Supplement: Details of the Fracture-Matrix Interchange Model and Transfer Function Method

Generation of transfer functions

This section describes the process for generating the transfer function curves. This is accomplished through a numerical solution of the model domain depicted in Figure 1 of the main paper. To do this, a two-dimensional numerical DFM was set up to perform transport simulations. The model consists of a regular grid domain of regular spacing of 6 m in the z direction (51 grid points in this direction for a total length of 300 m), and increasing grid spacing into the matrix in the x direction, starting with the first column of nodes of width equal to that of a fracture (22 grid points in this direction). These overall dimensions were selected for convenience, with the understanding that the transfer function curves depend on the three dimensionless parameters, rather than the values of individual terms such as length or fracture spacing. In the model simulations, fracture properties are given to the nodes of the first column, and the remaining nodes are given matrix properties. To ensure that parallel flow occurs in the fracture and matrix in the z direction, a flow permeability barrier is established between the fracture and matrix. Furthermore, for injection into the matrix, water is input and output from the boundary nodes in proportion to the volume of that cell. This model design ensures that flow streamlines remain completely in the z direction.

In addition, although the transfer functions being used are for unsaturated transport, there is no requirement that this submodel use unsaturated flow to generate them, as long as the water content values are known. Therefore, for simplicity, these simulations were performed for saturated flow conditions, with the fracture and matrix porosities used instead of water contents.
Note that other possibilities exist, including the use of a simpler approach for comparison purposes that consisted of a single node in the matrix connected to its fracture node; this alternative is described in SNL (2008), and is referred to as the dual-k model. For the dual-k approach, a simple grid was constructed with identical spacings in the z direction, but only one matrix cell in the x direction. Aside from the different grid, cell numbering, and application of boundary conditions, the process for generating the breakthrough curves and transfer functions is the same for the dual-k and DFM models. Furthermore, the use of these curves in an FEHM particle tracking simulation is completely transparent, requiring only a choice of which transfer function file to use.

In the simulations to generate the transfer functions, parameter $p_3$ is varied systematically from fracture-dominated to matrix-dominated flow by varying the relative water flux values in the fractures and matrix. Ranges of other parameter values consistent with the span of those parameters required for the UZ transport model are also selected. Table S-1 lists the variations of each parameter that were used in the formulation of the transfer function curves. Note that for the sorption coefficient $K_d$, the fact that the range of values only goes to 100 does not imply that the model is incapable of accurately simulating transport behavior for higher values of $K_d$. In a section below, a procedure for normalizing the transfer function curves is described whereby higher values of $K_d$ are properly handled. This procedure allows the code to cover arbitrarily large values of $K_d$ without the need to include transfer function curves that extend to such large values.
A four-dimensional matrix of parameters was established with the parameter values listed in the table, and the transfer function curves for each were computed, for a total of 13 × 12 × 3 × 10 = 4680 values of the parameter vector \((p_1, p_2, p_3)\).

Table S-1. List of Parameter Values Used to Compute Transfer Function Curves

<table>
<thead>
<tr>
<th>Parameter</th>
<th># of Values</th>
<th>List of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>(F_f = f_f / (f_f + f_m))</td>
<td>13</td>
<td>0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.99, 0.999, 0.9999</td>
</tr>
<tr>
<td>(D_m)</td>
<td>12</td>
<td>1.e-8, 3.e-9, 1.e-9, 3.e-10, 1.e-10, 3.e-11, 1.e-11, 3.e-12, 1.e-12, 3.e-13, 1.e-13, 1.e-20 m²/s</td>
</tr>
<tr>
<td>(\theta_f)</td>
<td>3</td>
<td>0.01, 0.1, 0.5</td>
</tr>
<tr>
<td>(K_d)</td>
<td>10</td>
<td>0., 0.3, 0.5, 1.0, 3.0, 5.0, 10.0, 30.0, 50.0, 100.0 mL/g</td>
</tr>
</tbody>
</table>

Total: 11 × 12 × 3 × 10 = 4,680 in Excel spreadsheet parameter runs.xls

\* K\(_d\) range of 0-100 does not mean that the model is limited to \(K_d\) values of 100 or less. See below for details on the normalization procedure for handling higher values of \(K_d\).

Two runs of the model are performed for each parameter set: one where solute mass is injected in the fracture, and another where mass is injected in the matrix. The functions themselves, after post-processing and normalization, are placed in a properly formatted file that becomes input to the FEHM computer code (see SNL, 2008 for details).

**Discussion of fracture-matrix submodel behavior**

This section explores the behavior of the fracture-matrix submodel for the two alternate formulations, DFM and dual-k. In contrast to the discrete fracture based transfer function model, the dual-k formulation has a single matrix block for each fracture block. All other aspects of the parameterization are kept the same. Figures S-1 and S-2 compare the DFM and dual permeability transport models for a flow situation consisting of 60 percent fracture flow, 40 percent matrix flow, over a range of diffusion coefficients given in Table S-2. Breakthrough curves from the fracture are presented for solute injection into the fracture at the inlet. Also
shown are vertical, dotted red lines representing the limiting behavior expected for fracture transport and no diffusion (dimensionless time of 1) and composite medium behavior for the case of infinitely large diffusion. Composite medium behavior is attained when the time for diffusion across the model domain $B$ is of that same order or smaller than transport time along the flow path. Under these conditions, the transport time through the system reduces to

$$\tau_{\text{comp}} = \frac{R_f m_f + R_m m_m}{f_f + f_m} \quad (S-1)$$

where $m_f$ and $m_m$ are the fluid masses in a cell for the fracture and matrix, respectively, and $f_f$ and $f_m$ are the fluid mass fluxes for the fracture and matrix, respectively. Intuitively, Eq. S-1 is derived by picturing a solute molecule traveling with fluid of total flux given by the denominator, with total storage volume (including sorption sites) given by the numerator. The time $\tau_{\text{comp}}$ is an important characteristic time for this system, and serves as a reference for understanding the behavior and deriving the detailed method for using transfer function, described in the next section.

### Table S-2. Diffusion Coefficients Used in Simulations

<table>
<thead>
<tr>
<th>Curve Label</th>
<th>Diffusion Coefficient (m$^2$/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1.e-8</td>
</tr>
<tr>
<td>b</td>
<td>1.e-10</td>
</tr>
<tr>
<td>c</td>
<td>1.e-11</td>
</tr>
<tr>
<td>d</td>
<td>1.e-12</td>
</tr>
<tr>
<td>e</td>
<td>1.e-13</td>
</tr>
<tr>
<td>f</td>
<td>1.e-20</td>
</tr>
</tbody>
</table>
Figure S-1. Transfer Function Computed for the Discrete Fracture Model Formulation: Solute Injection in Fracture, Breakthrough in Fracture, Diffusion Coefficients Given in Table S-2

Figure S-2. Transfer Function Computed for the Dual-k Model Formulation: Solute Injection in Fracture, Breakthrough in Fracture, Diffusion Coefficients Given in Table S-2
For each model, the behaviors at the extremes of low and high diffusion are similar. For negligible diffusion, transport times approach a dimensionless time of 1, and the normalized mass flux attains a plateau at 1, which is to say that all mass injected in the fracture leaves via the fracture. By contrast, at high diffusion, transport times approach $\tau_{comp}$, and the plateau of normalized mass flux approaches a value of $F_f$ (0.6 in this example), meaning that at this extreme, the probability of mass in the fracture leaving via the fracture equals the fraction of the total flow traveling through the fracture. It is at the intermediate values of diffusion coefficient that the two models diverge. Specifically, the dual-k formulation tends to predict early breakthrough due to rapid transport through the fracture for a significant portion of the mass, compared to the DFM formulation, for which smooth breakthrough curves at progressively longer transport times are predicted for increasing diffusion coefficients. This means that the dual-k formulation ought to predict earlier breakthrough for the first-arriving mass if the parameter ranges of the model are in this intermediate range. Conversely, the models should be similar behavior for high diffusion or low diffusion.

**Additional Implementation Considerations**

This section addresses some additional details concerning the implementation of the transfer function methodology. These considerations concern the nondimensionalization of the transfer function curves, and the method by which the model handles cases in which some parameters are selected that fall at or outside the range of values assumed when generating the transfer function curves.
Figures S-1 and S-2 demonstrated that, in addition to the fracture transport time $R_f \tau_f$ used to nondimensionalize time in the transfer function curves, the composite transport time $\tau_{comp}$ is a natural parameter for bracketing the behavior of the f/m interaction submodel. Time in the transfer functions supplied to FEHM, the simulation code used to generate the transfer functions, is $t/R_f \tau_f$, which contains no information relevant to the extreme of long transport times, which approach $\tau_{comp}$. To make the method more robust, an improved nondimensionalization for time can be made as follows:

$$
\tilde{t} = \frac{t - R_f \tau_f}{\tau_{comp} - R_f \tau_f}
$$

(S-2)

Assuming, as is the case for the unsaturated zone transport model, that $\tau_{comp} \gg R_f \tau_f$, Eq. S-2 normalizes the breakthrough times to values in the approximate range of 0 and 1 in Figures S-1 and S-2. Because FEHM reads in time values of $t/R_f \tau_f$, an expression is required for converting these to dimensionless times given by Eq. S-2. This is done by first dividing the top and bottom of Eq. S-2 by $R_f \tau_f$:

$$
\tilde{t} = \frac{t/R_f \tau_f - 1}{\tau_{comp} / R_f \tau_f - 1}
$$

(S-3)

The FEHM input time $t/R_f \tau_f$ minus 1 is the numerator, so the requirement is to determine a relation involving the dimensionless parameters $(p_1, p_2, p_3)$ that can be used to express the denominator. For this the following relation is obtained from simple algebra from Equations 26 to 28 of the main paper:
\[ \frac{p_2p_3}{p_1} = \left( \frac{\tau_f}{b\theta_f} \right) \left( \frac{\theta_mB}{\tau_m} \right) \quad (S-4) \]

Alternatively, recognizing that \( b\theta_f \) and \( \theta_mB \) are proportional to the fluid mass in the fracture and matrix, respectively, then this expression reduces to:

\[ \frac{p_2p_3}{p_1} = \frac{f_m}{f_f} \quad (S-5) \]

An equivalent expression using the definition \( F_f = f_f/(f_f + f_m) \) is

\[ F_f = \frac{p_1}{p_1 + p_2p_3} \quad (S-6) \]

Returning to the definition of \( \tau_{comp} \) (Eq. S-1), the following algebraic manipulations can be performed:

\[ \tau_{comp} = \frac{R_fm_f + R_mE_m}{f_f + f_m} = F_f(R_f\tau_f + R_m\tau_m\frac{f_m}{f_f}) = F_fR_f\tau_f(1 + \frac{R_mE_m}{R_f\tau_f\frac{p_2p_3}{p_1}}) = F_fR_f\tau_f(1 + \frac{p_2}{p_1}) \quad (S-7) \]

This series of steps is conducted using Equations S-2, S-5, and the expression for \( p_3 \), along with the definitions \( F_f = f_f/(f_f + f_m) \), \( \tau_f = m_f/f_f \), and \( \tau_m = m_m/f_m \). Finally, the denominator \( \tau_{comp}/R_f\tau_f - 1 \) in Eq. S-3 is obtained through further algebra and the use of Eq. S-6:

\[ \frac{\tau_{comp}}{R_f\tau_f - 1} = \frac{p_2(1 - p_3)}{p_1 + p_2p_3} \quad (S-8) \]

The important point here is that the transformation of Eq. S-3 can be made by subtracting 1 from the input dimensionless time and dividing by the expression in Eq. S-8. This operation is performed by FEHM upon reading in the transfer function curves. Then, after the normalized
time for a particle $\tau$ is obtained in the particle tracking algorithm, Eq. S-2 is used to back out the dimensional value of time of the particle. This approach takes advantage of the self-similarity of the family of curves such as those in Figures S-1 or S-2. That is, even if the parameters $(p_1, p_2, p_3)$ at a given location in the model are not exactly those used to generate the transfer function, the use of the times $R/\tau_f$ and $\tau_{comp}$ from the model at a given grid cell provide a means for scaling the transfer function accordingly. Also, if a relatively large number of transfer function curves are used, it is likely that in most instances a curve fairly close to the parameters used in the transfer function will be found.

Despite the effectiveness of this method, there are a few cases for which special consideration needed to be made. This is because of the extraordinarily wide range of parameter values required to be covered in the total system performance assessment for license application model. As a result, the way in which the model handles some of the extreme values of certain parameters is through the use of special rules designed to achieve accuracy. These methods, described below, all call for the adjustment of one or more of the parameters $(p_1, p_2, p_3)$ at a given location in the model so as to yield the desired behavior. Details are given below.

**Low Diffusion Coefficient:** It is often desirable to lower the diffusion coefficient to extremely low values to examine this end-member case. Furthermore, colloids are low-diffusion species that require accuracy at low values of $D_m$. The most fool-proof way to do this is to not use the transfer function model at all, but instead to simply route the particles through the model with advection and dispersion only. However, if this option is not chosen, the model still must perform properly at the low end of diffusion. The diffusion coefficient only affects parameters $p_1$ and $p_2$, so the search algorithm needs to account for the fact that below a minimum value of
$D_m$ ($10^{-18}$ m$^2$/s in the code), the precise values of these parameters are not important. To ensure that the search algorithm locates a curve with the correct value of $p_3$, $p_1$, and $p_2$, which are assigned values that were actually used in the generation of the low-diffusion transfer function curves (see Table S-1), so that during the search, $p_1$ and $p_2$ are de-emphasized, and $p_3$ is, in essence, the only parameter considered. In doing so, this approach ensures that the code finds the portion of the parameter space with the correct values for $p_3$. If this is not done, the least squares method for selecting the correct transfer function curve can sometimes compensate for the extreme parameters chosen by selecting an undesirable part of the parameter space. With the approach just described, the method is forced to select a low-diffusion regime while obtaining the correct ratio of transport times in the fracture and matrix.

**High Matrix Sorption:** Similar to the case just described, an extremely large value of $R_m$ beyond the range used in the transfer functions causes problems for the search algorithm. When searching for the closest transfer function, the uncorrected method compensates for a large $R_m$ by selecting a fracture-dominated flow case to attain as low a value of $p_3$ as possible. Similarly, the calculation of $p_1$ is also affected. Thus, to correct for this case, the maximum value of $R_m$ used in the transfer function curve generation (1000) is used as an upper limit when searching for the correct transfer function curve. However, note that this does not mean that the matrix retardation is limited to that value in the particle tracking transport time calculation. Recall that the transfer function curves themselves are normalized using Eq. S-2, which includes $\tau_{\text{comp}}$. In contrast to the determination of the closest transfer function curve, the actual value of $R_m$ is used in computing $\tau_{\text{comp}}$, which results in a determination of transport time that is scaled by the actual
sorption set in the matrix. Thus the correction is applied only to find an appropriate transfer function curve, and the method for nondimensionalizing those curves ensures that an appropriately large transport time is reproduced for the case of high matrix sorption.

**Fracture-Dominated Flow:** The parameterization of the transfer function curves is based on a model that has some flow within both the fracture and matrix. When the flow is fracture dominated \((F_f > 0.9999)\), the details of the actual fraction of flow should be unimportant, since advective transport in the matrix should be negligible. However, without correction for cases where \(F_f > 0.9999\), the algorithm for finding the transfer function will inappropriately attempt to select curves with high \(R_m\) to compensate for the fact that transfer functions with extremely large \(F_f\) are not included. To correct for this problem, the code makes use of the following rearranged form of Eq. S-2:

\[
p_3 = \frac{p_1 (1 - F_f)}{p_2 F_f}
\]  

(S-9)

When \(F_f > 0.9999\), the code uses 0.9999 and the values of \(p_1\) and \(p_2\) to compute \(p_3\) for the purposes of selecting the transfer function curve. This assures that a fracture-dominated transfer function is chosen with appropriate values for the other diffusion and sorption parameters.

**Matrix-Dominated Flow:** For this extreme, it is desirable to bypass the transfer function method altogether, since the transport time is trivially found to be \(\tau_m \tau_m\). Allowing the transfer function algorithm to be used for this case causes problems because the normalization procedure implicitly assumes that the matrix transport time is longer than the fracture transport time. To
cover the special case of essentially no flow in the fracture, the transport time is assigned a value of \( \tau_{comp} \), which reduces to \( R_m \tau_m \) under these conditions.

*Treatment of Wetted Surface Area:* The Active Fracture Model (AFM) developed by Liu et al. (1998) is formulated on the basis that only a fraction of the fractures flow. This requires that adjustments be applied to the interface area and the mean spacing between flowing fractures. These adjusted parameters can then be used in the UZ transport model calculations. Examining the individual terms of the mass balance for the fracture derived in the main paper, the accumulation term (Eq. 1) is unchanged by the AFM, because it is based on the storage volume in the fracture, as well as sorption parameters. Storage volumes in the dual permeability flow fields are fully defined by the fracture volume fractions and the fluid saturations in the fracture continuum. Fluid saturations are model output from the flow simulations, and no further correction for transport is required for the accumulation term. Likewise, the Darcy velocity in the advection term (Eq. 2) is fully defined by the flux through the fracture continuum, so no AFM corrections are required for advection either. The diffusion term (Eq. 3) consists of a flux

\[
\theta_m D_m \frac{\partial C_m}{\partial x} \bigg|_{z=b}
\]

times an interfacial area, which on geometrical grounds for the simple geometry of the transfer function model is \( d \Delta z \). This interfacial area term, according to the AFM, should be reduced to account for the fact that not all fractures are flowing. Liu et al. (1998) gives the following reduction factor for correcting the advective flux term [note: nomenclature from the report by Liu et al. (1998) is used in this equation]:

\[
R = \left( \frac{A_{fm,a}}{A_{fm}} \right) \left( \frac{n_{f,a}}{n_f} \right) \left( \frac{d}{d_a} \right)
\]

(S-10)
Although (Liu et al., 1998) refer to $R$ as the F-M interface area reduction factor, it is clear from their derivation that the term represents the ratio of the fluxes for the uncorrected and corrected cases, correcting for both the interface area and the transport length scale associated with the distance between the flowing fractures (the third term on the right hand side of this equation).

Therefore, in the present model, AFM-based adjustments should be applied to both the interface area and the spacing $B$. The term $d/d_a$ is the adjustment to the fracture spacing, and is accounted for by adjusting the spacing $B$ in the FEHM transport simulations using the following relation (Eq. 17 of Liu et al. 1998):

$$ \frac{d}{d_a} = S^\gamma_e $$  \hfill (S-11)

Thus, the geometric spacing is divided by $S^\gamma_e$ to obtain the spacing between flowing fractures.

The interface-area portion of the adjustment consists of the first two terms on the right hand side of Eq. S-10, the first to account for the reduction in wetted area within an individual fracture, and the second to account for the reduction in area associated with a smaller number of wetted fractures. This term can be related to the AFM parameters using Equations 13 and 14 of Liu et al. (1998):

$$ \left( \frac{A_{fm,a}}{A_{fm}} \right) \left( \frac{n_{f,a}}{n_f} \right) = S^{1-\gamma}_e S^\gamma_e = S_e $$  \hfill (S-12)

To implement this area reduction term in the current model, the geometrically defined aperture $b$ is divided by $S_e$. The adjustment to $b$ is for convenience, and actually arises from the need to adjust the interface area in the fracture transport equation. These adjustment factors have been incorporated into the current model so that for given AFM model parameters, $B$ and $b$, input as
geometrically defined parameters, are converted to hydrologic parameters for use in the transfer function methodology.

**References Cited in this Supplement**
