Simulation of Flow and Dispersion on Pore-Space Images
Peyman Mostaghimi, SPE, Branko Bijeljic, SPE, and Martin J. Blunt, SPE, Imperial College London

Abstract
We simulate flow and transport directly on pore-space images obtained by micro-CT scanning of rock cores. An efficient Stokes solver is used to simulate low-Reynolds number flows. The flow simulator uses a finite-difference method along with a standard predictor-corrector procedure to decouple pressure and velocity. An algebraic multigrid technique solves the linear systems of equations. We then predict permeability and the results are compared with lattice Boltzmann numerical results and available experimental data.

For solute transport we apply a streamline-based algorithm that is similar to the Pollock algorithm common in field-scale reservoir simulation, but which employs a novel semi-analytic formulation near solid boundaries to capture, with sub-grid resolution, the variation in velocity near the grains. A random walk method accounts for molecular diffusion. The streamline-based algorithm is validated by comparison with published results for Taylor-Aris dispersion in a single capillary with a square cross-section. We then accurately predict available experimental data in the literature for longitudinal dispersion coefficient as a function of Peclet number. We introduce a characteristic length based on ratio of volume to pore/grain surface area that can be used for consolidated porous media to calculate Peclet number.

Introduction
Accurate modeling of solute dispersion at porous media is of great importance in many branches of science and engineering (Bear, 1988, Adler, 1992). To model the flow and transport at the pore scale to predict dispersion coefficients, the morphology of the pore space needs to be known. To obtain three-dimensional models of the pore space different methods can be applied. It is routine to obtain two-dimensional high-resolution images of rock (Zinszner and Meynot, 1982, Thovert et al., 1993) from which a three-dimensional representation is constructed. This can be achieved using object-based methods, where grains of different size and shape are deposited (Øren et al., 1998) or an image is obtained that reproduces the statistics of the two-dimensional pictures (Adler et al., 1990, Hazlett, 1997, Roberts, 1997, Okabe and Blunt, 2004). The advent of X-ray computed tomography has made it possible to obtain three-dimensional images with a resolution of a few microns (Coenen et al., 2004, Flannery et al., 1987) which is sufficient to capture the pore space of most sandstones.

The next step is modeling flow and transport through the pore space. Traditionally, network modeling has been applied to idealize the system as a lattice of wide pores connected by throats through which displacement and transport can be computed semi-analytically (Fatt, 1956, Blunt, 2001). Network models have been used widely to simulate dispersion at the pore scale (Bruderer and Bernabé, 2001, de Arcangelis et al., 1986, Sahimi et al., 1986, Sorbie and Clifford, 1991, Sahimi, 1995, Bijeljic et al., 2004, Bijeljic and Blunt, 2007, Acharya et al., 2007, Jha et al., 2008). However, a more direct approach is to simulate transport directly on a pore-space image that removes the need to extract an equivalent network with its inherent approximations (Maier et al., 2003, Coelho et al., 1997, Maier et al., 1998, Yao et al., 1997, Salles et al., 1993). Coelho et al. (1997) used a finite difference technique to solve for flow and simulated dispersion through packings of grains of arbitrary shape and found a good agreement with experimental results on unconsolidated bead packs and sandstones. Yao et al. (1997) used a similar method to solve for the flow field and modeled dispersion on a statistically reconstructed geometry of Vosges sandstone. The agreement of the model prediction with experimental measurements for dispersion coefficient was not satisfactory (Adler and Thovert, 1998). Maier et al. (1998) used the lattice Boltzmann technique to model dispersion in a three-dimensional model of a sphere packing and successfully compared their numerical results with NMR experiments for a wide range of dimensionless Peclet number. Lattice
Boltzmann simulation has also been used in dispersion studies in cylinder packs, sphere packs, and uniform porous media by Zhang and Lv (2007), Maier et al. (2007), Hlushkou (2007) and Khirevich et al. (2010). Garmeh et al. (2007) used a finite-element scheme to solve for fluid flow and once the velocities were calculated, they solved the convection-diffusion equation to model dispersion in two and three dimensional packings of spheres.

These previous studies have been able to reproduce the rich behavior of the dispersion coefficient as a function of Peclet number, demonstrating how the subtle interplay of advection and diffusion in a disordered pore space affects transport. However, the models are restricted to reconstructed or simple media or network models that may miss some details of the pore structure. These limitations may be important when exploring possible extensions of the work to embrace reactive transport or multiphase flow.

We simulate the flow and transport directly on pore-space images obtained from micro-CT scanning by solving the Stokes equations in the pore space and by applying a novel streamline-based algorithm which employs a new semi-analytic formulation near solid boundaries to capture, with sub-grid resolution, the variation in velocity near the solid. A random walk method accounts for diffusion. We successfully predict different dispersion regimes in laminar flow and our results show a very good agreement with previous numerical and experimental studies. This is a first step towards the simulation of reactive transport, multiphase flow and transport in complex media, such as carbonates.

Method

Our algorithm to model the dispersion of non-reactive particles is

1. Solve for Stokes flow in the pore space.
2. Inject particles.
3. In each time step:
   3.1. Move particles by advection using streamline tracing.
   3.2. Move particles by diffusion using the random-walk method.
4. Compute dispersion coefficients from the overall movement of an ensemble of particles.

For step 1, we consider each void voxel in an image obtained by micro-CT scanning as a grid block in our algorithm for simulating fluid flow. The Stokes equations combined with the continuity equation are used to solve for incompressible steady viscous flow.

\[ \nabla \cdot \mathbf{v} = 0 \]  
\[ \mu \nabla^2 \mathbf{v} = \nabla P \]  
\[ \mathbf{v} = 0 \text{ on grains} \]

where \( \mathbf{v} = (u, v, w) \) is the vector of velocity, \( P \) is pressure, and \( \mu \) is the fluid viscosity.

For discretizing the governing equations in three dimensions, we have applied the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE), pioneered by Patankar and Spalding (1972) that has been widely applied to the fields of computational fluid dynamics and numerical heat transfer. This method converts the continuity equation into a numerical equation to correct for pressure, resulting in an iterative solution. We use structured marker-and-cell (MAC) gridding (Peyret and Taylor, 1985, Harlow and Welch, 1965). MAC gridding means that the nodes where different variables are being estimated are not the same: pressures are determined at the center of cells and normal velocities are defined at cell faces. Because of the MAC gridding system, the velocity component parallel to a solid surface is not automatically located at the surface of solid voxels. We have applied a non-centered higher-order finite-difference scheme to rewrite the formulation for cells in the neighborhood of solid voxels. In this technique, we put the no-slip condition exactly on the solid surface. To solve the resultant linear systems of equations, an algebraic multigrid technique has been used (Stüben, 2001, Hackbusch, 1985).

In step 2, 3,000 particles are injected uniformly in the pore space of the inlet face. We also performed some computations with 10,000 particles. The difference in predicted dispersion coefficient was typically 3% and always less than 10%.

In step 3, we move particles in each time step due to advection and diffusion. For advection, step 3.1, particles are moved along streamlines that follow the velocity field. Pollock (1988) suggested a linear interpolation between the normal velocities on opposite faces of a grid block. This is consistent with a divergence-free velocity field and the finite difference approximation used to compute pressure and flow, as well as ensuring continuity of normal velocity. This method is now standard in field-scale streamline-based reservoir simulation (Batycky et al., 1997). For voxels with no solid boundaries we use the same method here, which is – again – consistent with incompressible flow and the finite difference approximations used. However, at a solid surface we need to impose a zero normal and tangential velocity, which is not consistent with the Pollock algorithm that would only ensure that the velocity normal to the solid is zero. To overcome this limitation and to impose a strictly vanishing velocity at the solid we have introduced a novel semi-analytic formulation for the streamlines in voxels containing a solid boundary. The details of the
formulation – given in the Appendix – vary with the location and numbers of solid faces in a voxel, and give a tangential velocity that varies quadratically away from a boundary. The result is a method that strictly obeys the boundary conditions and allows a semi-analytic tracing of streamlines throughout the domain.

We can calculate the time needed for a particle inside the voxel to exit, \( \Delta \tau \). The time step is \( \Delta t \). If \( \Delta \tau > \Delta t \) the particle remains within the voxel, whereas for \( \Delta \tau < \Delta t \) the particle enters a new block. We reduce the time step by \( \Delta t \) and find the \( \Delta \tau \) for the new grid block and repeat the previous procedure.

In step 3.2, we model movement of particles due to diffusion using a random walk method which is an established Lagrangian technique to model conservative and reactive transport in porous media (Ahlstrom et al., 1977, Tompson, 1993, Maier et al., 1998). A particle instantaneously jumps over a fixed distance \( \zeta \):

\[
\zeta = \sqrt{6D \Delta t}
\]  

(4)

The coordinates of the particle after diffusion is:

\[
x = x_p + \zeta \sin \phi \cos \theta
\]

(5)

\[
y = y_p + \zeta \sin \phi \sin \theta
\]

(6)

\[
z = z_p + \zeta \cos \phi
\]

(7)

\( \theta \) is a random number in the range of \((0, 2\pi)\) and \( \phi \) is another random number in the range of \((0, \pi)\).

We use a reflecting boundary condition if the random jump places a particle in the solid. We allow multiple reflections if the particle encounters more than one solid surface.

If a particle exits the system during its advection motion, it will be re-injected at the inlet face at a flow-weighted random location; if the particle leaves during the diffusive step, it will be re-injected by an area-weighted random position on the inlet face. In addition, if due to diffusion, a particle exits through the inlet, it will be injected on the outlet face, again assigning a random area-weighted position.

In step 4, the longitudinal dispersion coefficient is estimated by calculating the variance of the distance traveled by particles in the main direction of flow:

\[
D_L = \frac{1}{2} \frac{d\sigma^2}{dt}
\]

(8)

where \( \sigma \) is the variance of the particle displacement:

\[
\sigma = \langle [x(t) - \langle x(t) \rangle]^2 \rangle
\]

(9)

The time step for the simulation is \(10^{-4}\) s. This value has been chosen to restrict the motion of particle at each time step be less than one voxel. The molecular diffusion coefficient in our simulations is \(10^{-9}\) m\(^2\) s\(^{-1}\). Therefore at each time step the size of the diffusive step, Eq. (4), is 0.77 \(\mu\)m which is less than the size of each voxel (5.345 \(\mu\)m). To study the effect of Peclet number, we vary the flow rate. We run the simulations until we reach an asymptotic regime where \(D_L\) computed by Eq. (8) no longer changes with time.

Validation

We first considered a very simple case: flow between two parallel plates. Fig. 1 shows a comparison of the computed velocity for different numbers of grid blocks across the opening compared to the analytical solution (White, 1999). For two or more cells between the plates, the face velocities are found to within machine error. We see the same for flow through a cylinder of square cross-section. This result suggests that we can obtain a reasonable representation of the flow field even when the flow channels are represented with only a few voxels in the micro-CT image.

Furthermore, we modeled three-dimensional flow around a cubic obstacle. The model contains 50x50x50 voxels and the cubic obstacle is located at the center of the opening with dimension of 15x40x40 voxels. We compared our results against the free open source CFD toolbox, OpenFOAM (www.openfoam.com). The difference in average estimated velocity is less than 0.1% between the two codes and the local velocities differ by no more than 0.5%. In this, and all subsequent simulations, at each iteration, we calculate the volumetric flow rate passing through each layer of the system normal to main direction of flow. We iterate for velocity and pressure, until the difference between flows of each pair of layers normalized by average flow rate of whole system is less than five percent.
We then modeled single-phase flow on a series of sand packs and sandstone micro-CT images. All the images are 300×300×300 voxels. The predicted permeabilities are in good agreement with published lattice-Boltzmann results (Dong and Blunt, 2009), see Table 1.

<table>
<thead>
<tr>
<th>Resolution (μm)</th>
<th>Porosity (%)</th>
<th>LBM permeability (D)</th>
<th>Predicted permeability (D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sandpack LV60A</td>
<td>10.0</td>
<td>38</td>
<td>35</td>
</tr>
<tr>
<td>Sandpack LV60B</td>
<td>8.6</td>
<td>37</td>
<td>31</td>
</tr>
<tr>
<td>Sandpack LV60C</td>
<td>10.0</td>
<td>37</td>
<td>19</td>
</tr>
<tr>
<td>Sandpack F42A</td>
<td>10.0</td>
<td>33</td>
<td>59</td>
</tr>
<tr>
<td>Sandpack F42B</td>
<td>10.0</td>
<td>33</td>
<td>52</td>
</tr>
<tr>
<td>Sandpack F42C</td>
<td>10.0</td>
<td>33</td>
<td>50</td>
</tr>
<tr>
<td>Sandstone 1</td>
<td>8.7</td>
<td>20</td>
<td>1.7</td>
</tr>
<tr>
<td>Sandstone 2</td>
<td>5.0</td>
<td>25</td>
<td>3.9</td>
</tr>
<tr>
<td>Sandstone 3</td>
<td>9.1</td>
<td>17</td>
<td>0.22</td>
</tr>
<tr>
<td>Sandstone 4</td>
<td>9.0</td>
<td>17</td>
<td>0.26</td>
</tr>
<tr>
<td>Berea Sandstone</td>
<td>5.3</td>
<td>20</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Experimental measurements of permeability, measured on larger core samples, are also available in some cases (Talabi and Blunt, 2010). The LV60 and F42 sand packs have permeabilities of 32.2 ± 0.3 D and 42 ± 4 D respectively which are in reasonably good agreement with the average of our numerical predictions on three different images. The core-measured permeability of Berea is 0.7 D, which is half the computed value. This is likely to be due to heterogeneity and the large size difference between the micro-CT sample that is only 1.5 mm across and the core sample with a length of 7 cm. The Berea image was used for subsequent analysis to compute dispersion coefficient.

To validate our particle tracking algorithm, we simulated dispersion in a single capillary with square cross section and compared it with the previous study by Bruderer and Bernabé (2001) and Bijeljic et al. (2004). The square opening was 50×50 grid blocks across. The characteristic length used in calculating the Peclet number is the half-length of the side of the square. Aris (1956) showed that the longitudinal dispersion coefficient in a single capillary is proportional to the square of the Peclet number:

$$\frac{D_L}{D_m} = \kappa Pe^2$$  (10)

where κ is a constant coefficient that depends on the shape of the capillary. Bruderer and Bernabé (2001) found that for capillary with square cross section, $\frac{1}{\kappa} = 30.4 \pm 2.6$ and Bijeljic et al (2004) obtained $\frac{1}{\kappa} = 29.24$.

The best fit to our results shows $\frac{1}{\kappa} = 28.93$ which is in good agreement with previous studies.
Results
Flow was computed on the Berea sandstone image. Fig. 2 shows the geometry, pressure distribution and velocity in the model. 

Figure 2. The geometry – gray shows the pore volume with scale bar in millimeter unit (a), normalized pressure field with a unit pressure difference across the model (b), normalized flow field (c) for the Berea sandstone image.

Once the velocity field was obtained, we have solved for transport over a range of Peclet number, defined as:

\[ Pe = \frac{u_{\text{avg}} L}{D_m} \]  

(11)

where \( u_{\text{avg}} \) is the average velocity and \( L \) is the characteristic length. Average velocity is calculated from:
\[ u_{\text{avg}} = \frac{Q}{A\phi} \]  

(12)

where \( Q \) is volumetric flow rate in the whole model, \( \phi \) is porosity, and \( A \) is the area perpendicular to the main direction of flow (the area of the inlet face).

For experiments on unconsolidated porous media, the average diameter of grains can be used as the characteristic length; however, this is not available directly from a micro-CT image of a consolidated rock. We defined the characteristic length based on a cubic packing of regular spheres. For this idealized system, the grain diameter is \( \pi V/S \), where \( V \) is the volume of the porous medium (pore plus grain) and \( S \) is the area of the pore/grain interface. We use the same definition for our images, since the volume and pore/grain area are readily computed:

\[ L = \frac{\pi V}{S} \]  

(13)

We obtained \( L = 131.13 \mu m \) for Berea.

Fig. 3 shows our numerical results for longitudinal dispersion coefficient as function of Peclet number along with published experimental data on unconsolidated bead packs obtained from breakthrough curves (Pfannkuch, 1963), data on packs of plastic beads obtained by Pulsed-field-gradient NMR (Ding and Candela, 1996), data on bead packs obtained by magnetic resonance imaging (Seymour and Callaghan, 1997, Kandhai et al., 2002, Khrapitchev and Callaghan, 2003), data on a packed column of activated carbon grains with a bimodal pore size distribution based on radioactive dispersion (Drazer et al., 1999), data bead packs obtained by planar laser-induced fluorescence method (Stöhr, 2003), data on sandy soil column measured by electrical resistance between rod electrodes and fitting analytical models of advection-dispersion equation to concentration breakthrough curves at axial positions along the column (Aggelopoulos and Tsakiroglou, 2007), and data on a planar glass-etched pore network obtained using an experimental set-up detecting color changes caused during the mixing (Theodoropoulou, 2007).
Figure 3. Reduced dispersion coefficient (ratio of longitudinal dispersion to molecular diffusion coefficient) against dimensionless Peclet number. Different characteristic lengths have been used for estimation of Peclet number in the numerical and experimental results. The vertical bars show results for sandstone samples in the restricted diffusion regime at low Peclet number (Dullien, 1992, Frosch et al., 2000).

The agreement between experiment and the numerical results is good, although there is a tendency for the predicted curve to be shifted to slightly larger Pe, indicating, perhaps, the difference in definition of characteristic length, particularly at high Pe. The reduced dispersion coefficient is less than unity in the low Peclet regime because of the presence of the solid, restricting diffusion: our numerical results here correspond to measurements on sandstones of similar porosity, but underestimate the dispersion for bead packs. At higher Pe we see a transition regime followed by an approximate power-law regime with $D_L \sim Pe^\delta$ with an exponent $\delta = 1.2$, consistent with measurements and other analyses (Brigham et al., 1961, Salter and Mohanty, 1982, Kinzel and Hill, 1989, Bijeljic et al., 2004).

The mechanical dispersion regime can be observed for $Pe > 600$ where $D_L \sim Pe$. In this regime, the effects of molecular diffusion can be ignored and transport is dominated by advection only. We underestimate the dispersion coefficient in this regime. This could be due to inertial and turbulent effects which have been ignored in our flow simulation (Sahimi, 1995).

Discussion and Conclusions
We have described a method to simulate flow and transport in pore-space images. A finite-difference method is used to compute Stokes flow: we show that for pores of constant cross-section, the velocity is obtained to machine accuracy if there are two or more voxels across the opening. This is important for practical application, where there is a trade-off between image resolution and the size of the sample, meaning that many pores are only resolved with a few voxels. We compute the permeability for a series of images and good agreement is obtained with experimental data and predictions using the lattice-Boltzmann technique.
We have developed a new streamline-tracing technique to model advection. For voxels with no solid boundary, the method is identical to the Pollock (1988) algorithm that is now standard in field-scale streamline-based reservoir simulators. Where a solid boundary is present, zero tangential and normal velocity is strictly imposed, resulting in a quadratic variation of tangential velocity away from the boundary. A semi-analytic description of the velocity field within a grid block is obtained for all combinations of solid boundaries. An ensemble of particles is moved along streamlines, while a random walk method is used to capture molecular diffusion.

We predict asymptotic longitudinal diffusion coefficient as a function of Peclet number. We introduce a new characteristic length based on the ratio of volume to pore/grain surface area, applicable to images where a typical grain size cannot be easily obtained. Our results compare well with measurements in the literature and show the different regimes, including restricted dispersion, a power-law dependence of dispersion coefficient on Peclet number and the mechanical dispersion regime at high Peclet number.

This work serves as a basis for further analysis of dispersion on a variety of different porous media. We plan to predict longitudinal and transverse dispersion for different rock types, including pre-asymptotic behavior where the apparent dispersivity evolves over time. Our methodology is also a useful first step in the development of a rigorous pore-scale treatment of geochemical reactions.

Appendix – Streamline Tracing Algorithm

We use the method of (Pollock, 1988) in voxels with no solid boundaries. The velocity normal to a face varies linearly across the block – see Fig. 4.

\[
\begin{align*}
    u &= \frac{u_2 - u_1}{\Delta x} (x - x_1) + u_i \\
    v &= \frac{v_2 - v_1}{\Delta y} (y - y_1) + v_i \\
    w &= \frac{w_2 - w_1}{\Delta z} (z - z_1) + w_i
    \end{align*}
\]

where \( u, v \) and \( w \) are the components of velocity in the x, y and z directions respectively. \( u_i \) and \( u_2 \) are the computed velocities on the faces normal to the x direction, \( v_1 \) and \( v_2 \) are velocities on the faces normal to the y direction, while \( w_1 \) and \( w_2 \) are the velocities on the faces normal to the z direction, \( \Delta x, \Delta y \) and \( \Delta z \) are the dimensions of the cell in the respective coordinate directions – see Fig. 4. The time necessary for a particle to exit the x-face is \( \Delta \tau_x \) and the time to exit y-face and z-face are \( \Delta \tau_y \) and \( \Delta \tau_z \) respectively.

\[
\begin{align*}
    \Delta \tau_x &= \frac{\Delta x}{u_2 - u_1} \ln \frac{u_1 \Delta x}{u_2 - u_1} + (u_2 - u_1)(x_p - x_1) \\
    \Delta \tau_y &= \frac{\Delta y}{v_2 - v_1} \ln \frac{v_1 \Delta y}{v_2 - v_1} + (v_2 - v_1)(y_p - y_1) \\
    \Delta \tau_z &= \frac{\Delta z}{w_2 - w_1} \ln \frac{w_1 \Delta z}{w_2 - w_1} + (w_2 - w_1)(z_p - z_1)
    \end{align*}
\]

where \((x_p, y_p, z_p)\) are the initial coordinates of the particle. The particle leaves the grid block with the smallest value among the computed increments in time of flight.

\[
\Delta \tau = \min(\Delta \tau_x, \Delta \tau_y, \Delta \tau_z)
\]

And the exit location can be obtained from equations below:

\[
x_x = x_1 + \frac{u_i \Delta x}{u_2 - u_1} + \left[ \frac{u_i \Delta x}{u_2 - u_1} + (x_p - x_1) \right] e^{\frac{\Delta \tau - \Delta \tau_x}{\Delta x}}
\]
\[
y_r = y_1 - \frac{v_1 \Delta y}{v_2 - v_1} + \left[ \frac{v_1 \Delta y}{v_2 - v_1} + (y_p - y_1) \right] e^{\frac{-v_1 \Delta y}{v_2 - v_1}}
\]

\[
z_r = z_1 - \frac{w_1 \Delta z}{w_2 - w_1} + \left[ \frac{w_1 \Delta z}{w_2 - w_1} + (z_p - z_1) \right] e^{\frac{-w_1 \Delta z}{w_2 - w_1}}
\]

Figure 4. Velocity interpolation if there is no bounding solid voxel. This is the Pollock (1988) algorithm, standard in field-scale streamline-based reservoir simulation.

This method is used if no face is solid. In other cases, we categorize different cases where one or more nearest neighbor voxel is solid. In all cases the velocity field everywhere is divergence free and obeys a zero velocity at all solid boundaries.

1. One of the neighboring voxels is solid

This condition can happen in six different ways depending on which face is solid. We show the formulation for one of the cases; the formulation for the others will be similar. If the solid voxel blocks the \( x_2 \)–face (Fig. 5), the velocity distribution is:

\[
u = \frac{u_1}{\Delta x} (x_2 - x)^2
\]

\[
v = \frac{2 \nu_1}{\Delta x} (x_2 - x) + \frac{2(v_2 - v_1)}{\Delta x \Delta y} (x_2 - x)(y - y_1)
\]

\[
w = \frac{2 \nu_1}{\Delta x} (x_2 - x) + \frac{2(w_2 - w_1)}{\Delta x \Delta z} (x_2 - x)(z - z_1)
\]

The time of flight increment across the block is:

\[
\Delta \tau_x = \frac{\Delta x^2}{u_1} \left( 1 - \frac{1}{x_2 - x_p} \right)
\]

\[
\Delta \tau_y = \frac{\Delta x^2}{u_1(x_2 - x_p)} \left( \nu_1 \Delta y + \frac{v_2 \Delta y}{2 \Delta x (v_2 - v_1)} \right) - \frac{\Delta x^2}{u_1(x_2 - x_p)}
\]
\[
\Delta \tau_z = \frac{\Delta x^2}{u_1(x_2 - x_p)} \left( \frac{w_2 \Delta z}{w_1 \Delta z + (w_2 - w_1)(z_p - z_1)} \right) \frac{u_1 \Delta z}{u_1(x_2 - x_p)} = \frac{\Delta x^2}{u_1(x_2 - x_p)}
\]

\[
\Delta \tau = \min(\Delta \tau_x, \Delta \tau_y, \Delta \tau_z)
\]

And the exit location is:

\[
x_e = x_2 - \left( \frac{\Delta \tau}{\Delta x^2} u_1 + \frac{1}{x_2 - x_p} \right)^{-1}
\]

\[
y_e = y_1 - \frac{v_1 \Delta y}{v_2 - v_1} + \frac{v_1 \Delta y + (v_2 - v_1)(y_p - y_1)}{(v_2 - v_1) \Delta y} (1 + \frac{u_1(x_2 - x_p)}{\Delta x^2} \Delta \tau) \frac{2\Delta(x_2 - x_p)}{u_0 \Delta y}
\]

\[
z_e = z_1 - \frac{w_1 \Delta z}{w_2 - w_1} + \frac{w_1 \Delta z + (w_2 - w_1)(z_p - z_1)}{(w_2 - w_1) \Delta z} (1 + \frac{u_1(x_2 - x_p)}{\Delta x^2} \Delta \tau) \frac{2\Delta(x_2 - x_p)}{u_0 \Delta z}
\]

2. Two of the neighboring voxels are solid

This condition can happen for fifteen cases that can be categorized in two groups: (1) when the two solid voxels block opposing faces (this can happen in three cases, for the x, y- or z-directions); and (2) when the solid voxels are located on adjoining faces that can happen in twelve cases.

2.1. There are two opposing solid faces

We show the equations where both solid blocks normal to the x-direction (Fig. 6); formulations for the two other cases can be obtained in the same way. Since both x-faces are blocked, the velocity component in the x direction is zero. The velocity distribution is:

\[
u = 0
\]

\[
v = \frac{6w_1}{\Delta x^2} (x_2 - x)(x - x_1) + \frac{6(v_2 - v_1)}{\Delta x^2} (x_2 - x)(x - x_1)(y - y_1)
\]

\[
w = \frac{6w_1}{\Delta x^2} (x_2 - x)(x - x_1) + \frac{6(w_2 - w_1)}{\Delta x^2} (x_2 - x)(x - x_1)(z - z_1)
\]

The time of flight increment across the block is:
\[
\Delta \tau_y = \frac{\Delta x^2 \Delta y}{6(v_2 - v_1)(x_2 - x_p)(x_p - x_1)} \ln\left(\frac{v_2 \Delta y}{v_1 \Delta y + (v_2 - v_1)(y - y_1)}\right)
\]

\[(37)\]

\[
\Delta \tau_z = \frac{\Delta x^2 \Delta z}{6(v_2 - v_1)(x_2 - x_p)(x_p - x_1)} \ln\left(\frac{w_2 \Delta z}{w_1 \Delta z + (w_2 - w_1)(z - z_1)}\right)
\]

\[(38)\]

\[
\Delta \tau = \min(\Delta \tau_y, \Delta \tau_z)
\]

Having the time of flight, the exit location is:

\[
x_e = x_p
\]

\[(40)\]

\[
y_e = y_1 - \frac{v_1 \Delta y}{v_2 - v_1} + \frac{v_1 \Delta y + (v_2 - v_1)(y_p - y_1)}{(v_2 - v_1)\Delta y} e^{\frac{6(v_2 - v_1)(x_p - x_1)(x_1 - x_p)\Delta \tau}{\Delta x^2 \Delta y}}
\]

\[(41)\]

\[
z_e = z_1 - \frac{w_1 \Delta z}{w_2 - w_1} + \frac{w_1 \Delta z + (w_2 - w_1)(z_p - z_1)}{(w_2 - w_1)\Delta z} e^{\frac{6(w_2 - w_1)(x_p - x_1)(x_1 - x_p)\Delta \tau}{\Delta x^2 \Delta z}}
\]

\[(42)\]

Figure 6. Two opposing voxels are solid.
2.2. Two adjacent voxels are solid

We show the equations for the case where one of solid voxels blocks the $x_2$-face and the other blocks the $z_1$-face (Fig. 7); for the other eleven cases, the formulation can be obtained similarly. The velocity distribution is:

$$u = \frac{2u_2}{\Delta x^2 \Delta y} (x - x_i)^2 (y - y_i)$$  (43)

$$v = \frac{2v_2}{\Delta y^2 \Delta z} (x - x_i) (y - y_i)^2$$  (44)

$$w = \frac{4w_1}{\Delta x \Delta y} (x - x_i) (y - y_i) + \frac{4(w_2 - w_1)}{\Delta x \Delta y \Delta z} (x - x_i) (y - y_i) (z - z_i)$$  (45)

The time of flight increment across the block is:

$$\Delta \tau_x = \frac{\left(\frac{w_1 \Delta x \Delta y}{u_2 \Delta y + v_2 \Delta z} + z_1 - z_p\right)^{0.5} \Delta x^2 \Delta y^2}{2(x_p - x_i)(y_p - y_i)[w_1 \Delta x \Delta y(u_2 \Delta y + v_2 \Delta z) + (z_1 - z_p)(u_2 \Delta y + v_2 \Delta z)]^{0.5}} \left[1 - \left(\frac{\Delta x}{x_p - x_i}\right)\frac{w_2 \Delta y + v_2 \Delta z}{w_2 \Delta y}\right]$$  (46)

$$\Delta \tau_y = \frac{\left(\frac{w_1 \Delta x \Delta y}{u_2 \Delta y + v_2 \Delta z} + z_1 - z_p\right)^{0.5} \Delta x^2 \Delta y^2}{2(x_p - x_i)(y_p - y_i)[w_1 \Delta x \Delta y(u_2 \Delta y + v_2 \Delta z) + (z_1 - z_p)(u_2 \Delta y + v_2 \Delta z)]^{0.5}} \left[1 - \left(\frac{\Delta y}{y_p - y_i}\right)\frac{w_2 \Delta z + u_2 \Delta x}{v_2 \Delta z}\right]$$  (47)

$$\Delta \tau_z = \frac{\Delta x^2 \Delta y^2}{2(x_p - x_i)(y_p - y_i)[w_1 \Delta x \Delta y(u_2 \Delta y + v_2 \Delta z) + (z_1 - z_p)(u_2 \Delta y + v_2 \Delta z)]^{0.5}} \times \left[\frac{w_2 \Delta x \Delta y}{u_2 \Delta y + v_2 \Delta z} + z_1 - z_p\right]^{0.5} - \left(\frac{w_1 \Delta x \Delta y}{u_2 \Delta y + v_2 \Delta z} - \Delta z\right)^{0.5}$$  (48)

$$\Delta \tau = \min(\Delta \tau_x, \Delta \tau_y, \Delta \tau_z)$$  (49)

And the exit location coordinates are:
3. Three of the neighboring voxels are solid

This condition can happen in twenty ways depending on the locations of solid voxels and can be categorized in two groups. The three surrounding solid voxels block all three coordinate directions; one solid voxel is on x-, the other one on y- and the last one on the z-direction face. This condition can happen in eight different ways. The second group is when two of the three solid voxels block the same direction (for example both are in the x-direction) and the third one is in another direction.

3.1. Three of the neighboring voxels are solid that block in different directions

We have one solid voxel in each coordinate direction and in each direction we have two potential positions, so there are eight cases. We show the equations for one of them (Fig. 8) and the others can be calculated in the same way.

\[
x_e = x_1 + (x_p - x_0)\left(1 - \frac{2(x_p - x_1)(y_p - y_1)\left[w_1\Delta x\Delta y(u_2\Delta y + v_2\Delta x) + (z_1 - z_p)(u_2\Delta y + v_2\Delta x)^2\right]^{0.5}}{w_1\Delta x\Delta y + z_1 - z_p}\right)\frac{u_2\Delta y}{u_2\Delta y + v_2\Delta x} \tag{50}
\]

\[
y_e = y_1 + (y_p - y_0)\left(1 - \frac{2(x_p - x_1)(y_p - y_1)\left[w_1\Delta x\Delta y(u_2\Delta y + v_2\Delta x) + (z_1 - z_p)(u_2\Delta y + v_2\Delta x)^2\right]^{0.5}}{w_1\Delta x\Delta y + z_1 - z_p}\right)\frac{v_2\Delta x}{u_2\Delta y + v_2\Delta x} \tag{51}
\]

\[
z_e = z_1 + \frac{w_1\Delta x\Delta y}{u_2\Delta y + v_2\Delta x} - \left[\frac{w_1\Delta x\Delta y}{u_2\Delta y + v_2\Delta x} + z_1 - z_p\right]^{0.5} - \frac{2(x_p - x_1)(y_p - y_1)\left[w_1\Delta x\Delta y(u_2\Delta y + v_2\Delta x) + (z_1 - z_p)(u_2\Delta y + v_2\Delta x)^2\right]^{0.5}}{\Delta x^2\Delta y^2} \tag{52}
\]

The time of flight increment across the block can be calculated:

\[
\Delta \tau_x = \frac{\Delta x^2\Delta y\Delta z}{4(x_2 - x_1)(y_2 - y_1)(z_2 - z_1)u_1} \ln\left(-\frac{x_p - x_1}{\Delta x}\right) \tag{56}
\]

\[
\Delta \tau_y = \frac{\Delta x\Delta y^2\Delta z}{4(x_2 - x_1)(y_2 - y_1)(z_2 - z_1)v_1} \ln\left(-\frac{y_p - y_1}{\Delta y}\right) \tag{57}
\]

\[
\Delta \tau_z = \frac{\Delta x\Delta y^2\Delta z}{4(x_2 - x_1)(y_2 - y_1)(z_2 - z_1)w_1} \ln\left(-\frac{z_p - z_1}{\Delta z}\right) \tag{58}
\]

\[
\Delta \tau = \min(\Delta \tau_x, \Delta \tau_y, \Delta \tau_z) \tag{59}
\]

And the exit location is:
Three of the neighboring voxels are solid and two of them are in the same direction

This condition can happen for twelve cases; we show the equations for the case where two solid voxels have blocked the x-direction (Fig. 9). Therefore, the velocity component in the x-direction will be zero. The velocity distribution is:

\[ u = 0 \] \hspace{1cm} (63)

\[ v = \frac{12v_1}{\Delta x^2 \Delta z} (x_2 - x)(x - x_1)(z - z_1) + \frac{12(v_2 - v_1)}{\Delta x^2 \Delta y \Delta z} (x_2 - x)(x - x_1)(z - z_1)(y - y_1) \] \hspace{1cm} (64)
\[ w = \frac{6w_2}{\Delta x^2 \Delta z^2} (x_2 - x)(x - x_i)(z - z_1)^2 \] (65)

Time of flight increment across the block is:

\[ \Delta \tau_y = \frac{\Delta x^2 \Delta z^2}{6w_2(x_2 - x_p)(x_p - x_i)(z_p - z_1)} \left(1 - \left(\frac{v_1 \Delta y + (v_2 - v_1)\Delta y}{v_1 \Delta y + (v_2 - v_1)(y_p - y_i)}\right) \right) \] (66)

\[ \Delta \tau_z = \frac{\Delta x^2 \Delta z^2}{6(x_2 - x_p)(x_p - x_i)w_2} \left(\frac{1}{z_p - z_1} - \frac{1}{\Delta z}\right) \] (67)

\[ \Delta \tau = \min(\Delta \tau_y, \Delta \tau_z) \] (68)

And the exit coordinations are:

\[ x_e = x_p \] (69)

\[ y_e = y_i - \frac{v_1 \Delta y}{v_2 - v_1} + \left(1 - \frac{6w_2(x_2 - x_p)(x_p - x_i)(z_p - z_1)}{\Delta x^2 \Delta z^2} \right) \frac{-2(y_p - y_i)}{w_2 \Delta \tau} \] (70)

\[ z_e = z_1 + \frac{\Delta x^2 \Delta z^2 (z_p - z_1)}{\Delta x^2 \Delta z^2 - 6w_2(x_2 - x_p)(x_p - x_i)(z_p - z_1) \Delta \tau} \] (71)

4. **Four of the neighboring voxels are solid**

This condition can happen in fifteen cases that can be categorized in two groups: (1) when in two directions we have just one solid voxel and in the third direction there are two solid voxels; and (2) when there is no solid voxel in one of the three coordinate directions.

4.1. **Four of the neighboring voxels are solid and three of them are in three different directions**

We consider the case where there are two solid voxels in the x-direction and the other two are in the y and z directions (**Fig. 10**). Since the x-direction is blocked from both sides, the velocity component in the x direction is zero. The equations for the other cases can be obtained similarly.

\[ u = 0 \] (72)

\[ v = \frac{12v_1}{\Delta x^2 \Delta y^2 \Delta z}(x_2 - x)(x - x_i)(y_2 - y)^2(z - z_1) \] (73)

\[ w = \frac{12w_2}{\Delta x^2 \Delta y \Delta z^2}(x_2 - x)(x - x_i)(y_2 - y)(z - z_1)^2 \] (74)

The time of flight increment is:

\[ \Delta \tau_y = \frac{\Delta x^2 \Delta y^2 \Delta z}{12v_1(y_2 - y_p)(z_p - z_1)(x_2 - x_p)(x_p - x_i)} \ln \left( \frac{y_2 - y_p}{\Delta y} \right) \] (75)
\[ \Delta \tau_{\varepsilon} = \frac{\Delta z^2 \Delta y^2 \Delta y}{12 w_1 (y_2 - y_p)(z_p - z_1)(x_2 - x_p)(x_p - x_1) \ln \left( \frac{\Delta z}{z_p - z_1} \right)} \] (76)

\[ \Delta \tau = \min(\Delta \tau_y, \Delta \tau_z) \] (77)

And the exit location is:

\[ x_e = x_p \] (78)

\[ y_e = y_2 - (y_2 - y_p)e \frac{-12w_1(x_2 - x_1)(y_2 - y_p)(z_2 - z_1)}{\Delta z^2 \Delta y^2 \Delta y} \Delta \tau \] (79)

\[ z_e = z_1 + (z_p - z_1)e \frac{12w_1(x_2 - x_1)(x_2 - x_p)(y_2 - y_p)(z_2 - z_1)}{\Delta x^2 \Delta z^2 \Delta y} \Delta \tau \] (80)

Figure 10. Four nearest-neighbor solid voxels and there is at least one solid voxel at each coordinate direction.

Figure 11. Four nearest-neighbor solid voxels where there is no solid voxel blocking one of the coordinate directions.
4.2. **Four of the neighboring voxels are solid and they block two directions**

We present the equations for the case where there is no solid voxel in the z direction and the velocity components in x and y directions are zero (Fig. 11). The formulation for the other cases can be obtained similarly.

\[ u = 0 \] (81)

\[ v = 0 \] (82)

\[ w = \frac{36 w_i}{\Delta x^2 \Delta y^2} (x_2 - x)(x - x_i)(y_2 - y)(y - y_i) \] (83)

The time of flight increment across the block is:

\[ \Delta \tau = \frac{\Delta t^2 \Delta y^2}{36 w_i (x_2 - x_p)(x_p - x_i)(y_2 - y_p)(y_p - y_i)} \] (84)

The exit location is:

\[ x_e = x_p \] (85)

\[ y_e = y_p \] (86)

\[ z_e = z_p + \frac{36 w_i}{\Delta x^2 \Delta y^2} (x_2 - x)(x - x_i)(y_2 - y)(y - y_i) \Delta \tau \] (87)

If five or six faces are solid, the velocity everywhere is zero.

**Acknowledgments**

We thank the sponsors of the Imperial College Consortium on Pore-Scale Modeling (BG, BP, JOGMEC, Saudi Aramco, Shell, Statoil and Total) for their financial support.

**References**


Letters, 57, 996-999.


