A Streamline Simulator to Model Field Scale Three-Dimensional Flow

Rod P. Batycky, Stanford University
Marco R. Thiele, Stanford University
Martin J. Blunt, Stanford University


ABSTRACT

We present a streamline method to model convective-dominated displacements in heterogeneous porous media. The method is applicable to 2D and 3D systems with any number of horizontal/vertical injectors and producers. In our method we trace streamlines from injectors to producers through a heterogeneous permeability field, and then treat each streamline as a 1D system along which any one-dimensional solution can be mapped. In this paper we map tracer, waterflood, and first-contact miscible one-dimensional displacements along the streamlines. By treating the streamlines as 1D systems we effectively decouple the physics of the displacement from the influences of heterogeneity, thereby allowing us to construct large 3D solutions efficiently. Results indicate that accurate performance predictions of multi-well simulations are possible with speed-up factors ranging between 10 and 200 times compared to conventional fully implicit finite-difference approaches. Furthermore, our results are free of numerical diffusion because of the analytical description of the streamline paths, and the ability to map analytical one-dimensional solutions along the streamlines. We present example solutions showing that numerical diffusion in conventional methods can give recovery estimates that are up to 10% higher than the numerical-diffusion free streamline results.

INTRODUCTION

Streamlines and streamtubes have traditionally been applied to model 2D areal flows in porous media. Early work using streamlines was by Fay & Pratts [9], who studied tracer and waterflood responses in a quarter five-spot pattern. Higgins & Leighton [10] introduced the use of streamtube bundles to multi-well fields and mapped 1D Buckley-Leverett solutions along each tube. Their method assumed that streamtube positions remained fixed in time. This is a valid assumption for moderately unfavorable mobility ratio waterfloods. To account for the changing mobility field using fixed streamtubes they altered tube injection rates in accordance with the total resistance along each streamtube. Martin & Wegner [15] introduced a model that periodically updated the streamtube positions and found they could then better predict highly favorable and mildly unfavorable mobility ratio floods. Renard [20] proposed a similar idea to also account for changing well rates. Emanuel et al. [7] extended the 2D streamtube model of Higgins & Leighton to a hybrid 3D model. Instead of mapping an analytical model to the areal streamtubes, they used a general finite difference 1D solution generated from a representative 2D cross-section, as proposed by Lake et al. [14]. The hybrid model gave excellent history matches when applied to field cases.

The work we present here extends that of Thiele et al. [21, 22, 23] who developed a new streamtube method in 2D. They periodically recalculated the streamtube positions, and also assumed that the same 1D solution could be remapped along each streamtube at each time step. The 1D solution was assumed valid regardless of how much the streamtube positions shifted. Thus they ignored whatever saturations were present along the recalculated streamtubes when finding the solution at the new time. They concluded that this was a valid approach for displacements dominated by heterogeneity. The benefit of the method was a considerable savings in CPU time, 10 - 10^5 orders of magnitude over conventional IMPES finite difference methods.
The simplicity of Thiele et al.’s method, the considerable time savings and the broad range of displacement mechanisms warrant the extension of their ideas to 3D multi-well field cases. In this paper we use streamlines instead of streamtubes, although the two methods will give identical results in 2D. Datta-Gupta & King [5] used a streamline method to model tracer and two-phase flow in 2D, and King et al. [12] modeled miscible displacements in 2D. The advantage of using streamlines over streamtubes is that they are easily extended to 3D systems directly, whereas streamtubes become complicated geometrical objects in 3D. The disadvantage of using streamlines is that they do not carry an explicit volumetric flux.

The streamline method presented here centers on decoupling the physics of flow from the geological heterogeneity. The influences of heterogeneity are captured in the 3D distribution of streamlines, while the physics of flow is captured in the appropriate 1D solution mapped along each streamline. The only difference between a tracer, waterflood, first contact miscible (FCM), or compositional displacement is simply the type of 1D solution used. This decoupling offers several advantages over conventional simulation methods: (1) large speed-up factors depending on the complexity of the displacement modeled, (2) analytical streamline calculations coupled with analytical 1D solutions yield results free of numerical diffusion (3) for a fixed level of memory we can solve models 10 times larger. These advantages make the streamline method ideally suited for systems with large numbers of gridblocks, processing multiple geostatistical realizations, and history matching.

TRACING STREAMLINES

Tracing streamlines from injectors to producers is based on the analytical description of a streamline path within a gridblock as outlined by Pollock [19]. The underlying assumption is that the velocity field in each coordinate direction varies linearly and is independent of the velocities in the other directions. Pollock’s method is attractive because it is analytical and consistent with the governing material balance equation. For completeness, we include the analytical equations here.

Consider the two-dimensional gridblock in Fig. 1, for which we know the interstitial velocity field and have defined a local coordinate system and origin. The velocity in the x-direction, $V_x$, is defined as

$$ V_x = V_{x,o} + m_x(x - x_o), $$

where $m_x$ is the velocity gradient across the gridblock and is given by

$$ m_x = \frac{V_{x,\Delta x} - V_{x,o}}{\Delta x}. $$

Knowing that $V_x = dx/dt$, one can integrate Eq. 1 to yield the time required to reach an x exit face $\Delta t_{e,x}$ as,

$$ \Delta t_{e,x} = \frac{1}{m_x} \ln \left\{ \frac{V_{x,o} + m_x(x_e - x_o)}{V_{x,o} + m_x(x_i - x_o)} \right\}, \quad (3) $$

where $x_i$ is the inlet position and $x_e$ is the exit position, in the x-coordinate direction. Similarly, the times required to reach the exit faces in the y or z directions are given by

$$ \Delta t_{e,y} = \frac{1}{m_y} \ln \left\{ \frac{V_{y,o} + m_y(y_e - y_o)}{V_{y,o} + m_y(y_i - y_o)} \right\}, \quad (4) $$

and

$$ \Delta t_{e,z} = \frac{1}{m_z} \ln \left\{ \frac{V_{z,o} + m_z(z_e - z_o)}{V_{z,o} + m_z(z_i - z_o)} \right\}. \quad (5) $$

The correct face which the streamline will exit is that which requires the smallest value of $\Delta t_e$ calculated from Eqs. 3, 4, and 5. Knowing the minimum time, $\Delta t_e$, the exact exit location of the streamline is determined by rewriting Eqs. 3, 4, and 5 as,

$$ x_e = \frac{1}{m_x}(V_{x,i} \exp\{m_x \Delta t_e\} - V_{x,o}). \quad (6) $$

$$ y_e = \frac{1}{m_y}(V_{y,i} \exp\{m_y \Delta t_e\} - V_{y,o}). \quad (7) $$

$$ z_e = \frac{1}{m_z}(V_{z,i} \exp\{m_z \Delta t_e, V_{z,o}) - V_{z,o}). \quad (8) $$

For the trivial case when velocity is uniform across a gridblock in a given direction ($m = 0$), Eq. 3, for example, simply becomes $\Delta t_{e,x} = (x_e - x_i)/(V_{x,o})$ and Eq. 6 becomes $x_e = x_o + \Delta t_{e,x} V_{x,o}$. Finally, for the situation where a flow divide exists in the x direction within a gridblock, one must assure that the sign
at a potential possibility of calculating negative logarithms in Eqs. 3, 4, and 5.

Calculating Time-of-Flight

King et al. [12] derived similar equations to trace streamlines and also introduced the time-of-flight (TOF) concept as a means to determine gridblock saturations. The time-of-flight, \( \tau \), is the time required to travel a distance, \( s \), along a streamline and is defined as,

\[
\tau = \int_{0}^{s} \frac{d\zeta}{V(\zeta)}. \tag{9}
\]

Note that evaluating \( \tau \) only requires knowing interstitial velocities. We calculate \( \tau \) along a streamline by using Eqs. 3 - 8, while keeping track of the elapsed time from the starting position. The starting positions for all streamlines are the outlet faces of gridblocks containing an injector. Because a streamline does not carry a specific flux, as a streamtube does, we instead assign the same flux to each streamline and launch streamlines from each injection gridblock face in proportion to the flux out of the face. This approximation only affects the calculated fluid cuts at individual producers. Total field fluid cut and gridblock properties are correct. We have found that the error between the flux calculated by adding streamline fluxes of arriving streamlines and the flux calculated from the pressure solution, for a given producer, is less than 0.1%.

For heterogeneous flow fields some gridblocks will not contain a streamline. To assign a TOF we simply trace backwards from the center of the missed gridblock until we reach a gridblock that does contain an average TOF already calculated from at least two streamlines. The missed block's TOF is then the known average TOF plus the time required to trace back.

Mapping a 1D Solution

To map 1D solutions along streamtubes that scaled by \( x_D/t_D \) Higgins & Leighton used the following definition of dimensionless porevolume, \( x_D \), within a streamtube

\[
x_D = \int_{0}^{s} \frac{\phi(\zeta) A(\zeta) d\zeta}{V_p}, \tag{10}
\]

and dimensionless time, \( t_D \)

\[
t_D = \int_{0}^{t} \frac{q dt}{V_p}, \tag{11}
\]

where \( \zeta \) is a coordinate along the streamtube, \( \phi \) is porosity, and \( A \) is the streamtube cross-sectional area. \( V_p \) is an arbitrary volume use for scaling, it is constant for all streamtubes, and is typically defined as the average streamtube volume in a model.

Streamlines do not carry a volumetric flux, \( q \), or have a volume associated with them. However, it is possible to establish the relationship between \( \tau \) and the volumetric information along a streamtube. Starting with the above definitions for a streamtube and assuming a constant flow rate along the tube of \( q \), one gets

\[
\frac{x_D}{t_D} = \int_{0}^{s} \frac{A(\zeta) \phi(\zeta) d\zeta}{q t} = \frac{1}{t} \int_{0}^{s} \frac{A(\zeta) \phi(\zeta) d\zeta}{V(\zeta)} = \frac{1}{t} \int_{0}^{s} \frac{d\zeta}{V(\zeta)} = \frac{\tau}{t}. \tag{12}
\]

The above equation reveals that any solution that scales as \( x_D/t_D \) along a streamtube is equivalent to a solution that scales as \( \tau/t \) along a streamline.

For a given gridblock we must ultimately determine a single property, whether saturation or composition, at a given time \( t \). For each streamline passing through a gridblock we first calculate (\( \bar{\tau} \)) as the average between the inlet and exit TOF's for the streamline. We then use Eq. 12 with \( \bar{\tau} \) to find \( x_D/t_D \) and the corresponding property for the streamline. A single property is assigned to the gridblock by then averaging all the properties for each streamline that pass through the block. Mathematically the average saturation for a gridblock with \( n_{sl} \) streamlines passing through it is calculated as,

\[
\bar{S}_{gb} = \frac{1}{n_{sl}} \sum_{i=1}^{n_{sl}} \bar{S}_i, \tag{13}
\]

where the average saturation \( \bar{S}_i \) along streamline \( i \) at time \( t \), is rigorously defined as

\[
\bar{S}_i = \frac{1}{\Delta \tau} \int_{\tau - \Delta \tau}^{\tau} S_i(\tau) d\tau |_{\tau}, \tag{14}
\]

but which we approximate as,

\[
\bar{S}_i \approx S_i(\bar{\tau}) |_{\tau}. \tag{15}
\]

Time Stepping

The difference between our method and a conventional finite difference technique is in how we move saturations/compositions forward in time. In our method we move from time \( t \) to time \( t + \Delta t \) as follows,

- At the start of a new time step, \( t + \Delta t \), solve for the pressure field \( P \) using,

\[
\nabla \cdot \overline{\lambda_t} \nabla P = 0, \tag{16}
\]

where \( \overline{\lambda_t} \) is the total mobility and is defined as

\[
\overline{\lambda_t} = \overline{K} \sum_{i=1}^{n_p} k_{r,i}/\mu_i. \tag{17}
\]

Eq. 16 is solved using a conventional seven point finite difference method, with no flow boundary conditions over the surface of the domain and specified pressure or rate at the wells.
A Streamline Simulator to Model Field Scale Three-Dimensional Flow

- Apply Darcy’s law and determine the interstitial velocity field.
- Reset all saturations/compositions and mobility information back to uniform initial conditions, for which the 1D solution is valid. Retain the current velocity field.
- Trace streamlines using Eqs. 3 - 8. While tracing use Eq. 12 with \( t \) now equal to \( t + \Delta t \) and map the 1D solution along each streamline. In this manner the pressure field lags behind the saturation/composition field by \( \Delta t \) (an IMPES method).
- After tracing all the streamlines, use Eq. 13 to determine gridblock average properties.
- Calculate the true time, \( T \), for the end of the current time step, as outlined in the Appendix.

This time stepping method is approximate since we ignore the current saturation information along streamlines when updating the solution. Recall that we assume initial uniform conditions prevail along the streamlines at the beginning of each mapping step. Because of the approximations in time stepping the method is best suited for flow dominated by heterogeneity. In these cases streamline paths and TOF’s along the streamlines change little through time implying that the assumption of uniform initial conditions is more closely obeyed.

The method we present here is also always stable no matter the size of the time step. The issue, instead of a CFL criterion for stability, is how many time steps are required to consider the solution converged. We considered a solution converged when there was no additional change in recovery with increased number of time steps over a fixed total simulation time. In the examples we show, typically 20 to 30 time steps were required for waterflooding and 50 to 60 time steps were required for FCM displacements, over a total simulation time of 2 pore volumes injected (PVI).

Quarter 5-Spot Tracer Case

To validate our method outlined in the previous section, we compare the tracer production at a producer in a quarter five-spot against the analytical solution [17] and results from ECLIPSE [11], on a 100X100 grid, as shown in Fig. 2. With our method, analytical tracing of streamline paths combined with mapping a 1D tracer solution should yield exact results, since for the special case of tracer flow the streamlines are fixed for all time. Our streamline results (3DSL) do match the analytical solution exactly, and there is no numerical diffusion or grid orientation effects, as in a conventional simulation. Similar results using streamlines were presented by Fay & Pratts [9] and more recently by Data-Gupta & King [5].

A Word About Time Comparisons

We have presented an IMPES method here and ideally should compare CPU times against other IMPES methods. However, given the large models we study, conventional IMPES finite difference solutions are prohibitively expensive. It is more likely that one would use a fully implicit method instead. As a result, speed up factors quoted are comparisons of CPU time taken for the streamline method versus an appropriate fully implicit finite difference solution. Note that these are considered pessimistic factors since our code is far from optimized.

IMMISCIBLE TWO-PHASE FLOW

To model a waterflood we use the classic Buckley-Leverett profile [2] as our 1D solution. The relative permeabilities are simple Corey-type curves [4] of the form \( k_{rw} = (S_w)^2 \) and \( k_{ro} = (1 - S_w)^2 \) and a fluid viscosity ratio of \( \mu_o/\mu_w = 10 \). The resulting Buckley-Leverett profile mapped along each streamline is shown in Fig. 3. This 1D profile is used to simulate a waterflood in a full five-spot 50,000 gridblock (50X50X20) pattern initially at 100% oil saturation. The producers are completed in the upper ten layers and the injector at the center is completed in the lower ten layers. We studied three cases each with different characteristics in the permeability distributions (Table 1). The dis-

Table 1: Correlation length parameters used to build geologic models for results in Fig. 4.

<table>
<thead>
<tr>
<th>Permeability Correlation Length ( \lambda )</th>
<th>Horiz. Rotation wrt. X-axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field ( \lambda_{x,x} )</td>
<td>( \lambda_{y,y} )</td>
</tr>
<tr>
<td>Case 1</td>
<td>0.3</td>
</tr>
<tr>
<td>Case 2</td>
<td>0.4</td>
</tr>
<tr>
<td>Case 3</td>
<td>0.4</td>
</tr>
</tbody>
</table>

*For rotated coordinate systems we use the longest model distance along the rotated coordinate axis, when calculating \( \lambda_{\cdot} \).
tributions were constructed using sequential Gaussian simulation [6].

The oil recovery curves are shown in Fig. 4 and compared with ECLIPSE fully implicit solutions. In each case 5,000 streamlines were used. A comparison of individual well watercuts for case 1 is also included and is shown in Fig. 5. The agreement for the three different permeability fields is excellent. Agreement is also excellent on a per well basis, although numerical diffusion in ECLIPSE does delay breakthrough at the two on-trend producers. For these examples, streamline results were computed 50 times faster than the conventional results.

![Image](image-url)  
**Figure 3:** Buckley-Leverett profiles for water saturation, water fractional flow, and total mobility as a function of \( \tau/t \).

**FIRST CONTACT MISCIBLE FLOW**

First contact miscible displacements are considerably more model than 2-phase displacements. The large mobility contrast between the solvent and the oil makes the problem highly nonlinear, while instabilities in the displacement front leads to substantial numerical diffusion in conventional methods. The method presented here cannot reproduce viscous fingering explicitly. Instead we choose to map a 1D solution along each streamline that models viscous fingering in an average sense. See Thiele et al. [22] for a thorough discussion on the choice of 1D model to use when simulating miscible flooding using streamtubes. For the FCM cases shown here, Koval's 1D model [13] is used with a fluid viscosity ratio of \( \mu_s/\mu_o = 10 \). The resulting 1D solution is shown in Fig. 6.

![Image](image-url)  
**Figure 4:** Field oil recovery in a full five-spot pattern for three different permeability fields.

![Image](image-url)  
**Figure 5:** Comparison of individual well watercuts for Case 1.

ECLIPSE scheme leads to an optimistic recovery, compared with the other methods. Fig. 7 highlights that with the proper finite difference method recovery predictions can agree. We compared our results in the remainder of this section against the fully implicit ECLIPSE two-point upstream method, since this method gives much better results than the standard scheme.

The three 3D permeability fields used in the waterflood example are again used here. The well configurations are unchanged. Fig. 8 shows the oil recovery predicted by the streamline method and ECLIPSE. Note that in two cases recovery is over predicted by conventional simulation. We attribute this to numerical diffusion in the finite difference solution. The fact that case 3 results agree suggests that different permeability characteristics result in different levels of numerical diffusion. We have found similar results when comparing tracer flood response on various fields. In summary, the streamline results were calculated with speedup factors of two orders of magnitude over the ECLIPSE results. To minimize numerical diffusion and improve recovery agreements for cases 1 and 2,
A Streamline Simulator to Model Field Scale Three-Dimensional Flow

ECMOR V, 1996

Figure 6: FCM Koval model profiles for gas saturation, gas fractional flow, and total mobility as a function of $\tau/t$.

Figure 7: Oil recovery in a FCM cross-section displacement by various simulation methods.

Ideally we should generate IMPES ECLIPSE results. Time constraints make this prohibitive and in fact this type of comparison would yield speed up factors of order 1000 or more [21].

MILLION GRIDBLOCK MODEL

The decoupling of heterogeneity from fluid flow makes the streamline method particularly suited to solve large problems. For a fixed amount of memory problems ten times larger can be studied with the streamline method than can be handled with conventional methods. Here we illustrate the solution of a 1.16 million gridblock model (220x220x24) waterflood using a standard workstation and compare the results with an upscaled 72,600 gridblock model (110x110x6) ECLIPSE solution. We could not generate a million gridblock ECLIPSE solution with our computer resources. A streamline solution for the upscaled model is also included for reference. Note that simple geometric upscaling of absolute permeability was used, with no upscaling of relative permeabilities. The fine-scale permeability field was generated with GSLIB. Correlation lengths were: on trend $\lambda_c = 0.6$, off-trend $\lambda_c = 0.05$, and the vertical correlation length was $\lambda_c = 0.6$. One producer was completed in each corner in the upper twelve layers, while the central injector was completed in the lower twelve layers.

Fig. 9 provides a visual comparison of the water saturation in the five-spot at $t_D=0.2$ for each model. Because of the absence of numerical diffusion, both streamline cases show water breakthrough at all four producers, not just the on-trend producers as with ECLIPSE. Also both streamline solutions show resolution to a single gridblock scale.

Fig. 10 compares the oil recovery of the three cases. Note that there is a greater change in recovery due to scale than due to the choice of simulation method. For this problem, having the ability to capture all of the first order influences (due to permeability) outweighs the approximations in the method. The upscaled finite difference solution over estimates recovery by 5%.

SCREENING EQUIPROBABLE IMAGES

The streamline method is ideally suited to screen multiple geostatistical realizations before resorting to a more conventional solution method [21]. As an example, consider a 100,000 gridblock waterflood model with 8 producers, 9 injectors, and one horizontal producer, completed in 5 spot patterns (Fig. 11). The injectors and vertical producers are full interval completions, while the horizontal producer is completed in layer five only. Thirty equiprobable realizations were created using sequential Gaussian simulation. First the streamline method was used to generate the corresponding oil recovery curves, then the permeability fields resulting in the high and low recoveries were rerun using ECLIPSE. The oil recovery results are summarized in Fig. 12. Clearly the range in recovery predicted by the streamline method is also predicted by the two ECLIPSE runs. Note however, that
Figure 9: Water saturation for a 5-spot waterflood at $t_D=0.2$ for (a) 1.16 million gridblock streamline model, (b) upscaled streamline model, (c) upscale ECLIPSE model. Note that all gridblocks with a water saturation of less than 20% have been removed.

The ECLIPSE results are shifted consistently up, which we attribute to numerical diffusion effects. For these multi-well field cases, 3DSL results for a single image were generated in 1.5 hours, while ECLIPSE results were generated in 140 hours for the low case and 300 hours for the high case. The speed up for this large waterflood problem is now 100-200 times, versus 50 times for the 3 smaller problems studied earlier.

For existing fields with large amounts of production data to honor, rather than using the streamline method to screen images it can be used in the history matching process [16, 7, 8] since multiple runs required to obtain a history match can be performed very quickly. Once a reservoir model that satisfies historical data is built, either the streamline method or a conventional simulator can be used for prediction purposes.

DISCUSSION

The streamline method we present here, does have some limitations. We have assumed incompressible flow, although streamlines can be defined for compressible flow [18]. Crossflow phenomena between streamlines or gravity effects cannot be modeled. Uniform initial conditions are required, thus changing the boundary conditions is not possible. For general application to field scale displacements it is important to model gravity and handle nonuniform initial conditions. We are successfully overcoming both problems by using a general 1D finite difference solver as the source for 1D solutions mapped along streamlines. This idea is similar to that used by Bommer & Schecter [1].

We have also shown the limitations inherent in conventional finite difference methods due to speed. Simulating million gridblock multiwell multiphase models is still prohibitive with conventional methods. By decoupling the 3D problem into a series of 1D problems we have demonstrated that accurate solutions to very large models can be generated hundreds of times quicker with the streamline method. Time savings will be even greater for 3-phase and compositional displacements.

CONCLUSIONS

We have successfully developed a 3D streamline simulator capable of modeling field scale flow. Our
method works for a variety of displacement mechanisms, can be applied to large models and can account for multiple wells, either vertical or horizontal. For heterogeneity dominated flow, the streamline method is a fast alternative to conventional simulations. Time savings over conventional implicit finite difference methods are as high as two orders of magnitude. Furthermore the solutions are free of numerical diffusion and illustrate that conventional methods, because of numerical artifacts, tend to over predict recovery in field situations.

Finally we showed that decoupling the problem into influences due to heterogeneity and influences due to displacement physics results in several benefits. The streamline method can solve larger problems thus reducing the need for substantial upscaling. Decoupling also reduces simulation time giving the ability to capture uncertainty quickly, act as a screening tool on equiprobable realizations, and accelerate the history matching process.

ACKNOWLEDGEMENTS

We gratefully acknowledge financial support from the DOE (Grant No. DE-FG22-92BC14852), members of the Stanford U. Petroleum Research Inst. for Gas Displacement (SUPRI-C). We also thank BP for financial support and use of their FCT code.

NOMENCLATURE

\[ A \] = cross-sectional area of streamtube
\[ f^i \] = field average frac. flow of injected phase at end of \( n^{th} \) time step
\[ f_{i} \] = field average frac. flow of injected phase over \( n^{th} \) time interval
\[ K \] = absolute permeability tensor
\[ k_{i} \] = relative permeability of phase \( i \)
\[ m_{a} \] = \( z \) velocity gradient across a gridblock in the \( z \) direction
\[ n_{p} \] = number of phases
\[ n_{sl} \] = number of streamlines in a gridblock

| \( P \) | pressure |
| \( Q \) | total field injection or production fluid rate |
| \( q \) | volumetric flux within a streamtube |
| \( S_{i} \) | average water saturation along \( i^{th} \) streamline in a gridblock |
| \( S_{g,b} \) | average water saturation in a gridblock |
| \( s \) | spatial distance along a streamline |
| \( T \) | corrected time |
| \( t \) | time |
| \( t_{D} \) | dimensionless time |
| \( \Delta t_{e,x} \) | time of flight required to reach an \( x \) exit face |
| \( V \) | interstitial velocity |
| \( V_{p} \) | average streamtube porevolume |
| \( W_{init} \) | volume of water present at start of simulation |
| \( W_{r}^{n+1} \) | volume of water remaining in reservoir at the end of the \( n+1 \) time step |
| \( W_{r}^{n} \) | volume of water injected up to end of the \( n^{th} \) time step |
| \( W_{p}^{n} \) | volume of water produced up to end of the \( n^{th} \) time step |
| \( x_{D} \) | dimensionless distance |
| \( x_{e} \) | \( z \) coord. value of streamline exit location |
| \( z_{i} \) | \( z \) coord. value of streamline inlet location |
| \( z_{e}, y_{e}, z \) | spatial coordinates |
| \( \lambda_{c} \) | dimensionless correlation length |
| \( \lambda_{t} \) | total mobility |
| \( \tau \) | time of flight |
| \( \phi \) | average TOP of a streamline in a gridblock |
| \( \mu_{i} \) | viscosity of phase \( i \) |

References

APPENDIX

Each time that the streamline positions are recalculated we introduce the problem of not knowing what the true time, $T^{n+1}$, is for a displacement at the end of time step $n+1$. The problem occurs because grid information is not moved forward in time explicitly, but instead is removed and a 1D solution scaled to $t + \Delta t$ is mapped along the new streamline positions. To determine the true time, historical information is accounted for by a volume balance on the injected phase (for example water) at the end of mapping all streamlines. This leads to an expression for the true time step size over the $n+1$ interval as,

$$
\Delta T^{n+1} = \frac{(W_{\text{init}} - W_{\text{f+1}} + W_{f} - W_{p})}{Q_{n+1}(f_{n+1} - 1)},
$$

where the average field production fractional flow is defined as $f_{n+1} = (f_{n} + f_{n+1})/2$, the average of the field production fractional flows at $n$ and $n+1$, and $Q_{n+1}$ is the total injection rate during the $n+1$ time step. $W_{\text{init}}$ is the initial volume of water in the reservoir, and $W_{p}$ is the volume of water present after mapping all the streamlines at the $n+1$ time step. Cumulative water production up to the previous time step is calculated by,

$$
W_{p} = \sum_{i=1}^{n} Q_{i} \Delta T^{i},
$$

and since all injectors inject 100% water, cumulative water injection volume is simply given by,

$$
W_{r} = \sum_{i=1}^{n} Q_{i} \Delta T^{i}
$$

Thus $t$ can be considered an internal time which is required for the mapping of the 1D solution, and $T$ is the true time that is external and of interest. We keep track of the true time by applying EqdelT after each remapping and then updating $T$ by $T^{n+1} = T^{n} + \Delta T^{n+1}$. Again, $T$ is introduced because of the mapping method we use here. For the tracer case, where streamlines do not change with time, $T = t$ throughout a simulation.