arithmetic complexity, we introduce a threshold which filters off small entries of the Jacobian matrix and produces less dense matrices of linear systems to be solved.

5. Comparison of linear and nonlinear TPFAs

We consider the water flooding benchmark in its simplest version with one water injector well and one oil producer well. Injected water pushes the oil towards the producer well and fills the pores of the medium. The idea of this benchmark is to study the impact of discretizations on the water front behaviour. In particular, we compare the time curves of the oil and water production rates and the moment of water breakthrough in the producer well.

The wells are located inside the domain and the meshes are refined in the way that the centers of the connected cells are always at the same position. The injector well is located at $(-40,-40)$ and the producer well is located at (40,40). Both wells have one connection to the reservoir and are incorporated through the bottom hole pressures. For the injector it is $p_{\rm bh,inj} = 4100$ psia and for the producer $p_{\rm bh,pr} = 3900$ psia. The well indexes are calculated according to 1.6 with $r_w = 5 \cdot 10^{-4}$ ft and the skin factor $s = 0$.

In the numerical experiments we use the following rock and fluid properties. Viscosities μ_α and volume factors B_α are set by Table 1 and densities are calculated as $\rho_\alpha = \rho_{\alpha,0}/B_\alpha$, where $\rho_{w,0} \approx 4.331 \cdot 10^{-1}$ psi/ft and $\rho_{o,0} \approx 3.898 \cdot 10^{-1}$ psi/ft. The rock matrix compressibility c_R is set to $10^{-6}\,\text{psi}^{-1}$. The dependences on S_w of capillary pressure p_c and relative permeabilities $k_{r\alpha}$ are presented in Fig. 2. The dimensions of the reservoir in feet are $\Omega = [-50,50] \times [-50,50] \times [4010,4020]$.

In this section we shall demonstrate the advantages of nonlinear TPFA.

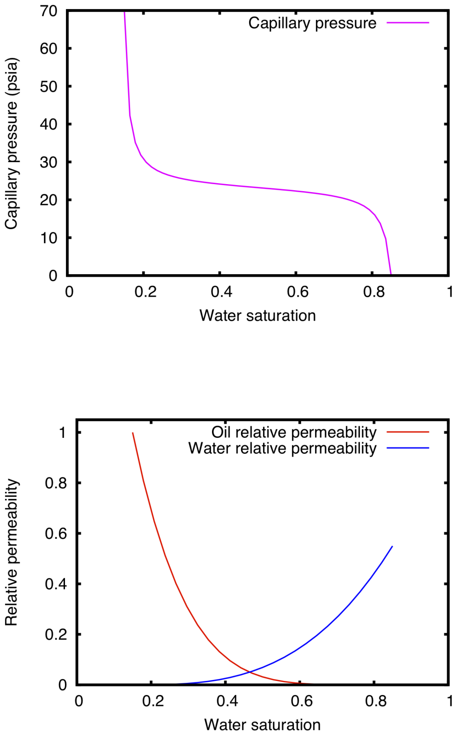

Figure 2. Dependences on S_w of capillary pressure (left) and oil and water relative permeabilities (right).

For the sake of simplicity, we exclude gravity and solve pseudo-3D problems using $N \times N \times 1$ grids with one cell layer. The experimental convergence analysis is performed on a sequence of orthogonal grids and a sequence of randomly distorted grids which are obtained by perturbation of uniform square meshes with the mesh size h. Each internal node (x,y) not adjacent to the well is relocated to a new position $(\tilde{x},\tilde{y})$ as follows:

$$\tilde{x} := x + \gamma\xi_x h, \qquad \tilde{y} := y + \gamma\xi_y h \tag{5.1}$$

where ξ_x and ξ_y are random variables with values between -0.5 and 0.5 and $\gamma \in [0,1]$ is the degree of distortion. We set $\gamma = 0.6$ to avoid mesh tangling. It is pertinent to emphasize that the distortion is performed on each refinement level. Examples of orthogonal and distorted grids are shown in Fig. 3.

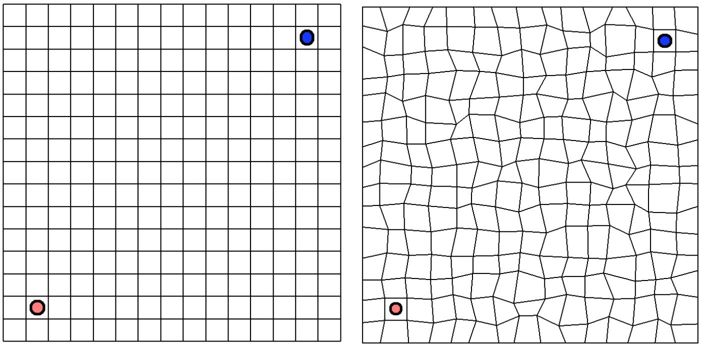

Figure 3. Examples of orthogonal grid (left) and distorted grid (right). Lower-left circle denotes the injector well, upper-right circle denotes the producer well.

In the first experiment we compare the results of simulations with linear and nonlinear TPFA on a sequence of grids which consist of $15 \times 15 \times 1$, $45 \times 45 \times 1$, $135 \times 135 \times 1$, and $405 \times 405 \times 1$ cells. The maximum time steps within the simulation are $\mathrm{d}t_{\max} = 27$, 9, 3 and 1 days, respectively. In every test we start with $dt = 0.005$ day and then calculate each next time step by the formula:

$$\alpha = \sqrt{\frac{\mathrm{d}t_{\max} - \mathrm{d}t}{\mathrm{d}t_{\max}}}, \quad \mathrm{d}t := \alpha \mathrm{d}t + (1-\alpha)\mathrm{d}t_{\max}.$$

The total duration of the simulation is 250 days.

The absolute permeability tensor is $\mathbb{K} = \mathrm{diag}(1000, 100, 50)$. Since the orthogonal grid is $\mathbb{K}$-orthogonal, the linear and nonlinear TPFAs are identical and we refer to the solution by (non)linear TPFA on the finest orthogonal grid as the reference solution.

Figure 4 (left) demonstrates the almost identical behaviour of the oil production rates computed with (non)linear TPFA on orthogonal grids and nonlinear TPFA on distorted meshes. On the other hand, the FV scheme with linear TPFA *diverges* on the sequence of *randomly distorted* grids. This is clearly observed in Fig. 4 (right) where the maxima of the solid lines shift in time and decrease their value compared to the production rate computed on orthogonal grids.

Therefore, the FV scheme with linear TPFA produces wrong results due to the loss of the approximation property, whereas the FV scheme with nonlinear TPFA demonstrates the first-order convergence of the oil production rate.

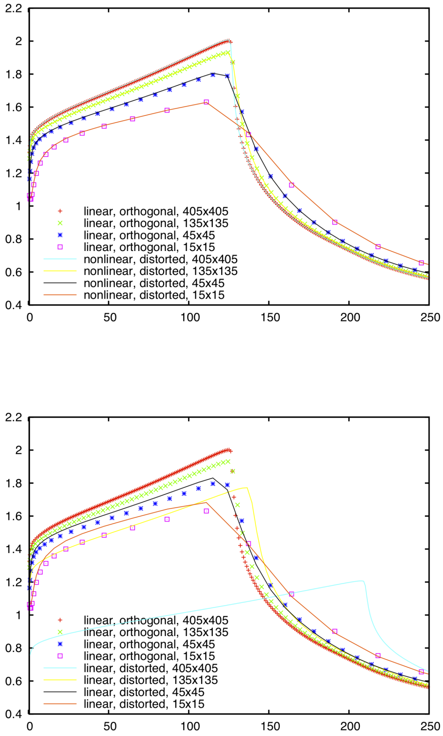

Figure 4. Oil production rates as functions of time. Left: Linear TPFA on orthogonal grids (dots) and nonlinear TPFA on distorted grids (lines). Right: Linear TPFA on orthogonal grids (dots) and distorted grids (lines).

Now we rotate the axes of the permeability tensor $\mathbb{K} = \mathrm{diag}(1000, 100, 50)$ by 45° in the Oxy plane and simulate water flooding on a randomly distorted mesh. We fix the grid $135 \times 135 \times 1$ and focus on the impact of Jacobian matrix modifications on the solver efficiency. We skip the comparison with linear TPFA, because it converges to a wrong solution. In Table 2 we present the total CPU time, total number of linear iterations (BiCGStab with ILU(1) preconditioner, reduction of the residual norm below 10^{-12}), and the CPU time per iteration for three strategies. The first one is to use (4.4)–(4.5) with full differentiation (4.6) resulting in less sparse Jacobians (denoted as true Jacobians in the table). The second strategy is to set $\Delta D_f^{\pm} = 0$ in (4.4)–(4.5) and obtain reduced Jacobians with the same sparsity as have the Ja-

Table 2.
Simulation performances for different strategies of Jacobian modifications.

Strategy	Total time	Linear iterations	Time per iteration
True Jacobian	1063.4	17274	0.062
Reduced Jacobian	15329.8	371372	0.041
Tolerance 10^{-7}	926.2	17227	0.054

cobians due to the linear TPFA. The third strategy is to filter off all entries of the true Jacobian that are below the threshold 10^{-7}. The second strategy turns to be inefficient due to the large numbers of nonlinear and linear iterations, although it has the same cost per iteration as the conventional linear TPFA. The first and the third strategies are similar in efficiency, but the latter outperforms the true Jacobian strategy by 10% due to more sparse matrices of linear systems to be solved.

6. Water flooding simulations on dynamic octrees

The dynamic octree grids are described by the following construction. We consider a coarse $M \times N \times K$ orthogonal grid, which defines the coarsest level. Each cell of this grid is converted to an octree [8]. The final grid is the forest of connected octrees. In this work we set $M = N = 5$, $K = 1$ and assume that the sizes of any two neighbouring cells in the locally refined mesh may differ at most by factor 2. The rules of the refinement and derefinement of dynamic grids are discussed below. In our discretization we treat the octree grid as a general conformal polyhedral mesh where each cell may have at most 24 neighbours and every face have at most 2 cells. In this test we do not refine cells in Oz direction.

Interpolation of functions from one octree grid to another is based on the assumption that local refinement and derefinement can produce only halving or doubling of the cell size. For interpolation we use a modification of the conservative weighted least-squares (WLSQR) reconstruction [9]. Assume that in a cell T_0 and in the neighbouring cells T_i the piecewise constant function u be given by values u_0 and u_i. Assume also that cell T_0 be split into 8 sub-cells and we have to interpolate u to these cells. We find a linear function $P(x,y,z) = ax+by+cz+d$ such that $\int_{T_0} P(x,y,z)\mathrm{d}T = |T_0|u_0$ (conservativity) and $\int_{T_i} P(x,y,z)\mathrm{d}T \approx |T_i|u_i$ (approximation). From the conservativity equation we fix d, and the approximation requirement is satisfied

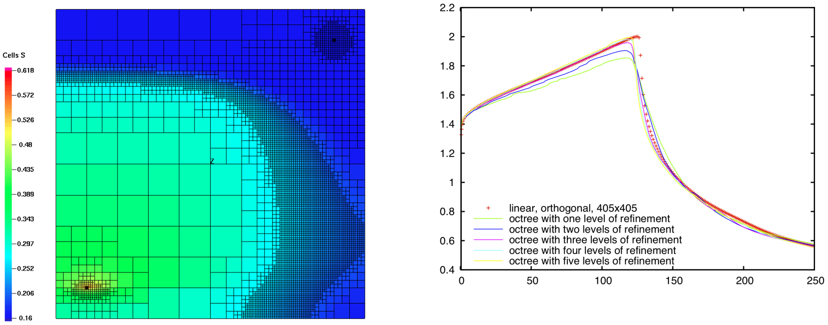

Figure 5. Left: example of the octree grid and water saturation field. Right: Oil production rate as function of time on octree grids with different levels of refinement.

by the least squares fit with variable coefficients a,b,c. The interpolated values of u are the values of $P(x,y,z)$ at the barycenters of the 8 sub-cells.

We return to the first test problem with the diagonal permeability tensor $\mathbb{K} = \text{diag}(1000, 100, 50)$. Since the orthogonal uniform grid is $\mathbb{K}$-orthogonal and nonlinear TPFA reduces to linear TPFA on $\mathbb{K}$-orthogonal grids, we can refer to the solution obtained by (non)linear TPFA on the finest $405 \times 405 \times 1$ grid with the minimal 1 day time step as the reference solution. In this section we use 1 day time step for all the meshes. Due to the anisotropy of $\mathbb{K}$ the water front moves along the x-axis faster than along the y-axis. The adaptive grid should track the water front as the region of a high gradient of water saturation. In order to reduce the numerical diffusion, we also apply local refinement in the regions of the high oil pressure gradient. More precisely, if $|\nabla S_w| > \text{tol}_{S_w}$ then the grid is refined towards the finest level l and if $|\nabla p_o| > \text{tol}_{p_o}$ then the grid is refined towards the level $l-1$ where $\text{tol}_{S_w} = 0.25$, $\text{tol}_{p_o} = 0.0005$. In addition, we require that the cell of the adaptive grid containing a well should be refined to the finest level l. This requirement produces two additional stationary local refinements towards the positions of the wells.

The left panel of Fig. 5 demonstrates the typical octree mesh and the water saturation field. The right panel of Fig. 5 shows the time curves of the oil production rates computed on the fine reference grid and the adaptive octree grids with different levels of local refinement l. One clearly observes the fast convergence of the octree solutions towards the reference solution.

We proceed to study the efficiency of using dynamic octree grids in terms of solver performance. The Newton iterations terminate as the nonlin-

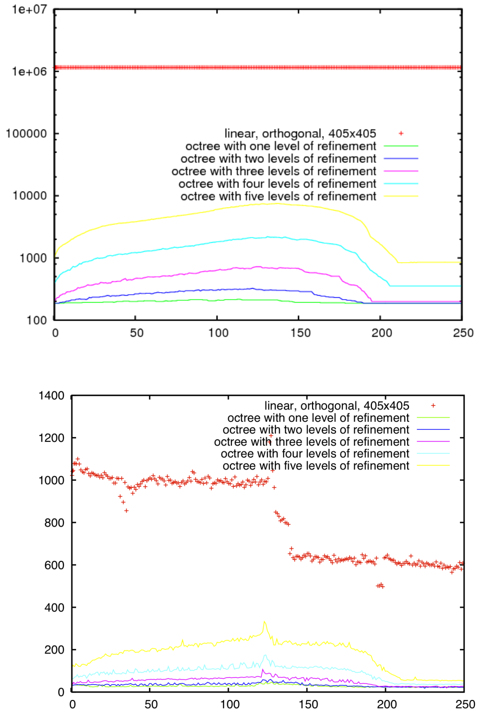

Figure 6. Numbers of cells in octree grids (left) and total numbers of linear iterations (right) at each time step on octree grids with different levels of refinement.

ear residual drops below 10^{-9} and BiCGStab ILU(1)-preconditioned iterations terminate as the residual drops below 10^{-12}. The essential difference in the numbers of unknowns for the finest reference grid and the adaptive octree grids results in much smaller numbers of BiCGStab iterations accumulated at each time step of adaptive grid simulations (see Fig. 6).

The dramatic reduction in the number of unknowns on octree grids provides considerable acceleration of the simulation (see Table 3). This acceleration is not affected by two computational overheads stemming from the use of dynamic octree grids: first, the number of non-zero entries per Jacobian row increases, second, the triplets for the nonlinear TPFA must be recomputed in the majority of mesh cells. The measurements have been carried on the Intel Xeon X5650 processor cadenced at 2.67 GHz.

In summary, water flooding simulation on adaptive dynamic octree grids provides the same accuracy with considerably smaller numbers of the de-

Table 3.
Run times of simulations on the reference uniform grid and dynamic octree grids and achieved accelerations.

Grid	CPU time, sec	T_{ref}/T_k
Reference uniform grid		
405×405	$T_{\text{ref}} = 83837$	1
Local refinement of $15 \times 15 \times 1$ grid		
1 level	$T_1 = 39.2$	2139
2 levels	$T_2 = 47.6$	1764
3 levels	$T_3 = 104$	808
4 levels	$T_4 = 361$	233
5 levels	$T_5 = 2105$	40

grees of freedom. The overhead is the need in the conservative interpolation between consecutive grids, recomputing triplets for nonlinear TPFA, and upscaling procedures for the case of heterogeneous media.

Conclusion

We have proposed an approach to simulation of two-phase water flooding with the fully implicit nonlinear finite volume method on distorted grids and dynamically adapted octree grids. We have shown that nonlinear TPFA provides compact stencils and convergent solutions independently of the grid $\mathbb{K}$-orthogonality. The criterion for octree refinement based on estimation of high gradients of S_w and p_o results in the considerable speed-up of simulation with a minimal loss in accuracy.

References

1. I. Aavatsmark, G. Eigestad, B. Mallison, and J. Nordbotten. A compact multipoint flux approximation method with improved robustness. *Numer. Meth. Part. Diff. Equ.* (2008) **24**, No. 5, 1329–1360.
2. U. Ascher, S.J. Ruuth, and T.R. Wetton. Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Numer. Anal.* (1995) **32**, No. 3, 797–823.
3. K. Aziz and A. Settari. *Petroleum Reservoir Simulation.* Applied Sci. Publ. Ltd, London, 1979.
4. I. Babuska and W.C. Rheinboldt. A posteriori error analysis of finite element solutions of one dimensional problems. *SIAM J. Numer. Anal.* (1981) **18**, 565–589.

5. M. B. Bieterman and I. Babuska. The finite element method for parabolic equations, I: A posteriori error estimation, II: A posteriori error estimation and adaptive approach. *Numerische Math.* (1982) **40**, 339–371 and 373–406.

6. B. N. Chetverushkin, N. G. Churbanova, M. A. Trapeznikova, A. A. Sukhinov, and A. A. Malinovskij. Adaptive cartesian mesh refinement for simulating multiphase flows in porous media. *Comp. Meth. Appl. Math.* (2008) **8** No. 2, 101–115.

7. A. Danilov and Yu. Vassilevski. A monotone nonlinear finite volume method for diffusion equations on conformal polyhedral meshes. *Russ. J. Numer. Anal. Math. Modelling* (2009) **24**, No. 3, 207–227.

8. M. Donald, Octree encoding: A new technique for the representation, manipulation and display of arbitrary 3-d objects by computer. *Tech. Report IPL-TR-80-111*, Rensselaer Polytechnic Institute, 1980.

9. J. Furst, A weighted least square scheme for compressible flows. *Flow, Turbul. Combustion* (2006) **76**, No. 4, 331–342.

10. I. Kapyrin. A family of monotone methods for the numerical solution of three-dimensional diffusion problems on unstructured tetrahedral meshes. *Dokl. Math.* (2007) **76**, No. 2, 734–738.

11. R. A. Klause and R. Winther. Convergence of multipoint flux approximations on quadrilateral grids. *Numer. Meth. Part. Differ. Equ.* (2006) **22**, 1438–1454.

12. K. Lipnikov, D. Svyatskiy, M. Shashkov, and Yu. Vassilevski. Monotone finite volume schemes for diffusion equations on unstructured triangular ans shape-regular polygonal meshes. *J. Comp. Phys.* (2007) **227**, 492–512.

13. K. Lipnikov, D. Svyatskiy, and Yu. Vassilevski. Interpolation-free monotone finite volume method for diffusion equations on polygonal meshes. *J. Comp. Phys.* (2009) **228**, No. 3, 703–716.

14. K. Lipnikov, D. Svyatskiy, and Yu. Vassilevski. A monotone finite volume method for advection-diffusion equations on unstructured polygonal meshes. *J. Comp. Phys.* (2010) **229**, 4017–4032.

15. K. Lipnikov, D. Svyatskiy, and Yu. Vassilevski. Minimal stencil finite volume scheme with the discrete maximum principle. *Russ. J. Numer. Anal. Math. Modelling* (2012) **27**, No. 4, 369–385.

16. K. Nikitin. Nonlinear finite volume method for multiphase flow model. *Math. Modelling* (2010) **22**, No. 11, 131–147 (in Russian).

17. K. Nikitin, K. Terekhov, and Yu. Vassilevski. A monotone nonlinear finite volume method for diffusion equations and multiphase flows. *Comp. Geosci.* (2013) (submitted).

18. K. Nikitin and Yu. Vassilevski. A monotone nonlinear finite volume method for advection-diffusion equations on unstructured polyhedral meshes in 3D. *Russ. J. Numer. Anal. Math. Modelling* (2010) **25**, No. 4, 335–358.

19. K. Nikitin and Yu. Vassilevski. A monotone finite volume method for advection diffu-

sion equations and multiphase flows. In: *Proc. ECMOR XIII, Biarritz, France*. EAGE, 2012.

20. J. M. Nordbotten, I. Aavatsmark, and G. T. Eigestad. Monotonicity of control volume methods. *Numer. Math.* (2007) **106**, No. 2, 255–288.

21. D. W. Peaceman. *Fundamentals of Numerical Reservoir Simulation*. Elsevier, New York, 1977.

22. D. W. Peaceman. Interpretation of well-block pressures in numerical reservoir simulation. *SPEJ* (1978) **253**, No. 2, 183–194.

23. C. Le Potier. Schema volumes finis monotone pour des operateurs de diffusion fortement anisotropes sur des maillages de triangle non structures. *C. R. Math. Acad. Sci. Paris* (2005) **341**, 787–792.

24. M. Saad and H. Zhang. Front tacking for two-phase flow in reservoir simulation by adaptive mesh. *Numer. Meth. Part. Differ. Equ.* (1997) **13**, No. 6, 673–697.

25. J. W. Sheldon, B. Zondek, and W. T. Cardwell. One-dimensional incompressible, noncapillary, two-phase fluid flow in a porous medium. *SPE AIME* (1959) **216**, 290–296.

26. Z. Sheng and A. Yuan. Monotone finite volume schemes for diffusion equations on polygonal meshes. *J. Comp. Phys.* (2008) **227**, 6288–6312.

27. H.L. Stone and Jr. Garder. Analysis of gas-cap or dissolved-gas reservoirs. *T. SPE AIME* (1961) **222**, 92–104.

28. Yu. Vassilevski and I. Kapyrin. Two splitting schemes for nonstationary convection-diffusion problems on tetrahedral meshes. *Comp. Math. Math. Phys.* (2008) **48**, No. 8, 1349–1366.