EFFICIENT PARTICLE TRACKING ALGORITHMS FOR SOLUTE TRANSPORT IN FRACTURED ROCK WITH ABSORPTION AND MATRIX DIFFUSION

by

DYLAN LOWELL KLEIN

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirement for the degree of
Master of Science
Department of Applied Mathematics
2013

This thesis entitled:

“Efficient Particle Tracking Algorithms
for Solute Transport in Fractured Rock
with Absorption and Matrix Diffusion”
written by Dylan Lowell Klein
has been approved for the Department of Applied Mathematics

____

Harihar Rajaram

____

Anne Dougherty

Date:____________________

The final copy of this thesis has been examined by the signatories, and we
Find that both the content and the form meet acceptable presentation standards
Of scholarly work in the above mentioned discipline.

Abstract

Klein, Dylan Lowell (M.S., Applied Mathematics)

"Efficient Particle Tracking Algorithms for Solute Transport in Fractured Rock with Absorption and Matrix Diffusion"

Thesis directed by Professor Harihar Rajaram

In this paper, we study solute transport in an individual fracture and the surrounding porous rock. Specifically, we consider a parallel-plate model of a single fracture that allows for the diffusion of solute within the matrix and the adsorption of solute to the fracture walls. We developed two stochastic particle-tracking methods to numerically solve for the concentration of the fracture model. The first is the hi-res method which captures the solute dynamics at a micro-scale. The second algorithm we develop, the upscaled method, captures the large-scale dynamics of the system at vastly reduced computational cost. We verified the accuracy of these methods by comparing their results to numerical results from the literature. We also compared the efficiency of the developed particle tracking methods to an existing particle tracking method from the literature in the case of no interface absorption.

Acknowledgements

First and foremost, I wish to thank my advisor, Hari Rajaram, for his guidance, insight and patience. Hari provided me with both support and freedom. I am truly fortunate to have had him as my advisor. I would also like to thank the other two members of my thesis committee, Professors Anne Dougherty and Jem Concoran, for their thoughtful feedback and encouragement. Finally, I want to thank my family for the support they provided me, throughout this process and throughout my entire life. Without them, I would never have started this thesis, much less finished it.

This research was supported by the Department of Energy (DE-FG02-08ER64661) and the National Science Foundation Sustainability Research Network (CEBET1240584).

Contents

List of Tables

List of Figures

1 Introduction

1.1 Background

The transport of solute in porous and fractured media is major topic of research for petroleum engineers, environmental engineers and hydrologists. Contexts in which this problem appears include:

- **Underground waste disposal:** Understanding how the contaminant can move underground is critical in making sure it is properly contained. This category notably includes the disposal of nuclear fuel waste.
- **Natural gas mining:** Soluble ores and natural gases are sometimes recovered by in-situ leaching. This process involves fracturing the rock, pumping a leeching solution into the deposit and then recovering the solution for processing. Understanding the solute's movement in a leeching solution is essential to recovering the dissolved ore or gas.

Particle tracking methods are numerical methods that have been used successfully to simulate solute transport in fractured media [LaBolle et al., 1996]. Particle tracking methods have several features that make them well-suited for this problem. Particle tracking methods avoid the problem of numerical dispersion, an issue grid-based methods encounter in dealing with systems dominated by advection. Furthermore, particle tracking methods are parallel algorithms; the computation can easily be distributed over multiple processors. However, particle tracking methods appearing in the literature do not allow for solutes that react with the interface separating the fractures from the surrounding rock matrix, thereby adhering to it. In this paper, we develop a particle tracking methods for a fracture model that includes interface absorption.

1.2 Problem Description

We shall develop a particle tracking method for a parallel-plate model of a single fracture:

$$\frac{\partial C_f}{\partial t} = -u\frac{\partial C_f}{\partial x} + D_f\left(\frac{\partial^2 C_f}{\partial x^2} + \frac{\partial^2 C_f}{\partial y^2}\right) \qquad for\ |y| < L \tag{1.2.1a}$$

$$\frac{\partial C_m}{\partial t} = D_m\left(\frac{\partial^2 C_f}{\partial x^2} + \frac{\partial^2 C_f}{\partial y^2}\right) \qquad for\ |y| > L \tag{1.2.1b}$$

$$C_f = C_m \qquad for\ y = \pm L \tag{1.2.1c}$$

$$\pm k_a\frac{\partial C_f}{\partial t} = \phi D_e\frac{\partial C_m}{\partial y} - D_f\frac{\partial C_f}{\partial y} \qquad for\ y = \pm L \tag{1.2.1d}$$

where

$$D_m = \frac{\phi \cdot D_e}{\phi + \rho_b k_d} \tag{1.2.2}$$

The name and units of this model's parameters are presented in Table 1. The geometry of the modeled system, the parallel-plate configuration, is shown in Figure 1. The parallel-plate geometry is used frequently in the literature [Maloszewski and Zuber, 1990; LaBolle, Fogg and Tompson, 1996]. However, this model is different from many of the models in the literature in that it includes linear solute absorption on the interface surface itself. The scope of this model is not limited to the fracture itself and allows the solute to diffuse into water permeating the surrounding rock matrix. We now familiarize the reader with the solute transport dynamics represented by each equation.

Equation (1.2.1a) governs the solute concentration within the fracture. The fracture occupies a straight band centered on the x-axis. This equation reflects that the solute disperses through the fluid with diffusivity D_f while being carried along with the fluid flow in the x-direction at speed u.

Equation (1.2.1b) governs the solute concentration in the fluid permeating the matrix. The matrix surrounding the fracture is composed of rock of porosity ϕ; fluid fills the gaps. The spread of solute in the fluid is slowed down by tortuous geometry of these gaps. As a consequence, the effective diffusivity for in-solution solute, D_e, is typically smaller than the diffusivity within the fracture, D_f, by an order of magnitude or more. The mass density of this in-solution solute then is the product of the volume fraction filled with fluid and the fluid's solute concentration: $\phi \cdot C_m$. Solute in the matrix can get stuck to exposed rock surface where the particle is temporarily held in place. The volume density of exposed rock surface area is denoted by ρ_b. These rock surfaces have a solute surface density proportional to the fluid concentration: $k_d \cdot C_m$. It follows that the volume density of this out-of-solution solute is the product of these two terms: $\rho_b k_d \cdot C_m$. The total solute mass density in the fracture is thus:

$$\rho_m = \phi \cdot C_m + \rho_b k_d \cdot C_m \tag{1.2.3}$$

The matrix's effective diffusive constant, D_m, is obtained by considering how particles adhering to matrix rock inhibits diffusion. It follows from the mass densities we just obtained that, in the matrix, the fraction of solute that is in-solution is $\frac{\phi}{\phi+\rho_b k_d}$. As the solution and rock surface are in solute equilibrium, we reason that all solute mass spends this fraction of time diffusing in solution with diffusivity D_e. The solute in the matrix therefore disperses with an overall effective diffusivity of $\frac{\phi \cdot D_e}{\phi+\rho_b k_d}$ or D_m.

The next two equations are boundary conditions for the interface between the two regions. The interface occupies two parallel lines. Equation (1.2.1c) requires that the solute concentration at the interface be continuous. This must be true because the solutions are in direct contact with each other at the boundary.

Symbol	Name	Units	Further Notes
C_f	Fracture solute concentration	${}^{kg}/_{m^3}$	-
C_m	Matrix solute concentration	${}^{kg}/_{m^3}$	-
L	Fracture half-width	m	-
D_f	Fracture diffusivity	${}^{kg}/_{m^3}$	-
D_m	Matrix retarded diffusivity	${}^{kg}/_{m^3}$	-
D_e	Matrix effective diffusivity	${}^{kg}/_{m^3}$	-
k_a	Interface surface distribution coefficient	m	Interface surface density per solution volume density
ϕ	Porosity of rock	$unitless$	Fraction of matrix volume unoccupied by rock
ρ_b	Bulk density	${}^{1}/_{m}$	Matrix rock surface area per volume
k_d	Distribution constant	m	Matrix rock surface density per solution volume density
u	Fracture fluid flow velocity	${}^{m}/_{s}$	-

Table 1: The name, symbol and units of the parameters of fracture model parameters appearing in equations (1.2.1). Clarifying notes are also included for some parameters.

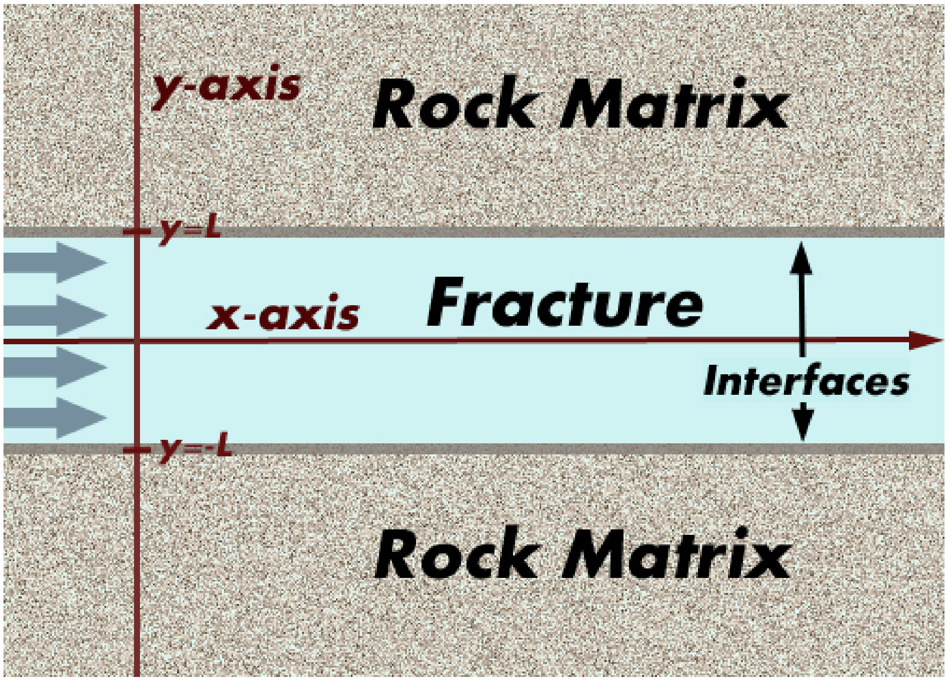


Figure 1: Diagram of the parallel-plate fracture system.

Equation (1.2.1d) enforces mass conservation at each interface. To rederive this condition, we consider the net mass flowing from each region onto the interface. At the interface at $y = L$, mass flows from fracture towards the interface at a rate of $-D_f \frac{\partial C_f}{\partial y}$ and flows from the matrix to the interface at a rate of $\phi D_e \frac{\partial C_m}{\partial y}$. The rate of total mass flow on to the interface itself is then the sum of these terms: $\phi D_e \frac{\partial C_m}{\partial y} - D_f \frac{\partial C_f}{\partial y}$. This must be equal to the rate at which mass accumulates on the interface's surface. The interface surface is in equilibrium with the surrounding solution and so has a surface density proportional to the solution concentration. By definition, the constant of proportionality is k_a; the surface density of solute on the interface is therefore $k_a \cdot C_f$. The rate of mass accumulation on the interface is equal to the time-derivative of this surface density: $k_a \frac{\partial C_f}{\partial t}$. The mass conservation equation equates this rate of solute accumulation with the rate of net incoming solute flow. The situation at the other interface is analogous; the only difference is that signs of the mass flows are flipped because the relative positions of the matrix and fracture are switched.

2 Particle Tracking Survey

In this section, we provide an overview of how particle tracking methods work. We then demonstrate the development of particle tracking methods for some example problems.

2.1 Particle Tracking Principles

Particle tracking methods operate by approximating continuous mass density functions as a set of discrete mass particles. These particles are then stochastically advanced forward by many time steps to the desired end time. A histogram of the particles then serves as an approximation of the system's mass density at the end time.

Particle tracking methods have the advantage of being parallel algorithms; one can effortlessly split the computational workload into independent subproblems. One simply simulates a separate group of particles on each processor and then combines the simulated particle sets afterwards. Consequently, the run time of particle tracking simulations can be reduced dramatically by using multiple processors. Particle tracking methods also don't suffer from numerical dispersion, a problem for grid-based methods in modeling systems dominated by advection.

A particle tracking method operates by simulating particles with the following properties: At all times, the probability density function of a particle's position must be proportional to the modeled mass density function. The total mass of the particle set must be the total mass in the modeled system. Finally, the number of particles simulated must be large; the larger the number of particles, the more accurate the results.

The particles can be initialized as follows. We first choose an arbitrary number of particles to be in the set. We then set the mass of each particle as necessary to ensure the particle's total mass is equal to the initial mass in the system. The particles' initial locations are simulated by normalizing the initial mass density function and generating random variables with resulting probability density function or pdf.

With the particle set initialized, the particles must all be advanced to the desired end time. To advance a particle forward by a time step, we first convert it to an analogous mass density function: a Dirac delta function at the particle's starting location scaled by the particle's mass. This mass density function is then analytically evolved forward by the time step's length using the governing equations of the modeled system. We then normalize this mass density function and interpret the result as a pdf of a random variable. We simulate the random variable having that pdf and take the result to be the particle's new location. This whole process must be repeated until every particle is at the end time.

Once all the particles have been advanced to the end time, the entire particle set must be converted back to a mass density function. As information is lost in representing a continuous function as a discrete set of particles, there is no way to perfectly recover the final mass density function. However, taking a histogram of the particle mass is a simple and fairly effective method of approximating the final mass density function at a set of points.

The second step outlined, of time-evolving a point mass density function and then simulating the random particle with the resulting pdf, is the key component. With a large number of time steps and particles, this procedure must be performed a vast number of times. It is therefore important to be able to carry it out as accurately and efficiently as possible. In this thesis, we focus on developing algorithms to accomplish this goal for the fracture model at hand.

2.2 Example Method: Simple Diffusion

We demonstrate the development of a particle tracking method on a simple problem: the uniform diffusion of mass on a line.

$$\frac{\partial \rho}{\partial t} = D\frac{\partial^2 \rho}{\partial x^2} \tag{2.2.1}$$

We consider a particle of unit mass at the arbitrary position x_0. We shall advance the particle forward forward by time step $\triangle t$. To do this, we first convert the particle to a form compatible with the governing differential equations: a mass density function. A particle is a point mass so the corresponding mass density function is a Dirac delta function. The initial mass density function is thus a Dirac delta function at x_0:

$$\rho(x, t = 0) = \delta_{x_0}(x) \tag{2.2.2}$$

The time evolution of this initial mass density after time step Δt is [Einstein, 1956]:

$$\rho(x, t = \triangle t) = \frac{1}{4\pi D \cdot \triangle t} e^{-\frac{(x-x_0)^2}{4D\cdot\triangle t}} \tag{2.2.3}$$

We now convert this time-evolved mass density function back to a discrete particle. The mass density function is spread out but the particle can only be at one location; we therefore choose the particle's new location probabilistically. We let the probability of the particle going to any given location be proportional to the mass density function at that point. As the initial conditions were of a point mass and this system obeys mass conservation, the time-evolved mass density function is already normalized. The pdf of the particle's new location, X, is thus equal to the time-evolved mass density function we just found:

$$f_X(x) = \frac{1}{4\pi D \cdot \triangle t} e^{-\frac{(x-x_0)^2}{4D\cdot\triangle t}} \tag{2.2.4}$$

We now observe that this function is a normal distribution and is thus the pdf of the normal random variable $N(x_0, 2D \cdot \Delta t^2)$. We consequently model the particle's position after the time step as this random variable:

$$X = x_0 + \sqrt{2D \cdot \Delta t} Z \tag{2.2.5}$$

where Z is a standard normal random variables. This formula allows for us to advance an arbitrary placed particle forward in time. The formula thus constitutes a particle tracking method for this system. This is a well-known result that was famously used by Einstein to link the macroscopic process of diffusion and the Brownian motion of individual particles [1956].

2.3 Example Method: Perfectly Absorbing Boundary

We provide one more example of the development of a particle tracking method. This time we consider a more complicated problem: uniform diffusion on the positive half-line with a perfectly absorbing boundary at the origin:

$$\frac{\partial \rho}{\partial t} = D \frac{\partial^2 \rho}{\partial x^2} \qquad for\ x > 0 \tag{2.3.1a}$$

$$\rho = 0 \qquad for\ x = 0 \tag{2.3.1b}$$

We consider a particle at an arbitrary location in the system. As the system's domain is x such that $x \geq 0$, the particle's initial position, x_0, must be non-negative. Instead of having a fixed length time step, we will use a variable length time step to demonstrate how they work. We define the time step length be the particle's first passage time to the perfectly absorbing boundary at the origin. The location of the particle after the time step is already known: the particle will be absorbed after the time step due to this definition of the time step's length. Instead, we must model the length of the time step. To achieve this, we first convert the particle into a mass density function: a Dirac delta function at the particle's coordinates. The initial mass density function is then:

$$\rho(x, t = 0) = \delta_{x_0}(x) \tag{2.3.2}$$

We now must find the distribution in time of mass being absorbed. We first find the mass density of the system after time t. This is readily obtained by using the method of images in conjunction with equation (2.2.3):

$$\rho(x, t) = \frac{1}{4\pi D \cdot t} e^{-\frac{(x-x_0)^2}{4D \cdot \Delta t}} - \frac{1}{4\pi D \cdot t} e^{-\frac{(x+x_0)^2}{4D \cdot \Delta t}} for\ y \geq 0 \tag{2.3.3}$$

Unlike in the previous example, the system's mass is not conserved: mass can leave the system through the absorbing boundary. Because of this, the time-evolved mass density function is not normalized. We can integrate the mass density function over the system to get the total mass remaining in the system as a function of time:

$$M_{system}(t) = erf\left(\sqrt{\frac{x_0^2}{2D}\cdot\frac{1}{2t}}\right) \tag{2.3.4}$$

By process of elimination, any mass no longer in the system must have been absorbed. The mass absorbed is then the equal to the original unit of mass in the system minus the amount of mass remaining in the system:

$$M_{absorbed}(t) = 1 - erf\left(\sqrt{\frac{x_0^2}{2D}\cdot\frac{1}{2t}}\right) = erfc\left(\sqrt{\frac{x_0^2}{2D}\cdot\frac{1}{2t}}\right) \tag{2.3.5}$$

We now take the time derivative of the absorbed mass to find the rate at which mass is absorbed:

$$Q_{absorption} = \frac{dM_{absorbed}}{dt} = \sqrt{\frac{x_0^2}{2D}}\cdot\frac{e^{-\frac{x_0^2}{D}\cdot\frac{1}{2t}}}{t^{3/2}} \tag{2.3.6}$$

This expression gives the time-distribution of mass absorption. As a particle must be instantaneously absorbed at some specific time, we model the particle's time of absorption probabilistically. We let the probability of the particle being absorbed at a given time be proportional to that time's mass absorption rate. Note the absorption rate integrates to one, indicating that no mass remains in the system forever. The pdf of the particle's absorption time, T, is consequently equal to the absorption rate:

$$f_T(x) = \sqrt{\frac{x_0^2}{2D}}\cdot\frac{e^{-\frac{x_0^2}{2D}\cdot\frac{1}{2t}}}{t^{3/2}} \tag{2.3.7}$$

At this point, we observe that the above distribution is a Levy distribution. More specifically, it is the pdf of the random variable $Levy(0, \frac{x_0^2}{2D})$. This random variable thus models the time that the particle first hits the origin and gets absorbed. We use the fact that this Levy distribution can be simulated as $\frac{x_0^2}{2D}\cdot\frac{1}{z^2}$ where Z is a normal random variable [Chambers et al., 1976]. The length of the time step, $\triangle t$, can therefore be simulated as follows:

$$\triangle t = \frac{x_0^2}{2D}\cdot\frac{1}{Z^2} \tag{2.3.8}$$

where Z is a standard normal random variable. Recall that after the time step, the particle has reached the origin and been absorbed by definition. We have thus modeled both the time and location of the particle after the time step. Consequently, this gives all the results needed for the particle tracking method. This is a well-known result from diffusion theory [Jacobs, 1976].

3 Fracture Model Reformulation

In this section, we reformulate the mass transport equations to be in terms of mass, a form more natural for deriving particle tracking methods. We then simplify the full fracture model to obtain subsystems that are easier to develop particle tracking methods for. Particle tracking methods developed for these subproblems can later be extended to the full system.

3.1 Mass Density Formulation

Particle tracking methods discretize a continuous mass distribution into discrete particles of mass. The particles must denote amounts of mass to ensure that the number of particles is conserved over time. This makes it natural to work in terms of mass density functions.

To convert concentration functions to mass density functions, we use physical reasoning. All solute in the fracture is in-solution and the fracture is completely filled with fluid. Therefore, within the fracture, the mass density is equal to the mass concentration:

$$\rho_f(x, y, t) = C_f(x, y, t) \tag{3.1.1}$$

Equation (1.2.3) gives the mass density in the matrix. We define a new parameter as follows for convenience:

$$\varphi = \phi + \rho_b k_d \tag{3.1.2}$$

With this new parameter, the fracture mass density can be expressed as simply:

$$\rho_m(x, y, t) = \varphi \cdot C_m(x, y, t) \tag{3.1.3}$$

Lastly, some solute mass will be attached to the interface between the two regions. The surface density of mass is defined to be $k_a \cdot C_f$ by k_a's definition. As the fracture concentration is equal to the fracture mass density by equation (3.1.1), we can express the interface mass density as follows:

$$\rho_i(x, y, t) = k_a \cdot \rho_f(x, y, t) \cdot [\delta_{-L}(y) + \delta_L(y)] \tag{3.1.4}$$

These three regional mass distributions together give the mass density in each region:

$$\rho(x, y, t) = \begin{cases} \rho_f(x, y, t) & |y| < L \\ \rho_m(x, y, t) & |y| > L \\ \rho_i(x, y, t) & |y| = L \end{cases} \tag{3.1.5}$$

We combine equations (3.1.1), (3.1.3), (3.1.4) and (3.1.5) to give a single formula allowing concentration functions to be converted into mass density functions:

$$\rho(x,y,0) = \begin{cases} C_f(x,y,0) & |y| < L \\ \varphi \cdot C_m(x,y,0) & |y| > L \\ k_a \cdot C_f(x,y,0) \cdot \delta_L(|y|) & |y| = L \end{cases}$$

Inverting this equation gives a method for converting a mass density function to a concentration function:

$$C_f(x,y,t) = \rho(x,y,t) \qquad for \ |y| < L \tag{3.1.6a}$$

$$C_m(x,y,t) = \frac{1}{\varphi}\rho(x,y,t) \qquad for \ |y| > L \tag{3.1.6b}$$

Finally, we substitute equations (3.1.1) and (3.1.3) into the original transport equations, equations (1.2.1), to obtain the mass transport equations in terms of mass density:

$$\frac{\partial \rho_f}{\partial t} = -u\frac{\partial \rho_f}{\partial x} + D_f\left(\frac{\partial^2 \rho_f}{\partial x^2} + \frac{\partial^2 \rho_f}{\partial y^2}\right) \qquad for\ |y| < L \tag{3.1.7a}$$

$$\frac{\partial \rho_m}{\partial t} = D_m\left(\frac{\partial^2 \rho_m}{\partial x^2} + \frac{\partial^2 \rho_m}{\partial y^2}\right) \qquad for\ |y| > L \tag{3.1.7b}$$

$$\rho_f = \frac{1}{\varphi}\rho_m \qquad for\ y = \pm L \tag{3.1.7c}$$

$$\pm k_a\frac{\partial \rho_f}{\partial t} = D_m\frac{\partial \rho_m}{\partial y} - D_f\frac{\partial \rho_f}{\partial y} \qquad for\ y = \pm L \tag{3.1.7d}$$

3.2 Single-Interface Subproblem

Before developing a particle tracking method for the full system, we will develop methods for simpler subsystems. We will later work to extend these results back to to the full problem.

Let us consider the mass-density formulation of the transport equations we just obtained, equations (3.1.7). As the two interfaces run parallel to each other, a particle will always be at least distance L from one of the interfaces. By keeping the time steps sufficiently small, it can be made arbitrarily unlikely that a particle will travel this distance in a single time step. It is then reasonable to ignore the interface farther from the particle's current location. We therefore will consider a single-interface subsystem. Without loss of generality, we will consider a particle in the proximity of the interface at $y = L$ and ignore the interface at $Y = -L$. Figure 2 shows a diagram of the two-dimensional single-interface subsystem. After shifting the coordinates so that the remaining interface lies on the x-axis, the resulting single-interface subproblem is governed by the following equations:

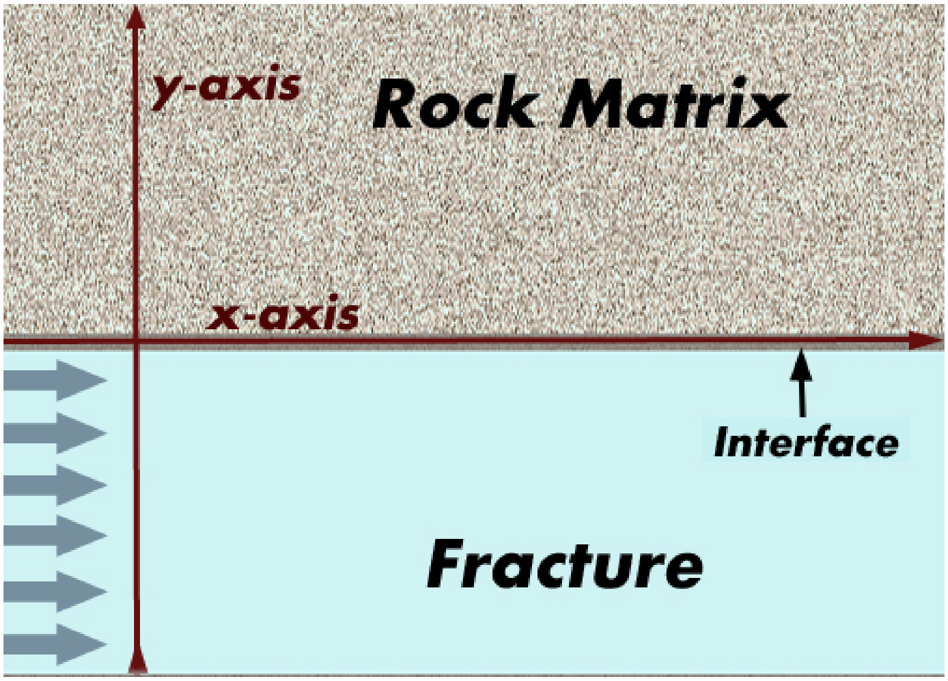


Figure 2: Diagram of the (two-dimensional) single-interface fracture subsystem.

$$
\begin{aligned}
\frac{\partial \rho_f}{\partial t} =& - u \frac{\partial \rho_f}{\partial x} + D_f \left(\frac{\partial^2 \rho_f}{\partial x^2} + \frac{\partial^2 \rho_f}{\partial y^2} \right) && for\, y < 0 && \text{(3.2.1a)} \\
\frac{\partial \rho_m}{\partial t} =& D_m \left(\frac{\partial^2 \rho_m}{\partial x^2} + \frac{\partial^2 \rho_m}{\partial y^2} \right) && for\, y > 0 && \text{(3.2.1b)} \\
\rho_f =& \frac{1}{\varphi} \rho_m && for\, y = 0 && \text{(3.2.1c)} \\
k_a \frac{\partial \rho_f}{\partial t} =& D_m \frac{\partial \rho_m}{\partial y} - D_f \frac{\partial \rho_f}{\partial y} && for\, y = 0 && \text{(3.2.1d)}
\end{aligned}
$$

The interface mass density for the subproblem is given by:

$$\rho_i(x, y, t) = k_a \cdot \rho_f(x, y, t) \cdot \delta_0(y) \tag{3.2.2}$$

The global mass density for the single-interface subproblem then is:

$$\rho(x, y, t) = \begin{cases} \rho_f(x, y, t) & y < 0 \\ \rho_m(x, y, t) & y > 0 \\ \rho_i(x, y, t) & y = 0 \end{cases} \tag{3.2.3}$$

3.3 One-Dimensional Single-Interface Subproblem

We next observe that these mass transport equations are invariant with respect to shifts along the x-axis. As a result of this, we can model the y-coordinate of particle without considering the x-coordinate at all. The x-coordinate can be modeled at a later point. We remove the x-coordinate from equations (3.2.1) by integrating this system of differential equations from $x = -\infty$ to $x = \infty$ in a process called volume averaging. The result is the one-dimensional single-interface subsystem:

$$
\begin{aligned}
\frac{\partial \rho_f}{\partial t} =& D_f \frac{\partial^2 \rho_f}{\partial y^2} && for\, y < 0 && \text{(3.3.1a)} \\
\frac{\partial \rho_m}{\partial t} =& D_m \frac{\partial^2 \rho_m}{\partial y^2} && for\, y > 0 && \text{(3.3.1b)} \\
\rho_f =& \frac{1}{\varphi} \rho_m && for\, y = 0 && \text{(3.3.1c)} \\
k_a \frac{\partial \rho_f}{\partial t} =& D_m \frac{\partial \rho_m}{\partial y} - D_f \frac{\partial \rho_f}{\partial y} && for\, y = 0 && \text{(3.3.1d)}
\end{aligned}
$$

The interface mass density for this subsystem is given by:

$$\rho_i(y,t) = k_a \cdot \rho_f(y,t) \cdot \delta_0(y) \tag{3.3.2}$$

The global mass distribution for this subsystem is:

$$\rho(x,y,t) = \begin{cases} \rho_f(y,t) & y < 0 \\ \rho_m(y,t) & y > 0 \\ \rho_i(y,t) & y = 0 \end{cases} \tag{3.3.3}$$

4 Single-Interface Particle Tracking Methods

We now develop a particle tracking method for the one-dimensional single-interface subsystem described by equations (3.3.1), (3.3.2) and (3.3.3). We shall consider particles starting in the fracture, matrix, and interface separately. We then will append a method to simulate a particle's new x-coordinate to get a particle tracking method for the two-dimensional single-interface system.

4.1 Particles Starting in Fracture or Matrix

We begin with particles starting in the fracture. We consider a particle starting at an arbitrary location in the fracture of the one-dimensional single-interface subsystem defined by equations (3.3.1), (3.3.2) and (3.3.3). For the particle to begin in the fracture region, the particle's initial y-coordinate, y_0, must be negative. We convert the particle to a mass density function to get the initial conditions as a mass density function:

$$\rho(y, t = 0) = \delta_{y_0}(y) \tag{4.1.1}$$

We translate these initial conditions into regional mass density functions compatible with the mass transport equations via equation (3.3.3):

$$\begin{aligned} \rho_f(y, t = 0) =& \delta_{y_0}(y) \\ \rho_m(y, t = 0) =& 0 \\ \rho_i(y, t = 0) =& 0 \end{aligned}$$

To avoid dealing with the other regions at this point, we will use a variable length time step that always ends before the particle spends time outside the fracture. We define the length of the time step to be $\triangle t_{max}$ or until the particle hits the interface location, whichever is shorter. With this defintion of the time step length, the time step always ends before the particle spends time outside the fracture. As a consequence, the other regions' properties have no effect on the particle's path during the time step. We can therefore treat the entire system as having the fracture's properties for the time step without affecting the results during the time step:

$$\frac{\partial \rho_f}{\partial t} = D_f \frac{\partial^2 \rho_f}{\partial y^2} \tag{4.1.2a}$$

$$\rho_f(x, y, t = 0) = \delta(y - y_0) \tag{4.1.2b}$$

Note that even though the interface doesn't appear in this system, the time step still must end if a particle reaches the interface location at the origin. We will now model the particle's position after the maximum time step of $\triangle t_{max}$ without worrying about whether the particle hits the interface during this period or not. The diffusion of a point mass in a region of homogenous diffusivity results in a normal distribution [Einstein, 1956]:

$$\rho_f(y,t) = \frac{1}{\sqrt{4\pi D_f t}} e^{-\frac{(y-y_0)^2}{4D_f t}} \tag{4.1.3}$$

To simulate the particle's y-coordinate after a time step of $\triangle t_{max}$, we interpret this mass density function at $t = \triangle t_{max}$ as a pdf:

$$f_{Y_{max}}(y,t) = \frac{1}{\sqrt{4\pi D_f t}} e^{-\frac{(y-y_0)^2}{4D_f t}} \tag{4.1.4}$$

As this is a normal distribution, we can simulate the particle's y-coordinate at $t = \triangle t_{max}$ in the altered system as the following normal random variable:

$$Y_{max} = y_0 + \sqrt{2D_f \cdot \triangle t_{max}} \cdot Z \tag{4.1.5}$$

where Z is a standard normal random variable.

We now model whether the particle's path crossed the interface before $t = \triangle t_{max}$, an event we denote as "*Hitting*". If the particle ends up outside the fracture, we can conclude that particle must have crossed the interface by path continuity. Even if the particle ends up on the same side of the interface, there is still a chance the particle crossed the interface at some point during the time step. This is equivalent to the probability that a particle experiencing pinned Brownian motion hits a given point while in transit. This probability is shown by Chang to be equal to $e^{-\frac{y_0 \cdot Y_{max}}{D_f \cdot \triangle t_{max}}}$ [2007]. The probability of the event of "*Hitting*" occurring given the particle's end location, Y_{max}, is therefore:

$$P(Hitting|Y_{max} = y) = \begin{cases} e^{-\frac{y_0 \cdot y_{max}}{D_f \cdot \triangle t_{max}}} & y < 0 \\ 1 & y \geq 0 \end{cases} \tag{4.1.6}$$

Given this probability, we can model the event of the particle hitting the interface during the time step as follows:

$$\text{The event of "Hitting" occurs if and only if } \left(Y_{max} > 0\right) \text{ or } \left(U < e^{-\frac{y_0 \cdot Y_{max}}{D_f \cdot \triangle t_{max}}}\right) \tag{4.1.7}$$

where U is a standard uniform random variable.

If the particle is found to have not hit the interface, we simply advance the particle forward in time by $\triangle t_{max}$ to the simulated end position Y_f. However, if the particle has hit the interface, then the time step ends at that time of first hitting and the particle ends the time step on the interface. In this case, we must model the particle's first hitting time conditioned on that first hitting time being less than or equal to $\triangle t_{max}$. The first hitting time for a particle undergoing uniform diffusion is known to be distributed as a Levy distribution with parameters $\mu = 0$ and $c = \frac{y_0^2}{2D_m}$ [Jacobs, 2010]. We showed a derivation of this in section (2.3). A Levy distributed random variable can be simulated as follows [Chambers et al., 1976]:

$$\tau = \frac{y_0^2}{2D_f} \cdot \frac{1}{|Z|^2} \tag{4.1.8}$$

where Z is a normal standard random variable. We condition this variable to be less than or equal Δt_{max} to get the desired first hitting time for this problem:

$$\tau_{\triangle t_{max}} = \frac{y_0^2}{2D_f} \cdot \frac{1}{Z^2} \Bigg| \frac{y_0^2}{2D_f} \cdot \frac{1}{Z^2} \leq \triangle t_{max} \tag{4.1.9}$$

where Z is a normal standard random variable. The condition on the first hitting time can be algebraically manipulated into an equivalent condition on the standard normal random variable Z:

$$\tau_{\triangle t_{max}} = \frac{y_0^2}{2D_f} \cdot \frac{1}{Z^2} \Bigg| |Z| \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \tag{4.1.10}$$

where Z is a normal standard random variable. Because the first hitting time is invariant with respect to the sign of Z, we can restrict Z to be positive without altering the first hitting time distribution:

$$\tau_{\triangle t} = \frac{y_0^2}{2D_f} \cdot \frac{1}{Z^2} \Bigg| Z \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \tag{4.1.11}$$

where Z is a standard normal random variable. This leaves the task of generating a standard random variable that is so conditioned. A brute-force approach would be to repeatedly generate standard normal random variables until one fell into the desired range. However, it can take many tries for a normal random variable to fall above the threshold, especially if that threshold is high. We instead utilize the fact that, in this case where the particle hits the interface during the time step, the conditioned normal random variable is equal in distribution to a function of Y_{max}:

$$\left[\frac{|Y_{max}| + |y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \Bigg| Hitting \right] \underset{dist.}{=} \left[Z \Bigg| Z \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \right] \tag{4.1.12}$$

We prove this equality in Appendix A. We substitute the function of Y_{max} on the left-hand side of the above in for Z in equation (4.1.11) and simplify to get:

$$\tau_{\triangle t} = \triangle t_{max} \cdot \left(\frac{|y_0|}{|Y_{max}| + |y_0|} \right)^2 \tag{4.1.13}$$

This models the time of first hitting the interface in the event that the particle is found to have hit the interface. We now have all the results necessary to advance a particle in the fracture.

Advancing a Particle in the Fracture (One-Dimensional Single-Interface Subproblem)

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.

2. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_f \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.

3. Model whether the particle has hit the interface location during the time step. If $Y_{max} \geq 0$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{y_0 \cdot Y_{max}}{D_f \cdot \triangle t}}$.

4. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0|}{|Y_{max}| + |y_0|} \right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = 0$, the location of the interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after the time step is $Y = Y_{max}$.

5. Advance the particle's time to $T = t_0 + \triangle T$.

The procedure for advancing particles starting the time step in the matrix can be derived in the same manner and is entirely analogous:

Advancing a Particle in the Matrix (One-Dimensional Single-Interface Subproblem)

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.
2. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_m \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.
3. Model whether the particle has hit the interface location during the time step. If $Y_{max} \leq 0$, the particle must have hit the interface.Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{y_0 \cdot Y_{max}}{D_m \cdot \triangle t}}$.
4. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0|}{|Y_{max}|+|y_0|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = 0$, the location of the interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after the time step is $Y = Y_{max}$.
5. Advance the particle's time to $T = t_0 + \triangle T$.

4.2 Particles Starting on Interface

We again consider the one-dimensional single-interface system expressed by equations (3.3.1), (3.3.2) and (3.3.3). This time we consider a particle starting on the interface. Converting the particle into a mass density function gives the initial global mass density function:

$$\rho(y, t = 0) = \delta_0(y) \tag{4.2.1}$$

We solve this system of equations in Appendix B to get the following mass densities in the fracture and matrix:

$$\rho_f = \frac{\sqrt{D_f}}{b} \cdot \left[\frac{b}{k\sqrt{D_f}} e^{\frac{b}{k\sqrt{D_f}} y} \cdot I_{y \leq 0}\right] \star \left[2 \cdot \frac{1}{\sqrt{4\pi D_f t}} e^{-\frac{y'^2}{4D_f t}} \cdot I_{y \leq 0}\right] \qquad for\, y \leq 0 \tag{4.2.2a}$$

$$\rho_m = \frac{\varphi\sqrt{D_m}}{b} \cdot \left[\frac{b}{k\sqrt{D_m}} e^{-\frac{b}{k\sqrt{D_m}} y} \cdot I_{y \geq 0}\right] \star \left[2 \cdot \frac{1}{\sqrt{4\pi D_m t}} e^{-\frac{y^2}{4D_m t}} \cdot I_{y \geq 0}\right] \qquad for\, y \geq 0 \tag{4.2.2b}$$

where $b = \varphi\sqrt{D_m} + \sqrt{D_f}$. The symbol '$\star$' denotes the cross-correlation operation, an operation similar to convolution, which is defined as follows:

$$f(t) \star g(t) = \int_0^\infty f(\tau) \cdot g(t+\tau)\, d\tau \tag{4.2.3}$$

Recall the global mass density function, ρ, as defined by equation (3.3.3). We interpret this global mass density function at $t = \triangle t$ as the pdf of the random variable modeling the particle's new y-coordinate, Y:

$$f_Y(y) = \rho(y, \triangle t) = \begin{cases} \rho_f(y, \triangle t) & y < 0 \\ \rho_m(y, \triangle t) & y > 0 \\ \rho_i(y, \triangle t) & y = 0 \end{cases} \tag{4.2.4}$$

To aid in simulating the random variable Y, we will first simulate the following function of Y:

$$h(Y) = \begin{cases} -\frac{1}{\sqrt{D_f}} Y & Y < 0 \\ 0 & Y = 0 \\ \frac{1}{\sqrt{D_m}} Y & Y > 0 \end{cases} \tag{4.2.5}$$

We now find the pdf of $h(Y)$. We first find all the roots of h or the values of y such that $h(y) = \lambda$ for any given λ. There are no roots for $\lambda < 0$. For $\lambda = 0$, h has only one root root:$h^{-1}(0) = 0$. For $\lambda > 0$, there are two roots:

$$h_-^{-1}(y) = -\sqrt{D_f} y \qquad for\ y > 0 \tag{4.2.6a}$$

$$h_+^{-1}(y) = \sqrt{D_m} y \qquad for\ y > 0 \tag{4.2.6b}$$

We use the formula for finding the pdf of the function of a random variable when that function has a countable number of differentiable roots, which is:

$$f_{h(Y)}(y) = \sum_{h^{-1}(y) \in R} f_Y(h^{-1}(y)) \cdot \left| \frac{dh^{-1}(y)}{dy} \right| \tag{4.2.7}$$

where R is the set of roots$\{r | h(r) = y\}$. Substituting the pdf of Y from equation (4.2.4) and the roots of h from equation (4.2.6) into the above yields:

$$\begin{aligned} f_{h(Y)}(y) &= \begin{cases} f_Y(-\sqrt{D_f} y) \cdot \left|-\sqrt{D_f}\right| + f_Y(\sqrt{D_m} y) \cdot \left|\sqrt{D_m}\right| & y > 0 \\ f_Y(y) & y = 0 \end{cases} \\ &= \begin{cases} \left[\frac{b}{k} e^{-\frac{b}{k} y} \cdot I_{y \geq 0}\right] \star \left[2 \cdot \frac{1}{\sqrt{4\pi t}} e^{-\frac{y^2}{4t}} \cdot I_{y \geq 0}\right] & y > 0 \\ \rho_i(0, \triangle t_i) & y = 0 \end{cases} \end{aligned} \tag{4.2.8}$$

We note the first bracketed term is an exponential distribution. The second bracketed term is a half-normal distribution or, equivalently, the pdf of the absolute value of a mean-zero normal random variable. We rewrite $h(Y)$'s pdf in terms of these simpler pdfs:

$$f_{h(Y)}(y) = \begin{cases} f_E(y) \star f_{2\cdot\triangle t\left|Z\right|}(y) & y > 0 \\ \rho_i(0, \triangle t_i) & y = 0 \end{cases} \tag{4.2.9}$$

where E is an exponential random variable with rate parameter $\lambda = \frac{k_a}{b}$ and Z is a standard normal random variable. We will now use a very useful property of cross-correlation: the cross-correlation of the two random variable's pdfs is the pdf of the difference of the two random variables. Symbolically, this property is written as:

$$f_{X_1-X_2}(x) = f_{X_2}(x) \star f_{X_1}(x) \tag{4.2.10}$$

Using this property, we can rewrite the cross-correlation of two pdfs as a the pdf of a single random variable:

$$f_{h(Y)}(y) = \begin{cases} f_{2\cdot\triangle t\left|Z\right|-E}(y) & y > 0 \\ \rho_i(0, \triangle t_i) & y = 0 \end{cases} \tag{4.2.11}$$

We do yet know the value of $\rho_i(y = 0, y = \triangle t_{max})$ but the requirement that $h(Y)$'s pdf be normalized uniquely determines its value:

$$\begin{aligned} \int_{-\infty}^{\infty} f_{h(Y)}(y)\, dy &= 1 \\ \int_{0_-}^{0_+} \rho_i(0, \triangle t_i)\, dy + \int_{0}^{\infty} f_{2\cdot\triangle t\left|Z\right|-E}(y)\, dy &= 1 \\ \int_{0_-}^{0_+} \rho_i(0, \triangle t_i)\, dy + P\left(2 \cdot \triangle t\left|Z\right| - E > 0\right) &= 1 \\ \int_{0_-}^{0_+} \rho_i(0, \triangle t_i)\, dy &= 1 - P\left(2 \cdot \triangle t\left|Z\right| - E > 0\right) \\ \int_{0_-}^{0_+} \rho_i(0, \triangle t_i)\, dy &= P\left(2 \cdot \triangle t\left|Z\right| - E \leq 0\right) \\ \rho_i(0, \triangle t_i) &= P\left(2 \cdot \triangle t\left|Z\right| - E \leq 0\right) \cdot \delta_0(y) \end{aligned} \tag{4.2.12}$$

We can now substitute this evaluation of $\rho_i(0, \triangle t_i)$ into equation (4.2.11):

$$f_{h(Y)}(y) = \begin{cases} f_{2\cdot\triangle t\left|Z\right|-E}(y) & y > 0 \\ P\left(2\cdot\triangle t\left|Z\right| - E \leq 0\right)\cdot\delta_0(y) & y = 0 \end{cases} \tag{4.2.13}$$

At this point, we can recognize the right-hand side of the above to be the pdf of $min(2\cdot\triangle t\left|Z\right| - E, 0)$. Therefore, $h(Y)$ and $min(2\cdot\triangle t\left|Z\right| - E, 0)$ are distributed identically:

$$f_{h(Y)}(y) = f_{min(2\cdot\triangle t\left|Z\right|-E,0)}(y) \tag{4.2.14}$$

As a result of this, we can model $h(Y)$ as this random variable that shares its pdf:

$$h(Y) = min(2\cdot\triangle t\left|Z\right| - E, 0) \tag{4.2.15}$$

where E is an exponential random variable with rate parameter $\lambda = \frac{k_a}{b}$ and Z is a standard normal random variable. With $h(Y)$ so determined, it remains to model Y itself. We already have found the roots of the function h. If $h(Y)$ is zero, the only root is zero and so Y must be zero. If $h(Y)$ is greater than zero though, there are two possible values of Y: $-\sqrt{D_f}h(Y)$ and $\sqrt{D_m}h(Y)$. The probability of Y being equal to each root is proportional to the pdf of Y at that value. The probability that Y is equal to the positive root can then be evaluated as follows:

$$P(Y > 0|h(Y) = c > 0) = \frac{f_Y(\sqrt{D_m}c)}{f_Y(-\sqrt{D_f}c) + f_Y(\sqrt{D_m}c)} = \frac{\sqrt{D_m}}{b} \tag{4.2.16}$$

Note that the probability of Y being equal to the positive root does not depend on the value of $h(Y)$. With these results, we can efficiently simulate Y by first simulating $h(Y)$ and then probabilistically selecting the appropriate root. The algorithm to simulate the particle y-coordinate after of the time step, Y, is as follows:

Advancing a Particle on the Interface (One-Dimensional Single-Interface Subproblem)

1. Let the step length be $\triangle t = T_{final} - t_0$, the length of time until the end of the simulation.
2. Define parameter $b = \varphi\sqrt{D_m} + \sqrt{D_f}$.
3. Simulate $h(Y)$ as $min(2\cdot\triangle t\left|Z_0\right| - E, 0)$ where E is an exponential random variable with rate parameter $\lambda = \frac{k_a}{b}$ and Z_0 is a standard normal random variable.
4. If $h(Y)$ is greater than zero, Y is equal to the positive root, $\sqrt{D_m}h(y)$, with probability $\frac{\sqrt{D_m}}{b}$ and is equal to the negative root, $-\sqrt{D_f}h(Y)$, the rest of the time. If $h(Y)$ is zero, we can immediately conclude that Y is equal to the only root of $h(0)$: zero.
5. Advance the particle's time to $T = t_0 + \triangle T$.

4.3 Extension to Two-Dimensions

We now have developed methods for advancing particles in the one-dimensional single-interface subsystem, no matter what region the particle starts in. We next extend this particle tracking method to two-dimensions. The two-dimensional one-interface system to which we extend the particle tracking method is described by equations (3.2.1), (3.2.2) and (3.2.3). The one-dimensional particle method we just developed can already advance the y-coordinate of a particle in this system. However, it still remains to simulate the new x-coordinate. Our approach will be to infer the time spent in each region by the particle during the time step. Once this is known, simulating the final x-coordinate is simply a matter of applying the drift and diffusion from the time spent in each region using the Fokker-Planck equation:

$$X = x_0 + \sum_{r \in R} \left[u_r t_r + \sqrt{2 D_r t_r} \cdot Z_r \right] \tag{4.3.1}$$

where R is the set of all regions, t_r is the time spent by the particle in region r during the time step, u_r is the drift in the positive x-direction in region r, D_r is the effective diffusivity within region r, and Z_r is a standard normal random variable.

The drift and diffusivity of each region is already known. The drift in the fracture is D_f and the drift is U in the positive x-direction. The effective diffusivity in the matrix is D_m and there is no drift in the matrix. Finally, there is no diffusion or drift for particles on the interface; the particles are stuck while attached to it.

We turn to modeling the time spent in each region during a time step. For particles starting in the fracture or matrix, the time step ends when the particle first hits the interface. As hitting the boundary is the only way for the particle to leave its starting region, the time step ends before the particle spends any time in the other regions. As a result of this, particles starting in the fracture or matrix spend the entire time step in their starting region. The regional residence times are thus:

$$t_f = \triangle t,\ t_m = 0,\ t_i = 0\ if\ y_0 < 0 \tag{4.3.2a}$$

$$t_m = \triangle t,\ t_f = 0,\ t_i = 0\ if\ y_0 > 0 \tag{4.3.2b}$$

Modeling the time spent in each region for a particle starting on the interface is less straightforward: the particle can spend time in all three regions during the time step. Moreover, these residence times are conditioned on the new y-coordinate simulated by the one-dimensional particle tracking method. In 4.2, we modeled the whether the particle's y-coordinate after the time step, Y, is on the interface as follows:

$$Y = 0 \text{ if and only if } 2 \cdot \triangle t \big| Z \big| - E < 0 \tag{4.3.3}$$

where E is distributed as $Exp(\lambda = \frac{k}{b})$ and Z is a standard normal random variable. We will now make the simplifying assumption that Z and E remain constant with varying $\triangle t$. With E and Z held fixed, the right-hand side of the above increases monotonically with respect to $\triangle t$. We may solve for $\triangle t$ to find when right-hand expression first becomes positive and thus when the particle first leaves the interface:

$$t_{exit} = \frac{E}{2 \cdot \left|Z\right|} \tag{4.3.4}$$

Due to our simplifying assumption of holding the random variables constant, the particle stays on the interface before this exit time. The particle then spends the remainder of the time step, if any, in the region of Y. The time a particle starting on the interface spends in each region is therefore:

$$t_i = \begin{cases} \triangle t & Y = 0 \\ t_{exit} & otherwise \end{cases}, \tag{4.3.5a}$$

$$t_m = \begin{cases} \triangle t - t_{exit} & Y > 0 \\ 0 & otherwise \end{cases} \tag{4.3.5b}$$

$$t_f = \begin{cases} \triangle t - t_{exit} & Y < 0 \\ 0 & otherwise \end{cases} \tag{4.3.5c}$$

where $t_{exit} = \frac{E}{2 \cdot \left|Z\right|}$. With these regional residence times approximated, we can substitute these terms into equation (4.3.1), the Fokker-Planck equation, to simulate the final x-coordinate of the particle originating on the interface:

$$X = x_0 + U \cdot t_f + \sqrt{2D_f \cdot t_f} \cdot Z_1 + \sqrt{2D_m \cdot t_m} \cdot Z_2 \tag{4.3.6}$$

We must stress that the simplifying assumption we made for equation (4.3.3), that the random variables E and Z remain constant with varying $\triangle t$, will cause error. While shorter time steps minimize this error, the error caused can be severe for longer time steps. For this reason, we suggest that the time step for particles starting on the interface, $\triangle t$, be limited to at least one hundredth of the overall length of the simulation. If there is still systematic error in the simulation's results, this step length may have to be reduced further.

Appending this simulation of the x-coordinate to the one-dimensional particle tracking method we already developed for a single interface yields the single-interface particle tracking method:

Advancing a Particle in the Fracture (Single-Interface Subproblem)

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.

2. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_f \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.

3. Model whether the particle has hit the interface location during the time step. If $Y_{max} \geq 0$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{y_0 \cdot Y_{max}}{D_f \cdot \triangle t}}$.

4. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0|}{|Y_{max}|+|y_0|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = 0$, the location of the interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after the time step is $Y = Y_{max}$.

5. Advance the particle x-coordinate to $X = x_0 + U \cdot \triangle T + \sqrt{2D_f \cdot \triangle T} \cdot Z_1$ where Z_1 is a standard normal random variable.

6. Advance the particle's time to $T = t_0 + \triangle T$.

Advancing a Particle in the Matrix (Single-Interface Subproblem)

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.

2. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_m \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.

3. Model whether the particle has hit the interface location during the time step. If $Y_{max} \leq 0$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{y_0 \cdot Y_{max}}{D_m \cdot \triangle t}}$.

4. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0|}{|Y_{max}|+|y_0|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = 0$, the location of the interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after time step is $Y = Y_{max}$.

5. Advance the particle x-coordinate to $X = x_0 + \sqrt{2D_m \cdot \triangle T} \cdot Z_1$ where Z_1 is a standard normal random variable.

6. Advance the particle's time to $T = t_0 + \triangle T$.

Advancing a Particle on the Interface (Single-Interface Subproblem)

1. Let $\triangle t$, the time step length, be equal to the smaller of the following two values:$\frac{1}{100}T_{final}$, one hundredth of the simulation's total length, or $T_{final} - t_0$, the length of time until the end of the simulation.

2. Define parameter $b = \varphi\sqrt{D_m} + \sqrt{D_f}$.

3. Simulate $h(Y)$ as $min(2 \cdot \triangle t\left|Z_0\right| - E, 0)$ where E is an exponential random variable with rate parameter $\lambda = \frac{k_a}{b}$ and Z_0 is a standard normal random variable.

4. If $h(Y)$ is greater than zero, Y is equal to the positive root, $\sqrt{D_m}h(y)$, with probability $\frac{\sqrt{D_m}}{b}$ and is equal to the negative root, $-\sqrt{D_f}h(Y)$, the rest of the time. If $h(Y)$ is zero, we can immediately conclude that Y is equal to the only root of $h(0)$: zero.

5. Approximate the particle's regional residence times as $t_i = \begin{cases} \triangle t & Y = 0 \\ t_{exit} & otherwise \end{cases}$, $t_m = \begin{cases} \triangle t - t_{exit} & Y > 0 \\ 0 & otherwise \end{cases}$ and $t_f = \begin{cases} \triangle t - t_{exit} & Y < 0 \\ 0 & otherwise \end{cases}$ where $t_{exit} = \frac{E}{2 \cdot \left|Z_0\right|}$.

6. Advance the particle's x-coordinate to $X = x_0 + U \cdot t_f + \sqrt{2D_f \cdot t_f} \cdot Z_1 + \sqrt{2D_m \cdot t_m} \cdot Z_2$where Z_1 and Z_2 are standard normal random variables.

7. Advance the particle's time to $T = t_0 + \triangle T$.

5 Full Particle Tracking Methods

In this section, we will extend the particle tracking method we developed in the previous chapter to the full two-interface system presented by equations (3.1.7), (3.1.4) and (3.1.5). We do this in two separate ways, each yielding a different particle tracking method.

5.1 Hi-res Method

We here extend the single-interface particle tracking method we developed to the full system by constraining the time step so that the particle interacts with no more than one interface each time step. Because the resulting method captures all of the model's features at a small scale, we call it the high-resolution or hi-res particle tracking method.

We first recall the reasoning that originally led us to consider the single-interface subsystem. As the two interfaces are parallel to each other, a particle will always be at least distance L from one of the interfaces. By choosing a sufficiently small maximum time step length, the probability of the particle interacting with the far interface during the time step can be made arbitrary small, no matter where the particle starts. We can then safely ignore the far interface with minimal effect. Keeping the time step small is not necessary for particles in the matrix; they could not reach the far interface in a single time step anyway due to the time step automatically ending when they first exit the matrix.

How small must $\triangle t_{max}$ for particles starting in the fracture or on the interface to ensure they do not not interact with the far interface during the time step? The closest a particle can be to the far interface is distance L if it starts in the center of the fracture. Consequently, the probability of a particle traveling this distance via fracture diffusion after the time step must be vanishingly small. For a particle on a line that experiences Brownian motion with diffusivity D_f, the time of first hitting a point that distance L from the starting initial point is distributed according to a Levy distribution with parameters $\mu = 0$ and $c = \frac{y_0^2}{2D_m}$ [Jacobs, 2010], a result we confirmed in section 2.3. The cdf of this Levy distribution is:

$$F_\tau(t) = \sqrt{\frac{\left(\frac{L^2}{2D_f}\right)}{2\pi}}\,\frac{1}{t^{3/2}}\,e^{-\frac{\left(\frac{L^2}{2D_f}\right)}{2t}} \tag{5.1.1}$$

This cdf gives the probability that the first hitting time will be less then any given t. It follows that $F_\tau(\triangle t_{max})$ bounds the probability that the particle hits the interface location within the time step:

$$P_{hitting} \leq Erfc\left(\frac{L}{\sqrt{4D_f \cdot \triangle t_{max}}}\right) \tag{5.1.2}$$

We now solve for $\triangle t_f$ to find the time step length necessary to obtain any given value of $P_{hitting}$:

$$\triangle t_{max} = \frac{L^2}{4 \cdot D_f} \cdot \left(Erfc^{-1}(P_{hitting}) \right)^{-2} \tag{5.1.3}$$

Rather than attempt to theoretically calculate an acceptable hitting probability for a simulation, we instead note that the maximum time step length is proportional to $\frac{L^2}{D_f}$, the characteristic diffusion time scale across the fracture. After comparing numerical results produced with different maximum time steps, we empirically found that the results tended to be stable for time steps of length no greater than $\frac{1}{10}\frac{L^2}{D}$.

With the time steps so constrained, we can assume the particle will not interact with the interface further from its starting point during a time step and so ignore that interface during each time step. This reduces the problem to the single-interface subproblem during each time step, albeit shifted and possibly flipped. Because of this, we can advance the particle using the single-fracture particle tracking method with an appropriately reduced maximum time step for particles starting in the fracture or on an interface. This effectively extends the single-interface method to the full problem. Finally, in the rare event that a particle *does* end a time step in matrix past the ignored interface, we will place the particle on that ignored interface instead. This is a rough measure to prevent the particle from wrongly getting stuck deep in the matrix due to our approximation.

Implementing this approach yields the hi-res particle tracking method for the original system:

Hi-res Method: Advancing a Particle in the Fracture

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.
2. Let y_i be the y-coordinate of the near interface, $sgn(y_0) \cdot L$. If $y_0$is exactly zero, let y_i be L by default.
3. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_f \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.
4. Model whether the particle has hit the interface location during the time step. If $\frac{Y_{max}}{y_i} \geq 1$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{y_0 \cdot Y_{max}}{D_f \cdot \triangle t}}$.
5. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0 - y_i|}{|Y_{max} - y_i| + |y_0 - y_i|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = y_i$, the location of the near interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after the time step is $Y = Y_{max}$.
6. If $\frac{Y}{y_i} \geq -1$, then the particle has violated our assumption of not interacting with the far interface. We mitigate the error due to this violation by setting $Y = -y_i$.
7. Advance the particle x-coordinate to $X = x_0 + U \cdot \triangle T + \sqrt{2D_f \cdot \triangle T} \cdot Z_1$ where Z_1 is a standard normal random variable.
8. Advance the particle's time to $T = t_0 + \triangle T$.

Hi-res Method: Advancing a Particle in the Matrix

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.
2. Let y_i be the y-coordinate of the near interface, $sgn(y_0) \cdot L$.
3. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_m \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.
4. Model whether the particle has hit the interface location during the time step. If $\frac{Y_{max}}{y_i} \leq 1$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{(y_0 - y_i)\cdot(Y_{max} - y_i)}{D_m \cdot \triangle t}}$.
5. If the particle is found to have hit the interface during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0 - y_i|}{|Y_{max} - y_i| + |y_0 - y_i|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = y_i$, the location of the near interface. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after time step is $Y = Y_{max}$.
6. Advance the particle x-coordinate to $X = x_0 + \sqrt{2D_m \cdot \triangle T} \cdot Z_1$ where Z_1 is a standard normal random variable.
7. Advance the particle's time to $T = t_0 + \triangle T$.

Hi-res Method: Advancing a Particle on the Interface

1. Let $\triangle t$, the time step length, be equal to the smaller of the following two values:$\frac{1}{100}T_{final}$, one hundredth of the simulation's total length, or $T_{final} - t_0$, the length of time until the end of the simulation.
2. Define parameter $b = \varphi\sqrt{D_m} + \sqrt{D_f}$.
3. Simulate $\hat{h}(Y)$ as $min(2 \cdot \triangle t\left|Z_0\right| - E, 0)$ where E is an exponential random variable with rate parameter$\lambda = \frac{k_a}{b}$ and Z_0 is a standard normal random variable.
4. If $\hat{h}(Y)$ is greater than zero, Y is equal to root $y_0 + sgn(y_0) \cdot \sqrt{D_m}\hat{h}(y)$ with probability $\frac{\varphi\sqrt{D_m}}{b}$ and is equal to root $y_0 - sgn(y_0) \cdot \sqrt{D_f}\hat{h}(Y)$ the rest of the time. If $\hat{h}(Y)$ is zero, we can immediately conclude that Y is equal to the only root of $\hat{h}(0)$: y_0.
5. If $\frac{Y}{y_i} \geq -1$, then the particle has violated our assumption of not interacting with the far interface. We mitigate the error due to this violation by setting $Y = -y_i$.
6. Approximate the particle's regional residence times as $t_i = \begin{cases} \triangle t & Y = 0 \\ t_{exit} & otherwise \end{cases}$, $t_m = \begin{cases} \triangle t - t_{exit} & Y > 0 \\ 0 & otherwise \end{cases}$ and $t_f = \begin{cases} \triangle t - t_{exit} & Y < 0 \\ 0 & otherwise \end{cases}$ where $t_{exit} = \frac{E}{2 \cdot \left|Z_0\right|}$.
7. Advance the particle's x-coordinate to $X = x_0 + U \cdot t_f + \sqrt{2D_f \cdot t_f} \cdot Z_1 + \sqrt{2D_m \cdot t_m} \cdot Z_2$where Z_1 and Z_2 are standard normal random variables.
8. Advance the particle's time to $T = t_0 + \triangle T$.

5.2 Upscaled Method

In this section, we will extend the single-interface particle tracking method we developed to the full system by treating the combined fracture-interface system like a single interface. As we are capturing the cumulative effect of the system's fracture-interface dynamics with a courser effective model, we term the resulting method the upscaled particle tracking method.

To motivate this method, we first note that the effective diffusivity within the fracture, D_f, is typically at least an order of magnitude greater than the effective diffusivity within the matrix, D_m. Because of this, the fracture solute distribution tends to quickly become uniform across the fracture. The mass density at each edge of the fracture is then proportional to the mass on the the interface which is in turn proportional to the mass density at the adjacent matrix edge. Thus, the entire fracture-interface system tends to be in equilibrium with the edges of the matrix, much like an interface.

To make this analogy more explicit, we take the limit of the mass transport equations, equations (3.1.7), as the y-component of the fracture's diffusion goes to infinity. We decouple the fracture's widthwise diffusivity from the lengthwise diffusivity and denote the new widthwise diffusivity parameter by D_w:

$$\frac{\partial \rho_f}{\partial t} = - u\frac{\partial \rho_f}{\partial x} + D_f\frac{\partial^2 \rho_f}{\partial x^2} + D_w\frac{\partial^2 \rho_f}{\partial y^2} \qquad for\ |y| < L \tag{5.2.1a}$$

$$\frac{\partial \rho_m}{\partial t} = D_m\left(\frac{\partial^2 \rho_m}{\partial x^2} + \frac{\partial^2 \rho_m}{\partial y^2}\right) \qquad for\ |y| > L \tag{5.2.1b}$$

$$\rho_f = \frac{1}{\varphi}\rho_m \qquad for\ y = \pm L \tag{5.2.1c}$$

$$\pm k_a\frac{\partial \rho_f}{\partial t} = D_m\frac{\partial \rho_m}{\partial y} - D_w\frac{\partial \rho_f}{\partial y} \qquad for\ y = \pm L \tag{5.2.1d}$$

We take the limit of these equations as $D_w \to \infty$ for $t > 0$ and, after much manipulation shown in Appendix C, we get the upscaled mass transport equations:

$$\frac{\partial \rho_{m1}}{\partial t} = D_m\left(\frac{\partial^2 \rho_{m1}}{\partial x^2} + \frac{\partial^2 \rho_{m1}}{\partial y^2}\right) \qquad for\ y < 0 \tag{5.2.2a}$$

$$\frac{\partial \rho_{m2}}{\partial t} = D_m\left(\frac{\partial^2 \rho_{m2}}{\partial x^2} + \frac{\partial^2 \rho_{m2}}{\partial y^2}\right) \qquad for\ y > 0 \tag{5.2.2b}$$

$$\rho_{m1} = \rho_{m2} \qquad for\ y = 0 \tag{5.2.2c}$$

$$\frac{2\left(k_a + L\right)}{\varphi}\frac{\partial \rho_{m1}}{\partial t} = D_m\frac{\partial \rho_{m2}}{\partial y} - D_m\frac{\partial \rho_{m1}}{\partial y} + \frac{2L\cdot u}{\varphi}\frac{\partial \rho_m}{\partial x} - \frac{2L\cdot D_f}{\varphi}\frac{\partial^2 \rho_m}{\partial x^2} \qquad for\ y = 0 \tag{5.2.2d}$$

where the new upscaled mass density functions are defined as:

$$\rho_{m1}(x, y, t) = \rho_m(x, y - L, t) \qquad for\ y \leq 0 \tag{5.2.3a}$$

$$\rho_{m2}(x, y, t) = \rho_m(x, y + L, t) \qquad for\ y \geq 0 \tag{5.2.3b}$$

$$\rho_{f/i}(x, y, t) = \delta_0(y)\cdot\int_{-\infty}^{\infty}\rho_f(x, \hat{y}, t) + \rho_i(x, \hat{y}, t)\, d\hat{y} \qquad for\ y = 0 \tag{5.2.3c}$$

We also define a new global upscaled mass density function:

$$\rho_{upscale}(x,y,t) = \begin{cases} \rho_{m1}(x,y,t) & y<0 \\ \rho_{m2}(x,y,t) & y>0 \\ \rho_{i/f}(x,y,t) & y=0 \end{cases} \tag{5.2.4}$$

This system of PDEs is of nearly identical form to the mass transport equations for the one-dimensional single-interface system, equations. (3.2.1). The only difference is that, in this new system, particle on the effective interface at $y = 0$ experience advection and diffusion along the x-direction. To make this similarity even more apparent, we will remove the x-dimension from equations (5.2.2) by integrating the system of differential equations from $x = -\infty$ to $x = \infty$ to get:

$$\frac{\partial \rho_{m1}}{\partial t} = D_m \left(\frac{\partial^2 \rho_{m1}}{\partial x^2} + \frac{\partial^2 \rho_{m1}}{\partial y^2} \right) \qquad for\, y<0 \tag{5.2.5a}$$

$$\frac{\partial \rho_{m2}}{\partial t} = D_m \left(\frac{\partial^2 \rho_{m2}}{\partial x^2} + \frac{\partial^2 \rho_{m2}}{\partial y^2} \right) \qquad for\, y>0 \tag{5.2.5b}$$

$$\rho_{m1} = \rho_{m2} \qquad for\, y=0 \tag{5.2.5c}$$

$$\frac{2\left(k_a+L\right)}{\varphi}\frac{\partial \rho_{m1}}{\partial t} = D_m \frac{\partial \rho_{m2}}{\partial y} - D_m \frac{\partial \rho_{m1}}{\partial y} \qquad for\, y=0 \tag{5.2.5d}$$

These equations are exactly the same as equations 3.3.1, the one-dimensional single-interface mass transport equations, save with different parameters. Thus, if we first convert the particle's coordinates into upscaled coordinates, we can advance those upscaled coordinates forward in time using the single-fracture particle tracking method we already developed. The particle's upscaled coordinates can then be translated back into the particle's actual coordinates. We will now work out the details of each of these steps.

The single-interface method can be applied directly to this upscaled system representation. The new upscaled parameters can read off of the upscaled system of PDEs. The absorption constant of the fracture-interface system is $k_{eff} = \frac{2(k_a+L)}{\varphi}$. The matrix regions to each side of the fracture-interface system have diffusivity D_f and no drift. The parameter φ is trivially equal to one. Unlike the single-interface model, particles on the effective interface at the origin experience drift and diffusion along the x-dimension. To account for this in advancing the x-coordinate, we simply use new effective drift and diffusivity parameters in equation 4.3.1, the Fokker-Planck equation. The drift and diffusivity in the x-direction of the interface-fracture can be read off the upscaled system of PDEs as $\frac{L}{k_a+L} \cdot u$ and $\frac{L}{k_a+L} \cdot D_f$ respectively. However, these values don't take account subtle effects like the extra effective diffusion due by variance in the time the particles spend experiencing advection in the fracture. To capture these effects, we instead use values generated by the "asymptotic spectral comparison method" as applied to the parallel-fracture models by Bloechle [2001]:

$$u_{eff} = \frac{L}{k_a + L} \cdot u \tag{5.2.6a}$$

$$D_{eff} = \left[\frac{L}{k_a + L} + \frac{k_a^2 \cdot L}{6 \cdot (k_a + L)^3} \cdot \left(\frac{L \cdot u}{D_f} \right)^2 \right] \cdot D_f \tag{5.2.6b}$$

To convert the particle's coordinates into upscaled coordinates, we consider a mass density function of a two-dimensional Dirac delta function at the particle's coordinates and find the corresponding upscaled mass density function. The initial mass density function representing a particle at an arbitrary location is:

$$\rho(x, y, t_0) = \delta_{x_0}(x) \cdot \delta_{y_0}(y) \tag{5.2.7}$$

We break this global mass density function apart into the regional mass density functions using equation (3.1.5):

$$\rho_f = \delta_{x_0}(x) \cdot \delta_{y_0}(y) \qquad for\ |y| < L \tag{5.2.8a}$$

$$\rho_m = \delta_{x_0}(x) \cdot \delta_{y_0}(y) \qquad for\ |y| > L \tag{5.2.8b}$$

$$\rho_i = \delta_{x_0}(x) \cdot \delta_{y_0}(y) \qquad for\ |y| = L \tag{5.2.8c}$$

We now convert these into upscaled regional mass density functions using equations (5.2.3):

$$\rho_{m1} = \delta_{x_0}(x) \cdot \delta_{y_0+L}(y) \qquad for\ y < 0 \tag{5.2.9a}$$

$$\rho_{m2} = \delta_{x_0}(x) \cdot \delta_{y_0-L}(y) \qquad for\ y > 0 \tag{5.2.9b}$$

$$\rho_{f/i} = \delta_{x_0}(x) \cdot \delta_0(y) \cdot I_{-L \le y_0 \le L} \qquad for\ y = 0 \tag{5.2.9c}$$

We substitute these expressions into equation (5.2.4) to get:

$$\rho_{upscaled} = \begin{cases} \delta_{x_0}(x) \cdot \delta_{y_0+L}(y) & for\ y < 0 \\ \delta_{x_0}(x) \cdot \delta_{y_0-L}(y) & for\ y > 0 \\ \delta_{x_0}(x) \cdot \delta_0(y) \cdot I_{-L \le y_0 \le L} & for\ y = 0 \end{cases} \tag{5.2.10}$$

This upscaled mass density function actually works out to be another two dimensional Dirac delta function, regardless of the values of x_0 and y_0. The location of the Dirac delta function can be interpreted as the particle's upscaled coordinates. The particles upscaled coordinates are therefore:

$$x_{0,upscaled} = x_0 \tag{5.2.11a}$$

$$y_{0,upscaled} = \begin{cases} y_0 + L & if\, y_0 < -L \\ y_0 - L & if\, y_0 > L \\ 0 & otherwise \end{cases} \tag{5.2.11b}$$

After advancing these upscaled coordinates forward in time with the single-interface particle tracking method,we must convert the upscaled coordinates back into the particle's actual coordinates. To do this, we consider an upscaled mass density function of a two-dimensional Dirac delta function at the upscaled coordinates and find the corresponding mass density function. The upscaled mass density function is:

$$\rho_{upscaled}(x, y, T) = \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}}(y) \tag{5.2.12}$$

We now break this down into the upscaled regional mass density functions using equations (5.2.4):

$$\rho_{m1} = \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}}(y) \qquad for\, y < 0 \tag{5.2.13a}$$

$$\rho_{m2} = \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}}(y) \qquad for\, y > 0 \tag{5.2.13b}$$

$$\rho_{f/i} = \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}}(y) \qquad for\, y = 0 \tag{5.2.13c}$$

We now invert the regional upscaled mass density functions, given in equations (5.2.3), to get the true mass density function. It is straightforward to invert the definitions of $\rho_{m1}$and ρ_{m2} to give the true matrix mass density function:

$$\rho_m = \begin{cases} \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}-L}(y) & y < 0 \\ \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}+L}(y) & y > 0 \end{cases} \tag{5.2.14}$$

Inverting the definition of $\rho_{f/i}$ to get the true fracture and interface mass density functions is less straightforward. Plugging in the value of $\rho_{f/i}$ from equation (5.2.13c) into equation (5.2.3c) gives that:

$$\delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}}(0) = \delta_0(0) \cdot \int_{-\infty}^{\infty} \rho_f(x, \hat{y}, t) + \rho_i(x, \hat{y}, t)\, d\hat{y} \tag{5.2.15}$$

Dividing through both sides by $\delta_0(0)$ yields that:

$$\delta_{X_{upscaled}}(x) \cdot I_{Y_{upscaled}=0} = \int_{-\infty}^{\infty} \rho_f(x,\hat{y},t) + \rho_i(x,\hat{y},t)\; d\hat{y} \tag{5.2.16}$$

The definition of the interface mass density, equation (3.1.4), gives that the interface mass density is $\rho_i = k_a \cdot \rho_f(x,y,t) \cdot [\delta_{-L}(y) + \delta_L(y)]$. We substitute this into the above:

$$\delta_{X_{upscaled}}(x) \cdot I_{Y_{upscaled}=0} = \int_{-\infty}^{\infty} \rho_f(x,\hat{y},t) + k_a \cdot C_f(x,y,t) \cdot [\delta_{-L}(y) + \delta_L(y)]\; d\hat{y} \tag{5.2.17}$$

Equation (C.3) of appendix A shows that $\rho_f(x,y,t)$ is invariant with respect to y within the fracture with the upscaled assumptions. We use this equation to change the integral into a product:

$$\begin{aligned}\delta_{X_{upscaled}}(x) \cdot I_{Y_{upscaled}=0} =& \int_{-\infty}^{\infty} I_{-L\le y\le L} \cdot \rho_f(x,y_f,t) + k_a \cdot \rho_f(x,y_f,t) \cdot [\delta_{-L}(y) + \delta_L(y)]\; d\hat{y} \\ =& 2 \cdot L \cdot \rho_f(x,y_f,t) + 2 \cdot k_a \cdot \rho_f(x,y_f,t) \\ =& 2 \cdot (L + k_a) \cdot \rho_f(x,y_f,t) \end{aligned} \tag{5.2.18}$$

We now solve for $\rho_f(x,y_f,t)$ to get the value of $\rho_f(x,y_f,t)$ for any value y in $[-L,L]$:

$$\rho_f(x,y,T) = \frac{1}{2\cdot(L+k_a)} \cdot \delta_{X_{upscaled}}(x) \cdot I_{Y_{upscaled}=0}\; for\; -L<y<L \tag{5.2.19}$$

We relate this fracture mass density to the interface mass density using equation (3.1.4) :

$$\begin{aligned}\rho_i(x,y,T) &= k_a \cdot \rho_f(x,y,t) \cdot [\delta_{-L}(y) + \delta_L(y)] \\ &= \frac{k_a}{2\cdot(L+k_a)}\,[\delta_{-L}(y) + \delta_L(y)] \cdot \delta_{X_{upscaled}}(x) \cdot I_{Y_{upscaled}=0}\end{aligned} \tag{5.2.20}$$

We now have found all the regional mass densities. We combine them together with equation (3.1.5) and simplify to get the global mass density:

$$\rho = \begin{cases} \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}-L}(y) & Y_{upscaled} < 0 \\ \delta_{X_{upscaled}}(x) \cdot \delta_{Y_{upscaled}+L}(y) & Y_{upscaled} > 0 \\ \delta_{X_{upscaled}}(x) \cdot \left[\frac{k_a\cdot\delta_{-L}(y) + I_{-L\le y\le L}(y) + k_a\cdot\delta_L(y)}{2\cdot(L+k_a)}\right] & Y_{upscaled} = 0 \end{cases} \tag{5.2.21}$$

We can now see that the global mass density function is equal to a two-dimensional Dirac delta function except when $Y_{upscaled} = 0$. In that case, we shall interpret the mass density function as a probability density function of the particle's location and simulate the particle's coordinates accordingly. The y-coordinate when $Y_{upscaled} = 0$ can be simulated accordingly as follows:

$$M = \begin{cases} (2 \cdot U_1 - 1) \cdot L & if\, U_0 < \frac{L}{L+k_a} \\ sgn\,(2 \cdot U_1 - 1) \cdot L & if\, U_0 > \frac{L}{L+k_a} \end{cases} \tag{5.2.22}$$

where U_0 and U_1 are standard uniform random variables. This gives the following method of converting the upscaled coordinates back into the particle's actual coordinates:

$$X = X_{upscaled} \tag{5.2.23a}$$

$$Y = \begin{cases} Y_{upscaled} - L & Y_{upscaled} < 0 \\ Y_{upscaled} + L & Y_{upscaled} > 0 \\ M & Y_{upscaled} = 0 \end{cases} \tag{5.2.23b}$$

where $M = \begin{cases} (2 \cdot U_1 - 1) \cdot L & if\, U_0 < \frac{L}{L+k_a} \\ sgn\,(2 \cdot U_1 - 1) \cdot L & if\, U_0 > \frac{L}{L+k_a} \end{cases}$ and U_0, U_1 are standard uniform random variables.

Combining these steps of converting the particle to upscaled coordinates, advancing it forward in time and then converting the coordinates back the the particles true new position yields the upscaled particle tracking method:

Upscaled Method: Advancing a Particle in the Matrix

1. Let the maximum time step length be $\triangle t_{max} = T_{final} - t_0$, the length of time until the end of the simulation.
2. Let y_i be the y-coordinate of the near interface, $sgn(y_0) \cdot L$.
3. Simulate the particle's y-position assuming there was no interface as $Y_{max} = y_0 + \sqrt{2D_m \cdot \triangle t_{max}} \cdot Z_0$ where Z_0 is a standard normal random variable.
4. Model whether the particle has hit the interface location during the time step. If $\frac{Y_{max}}{y_i} \leq 1$, the particle must have hit the interface. Otherwise, the probability the particle has hit the interface is $P_{hitting} = e^{-\frac{(y_0 - y_i)\cdot(Y_{max} - y_i)}{D_m \cdot \triangle t}}$.
5. If the particle is found to have entered the fracture-interface system during the time step, then the time step length is $\triangle T = \triangle t_{max} \cdot \left(\frac{|y_0 - y_i|}{|Y_{max} - y_i| + |y_0 - y_i|}\right)^2$, the first hitting time, and the particle's y-coordinate after the time step is $Y = M$ where $M = \begin{cases} (2 \cdot U_1 - 1) \cdot L & if\, U_0 < \frac{L}{L + k_a} \\ sgn\,(2 \cdot U_1 - 1) \cdot L & if\, U_0 > \frac{L}{L + k_a} \end{cases}$ and U_0, U_1 are standard uniform random variables. Otherwise the step length is $\triangle T = \triangle t_{max}$, the maximum time step length, and the particle's y-coordinate after time step is $Y = Y_{max}$.
6. Advance the particle x-coordinate to $X = x_0 + \sqrt{2D_m \cdot \triangle T} \cdot Z_1$ where Z_1 is a standard normal random variable.
7. Advance the particle's time to $T = t_0 + \triangle T$.

Upscaled Method: Advancing a Particle in the Fracture or on the Interface

1. Let $\triangle t$, the time step length, be equal to the smaller of the following two values: $\frac{1}{100}T_{final}$, one hundredth of the simulation's total length, or $T_{final} - t_0$, the length of time until the end of the simulation.

2. Define effective upscaled parameters $b_{eff} = 2\sqrt{D_m}$, $k_{eff} = \frac{2(k_a+L)}{\varphi}$, $u_{eff} = \frac{L}{k_a+L} \cdot u$ and $D_{eff} = \left[\frac{L}{k_a+L} + \frac{k_a^2 \cdot L}{6\cdot(k_a+L)^3} \cdot \left(\frac{L\cdot u}{D_f}\right)^2\right] \cdot D_f$.

3. Simulate $h(Y_{upscaled})$ as $min(2 \cdot \triangle t\left|Z_0\right| - E, 0)$ where E is an exponential random variable with rate parameter $\lambda = \frac{k_{eff}}{b_{eff}}$ and Z_0 is a standard normal random variable.

4. If $h(Y_{upscaled})$ is greater than zero, Y is equal to the positive root, $L + \sqrt{D_m}h(y)$, with probability $\frac{1}{2}$ and is equal to the negative root, $-L - \sqrt{D_m}h(Y)$, the rest of the time. If $h(Y_{upscaled})$ is zero, we conclude that $Y_{upscaled} = 0$ and so $Y = M$ where $M = \begin{cases} (2 \cdot U_1 - 1) \cdot L & if\, U_0 < \frac{L}{L+k_a} \\ sgn\,(2 \cdot U_1 - 1) \cdot L & if\, U_0 > \frac{L}{L+k_a} \end{cases}$ and U_0, U_1 are standard uniform random variables.

5. Approximate the particle's fracture-interface residence time as $t_{f/i} = \begin{cases} \triangle t & |Y| \leq L \\ t_{exit} & otherwise \end{cases}$ and $t_{m1/m2} = \begin{cases} \triangle t - t_{exit} & |Y| > L \\ 0 & otherwise \end{cases}$ where $t_{exit} = \frac{E}{2\cdot\left|Z_0\right|}$.

6. Advance the particle's x-coordinate to $X = x_0 + U_{eff} \cdot t_{f/i} + \sqrt{2D_{eff} \cdot t_{f/i}} \cdot Z_1 + \sqrt{2D_m \cdot t_{m1/m2}} \cdot Z_2$ where Z_1 and Z_2 are standard normal random variables.

7. Advance the particle's time to $T = t_0 + \triangle T$.

6 Verification of Proposed Methods

To verify that these new methods are accurate, we will compare their results with accepted numerical results from the literature for different end times and parameter values. We also include the results of particle tracking method developed by LaBolle, Fogg, and Tompson [1996] as another point of comparison, hereafter referred to as LaBolle's method. As LaBolle's method does not allow for interface absorption, we only include its results for simulations where there is no interface absorption.

For all simulations, the initial conditions are taken to be unit mass of solute evenly distributed across the fracture's width at $y = 0$. We use the four parameter sets chosen to be consistent with the value ranges presented by Reimus et al. [2003]. These parameter sets are shown in the Table 2. For parameter set one and two, there is no interface absorption and so $k_a = 0$. For parameters sets one and three, there is no matrix rock absorption and so $\rho_b \cdot k_d = 0$. The parameter sets are otherwise identical.

6.1 Particle Tracking Method Comparisons

For each parameter set, we first simulate three end times given in terms of the characteristic diffusion time scale across the fracture, $\frac{a^2}{D_f}$: $T = 10^2 \frac{a^2}{Df}$, $T = 10^3 \frac{a^2}{Df}$ and $T = 10^4 \frac{a^2}{Df}$. The longest of these end times, $T = 10^4 \frac{a^2}{Df}$, is roughly 29 days. We use 10,000 particles for each of these simulations. For the hi-res method, we let the maximum time step be $\frac{1}{10} \cdot \frac{a^2}{D_f}$. For the upscaled method, we let the maximum time step be $\frac{1}{100} T_{end}$, a hundredth of the simulated time length. For LaBolle's method, we let the time steps be of length $\frac{1}{10} \cdot \frac{a^2}{D_f}$.

We first look at the computational cost of these simulations. Table 3 shows an estimate of the floating point operations or FLOPs performed during each simulation as given by the Lightspeed Matlab toolbox. The clearest trend is that the upscaled method consistently requires the fewest FLOPs by far. This is due to the low number of time steps taken by the upscaled method: in these simulations it takes about one-hundred or so steps per particle, no matter the end time. In contrast, the hi-res method and LaBolle's method are each taking at least 1,000 time steps per particle for the $T = 10^2 \frac{a^2}{D_f}$ simulations and 100,000 time steps per particle for the $T = 10^4 \frac{a^2}{D_f}$ simulations. A second trend is that LaBolle's method generally, though not always, uses less FLOPs than the hi-res method.

We now examine the accuracy of these simulation results. We first look at the concentration profiles along the y-axis, the direction perpendicular to the fracture's length. We take the true profile to be the series solution presented by Bloechle [2001]. We compare the results of each particle tracking method simulation at 24 sample points with Bloechle's concentration profile solution in Figure 3, Figure 4, Figure 5 and Figure 6. In Table 4, we show a qualitative measure of the

Set Name	a	D_f	D_e	ϕ	u	k_a	$\rho_b \cdot k_d$
Parameter Set 1	$5 * 10^{-4}\, m$	$10^{-9}\, \mathrm{m^2/s}$	$10^{-10}\, \mathrm{m^2/s}$	0.1	$1\, \mathrm{m/day}$	$0\, m$	0
Parameter Set 2	$5 * 10^{-4}\, m$	$10^{-9}\, \mathrm{m^2/s}$	$10^{-10}\, \mathrm{m^2/s}$	0.1	$1\, \mathrm{m/day}$	$0\, m$	10
Parameter Set 3	$5 * 10^{-4}\, m$	$10^{-9}\, \mathrm{m^2/s}$	$10^{-10}\, \mathrm{m^2/s}$	0.1	$1\, \mathrm{m/day}$	$5 * 10^{-4}\, m$	0
Parameter Set 4	$5 * 10^{-4}\, m$	$10^{-9}\, \mathrm{m^2/s}$	$10^{-10}\, \mathrm{m^2/s}$	0.1	$1\, \mathrm{m/day}$	$5 * 10^{-4}\, m$	10

Table 2: Parameter value sets used for the test simulations.

Simulation		Hi-res Method	Upscaled Method	LaBolle's Method
Parameter Set 1	$T = 10^2 \frac{a^2}{D_f}$	$1.34 \cdot 10^3$	$1.13 \cdot 10^2$	$4.72 \cdot 10^2$
	$T = 10^3 \frac{a^2}{D_f}$	$1.31 \cdot 10^4$	$1.10 \cdot 10^2$	$4.75 \cdot 10^3$
	$T = 10^4 \frac{a^2}{D_f}$	$1.08 \cdot 10^5$	$9.88 \cdot 10^1$	$4.70 \cdot 10^4$
Parameter Set 2	$T = 10^2 \frac{a^2}{D_f}$	$8.03 \cdot 10^2$	$7.95 \cdot 10^1$	$4.57 \cdot 10^2$
	$T = 10^3 \frac{a^2}{D_f}$	$4.31 \cdot 10^3$	$5.46 \cdot 10^1$	$3.70 \cdot 10^3$
	$T = 10^4 \frac{a^2}{D_f}$	$1.55 \cdot 10^4$	$3.76 \cdot 10^1$	$4.39 \cdot 10^4$
Parameter Set 3	$T = 10^2 \frac{a^2}{D_f}$	$1.11 \cdot 10^3$	$1.13 \cdot 10^2$	-
	$T = 10^3 \frac{a^2}{D_f}$	$1.08 \cdot 10^4$	$1.12 \cdot 10^2$	-
	$T = 10^4 \frac{a^2}{D_f}$	$1.02 \cdot 10^5$	$1.05 \cdot 10^2$	-
Parameter Set 4	$T = 10^2 \frac{a^2}{D_f}$	$8.43 \cdot 10^2$	$9.31 \cdot 10^1$	-
	$T = 10^3 \frac{a^2}{D_f}$	$5.15 \cdot 10^3$	$7.52 \cdot 10^1$	-
	$T = 10^4 \frac{a^2}{D_f}$	$2.09 \cdot 10^4$	$4.76 \cdot 10^1$	-

Table 3: Floating point operations (FLOPs) used for running a series of test simulations.

error of the simulated concentration profiles: a Riemann error sum. More precisely, we show the sum of errors of the particle tracking method as compared to Bloechle's series solution at the 24 plotted sample points multiplied by the distance between the sample points. In these figures, the three particle-tracking methods all seem to do fairly well. What error does exist is chaotic and noisy, which suggests that the error is predominantly due to the natural randomness inherent to all particle tracking methods and can be reduced by using more particles.

We also examine the simulated fracture concentrations as a function of x. We take the true fracture concentration function to be that presented by Reimus et al. [2003], hereafter referred to as Reimus' solution. The effective parameters used in Reimus' solutions were obtained by comparing his effective fracture model directly with our own upscaled model in section 5.2. His solution was given in Laplace transform space and so it had to numerically inverted. We show the results of each particle tracking method simulation at 50 sample points as compared to Reimus' solution in Figure 7, Figure 8, Figure 9 and Figure 10. In Table 5, we again show a Riemann error sum. Specifically, we show the sum of errors of the particle tracking method as compared to Bloechle's series solution at the 50 plotted sample points multiplied by the distance between the sample points. The particle tracking methods all do fairly well here as well. Like before, the solutions have some noise that can be reduced by using more particles. The upscaled method also shows some systematic error, especially for parameter set 1. This systematic error is due to very large time steps taken by the upscaled method. We examine whether this error can be reduced by taking smaller time steps next.

6.2 Upscaled Method Step-Size Comparisons

We now look at how the error of the upscaled method varies with different maximum time steps. We run these simulations for parameter set 1 of Table 2 with end time $T = 10^4 \frac{a^2}{Df}$: it was with these conditions that the upscaled method exhibited the most error in the previous tests. As before, each of these simulations uses 10,000 particles.

In Figure 11, we show the fracture concentration functions produced by the upscaled method with different maximum time steps. We show the computational costs and a measure of the error for each of these upscaled simulations in Table 6. In both the figure and table, it can be seen that upscaled method performs as well the hi-res method and LaBolle's method did before for maximum time steps of $\triangle t_{max} = 10^{-1} \frac{a^2}{Df}$, $\triangle t_{max} = 10^{0} \frac{a^2}{Df}$ and $\triangle t_{max} = 10^{1} \frac{a^2}{Df}$. Only on increasing the maximum time step to $\triangle t_{max} = 10^{2} \frac{a^2}{Df}$ does the accuracy of the upscaled method fall significantly. With a time step of $\triangle t_{max} = 10^{1} \frac{a^2}{Df}$, the upscaled method produces equally good results as the hi-res method and LaBolle's method for this simulation at a fraction of the computational cost: the FLOP count of the upscaled method with that time step is 2% of LaBolle's method's FLOP count and less than 1% of the hi-res method's FLOP count.

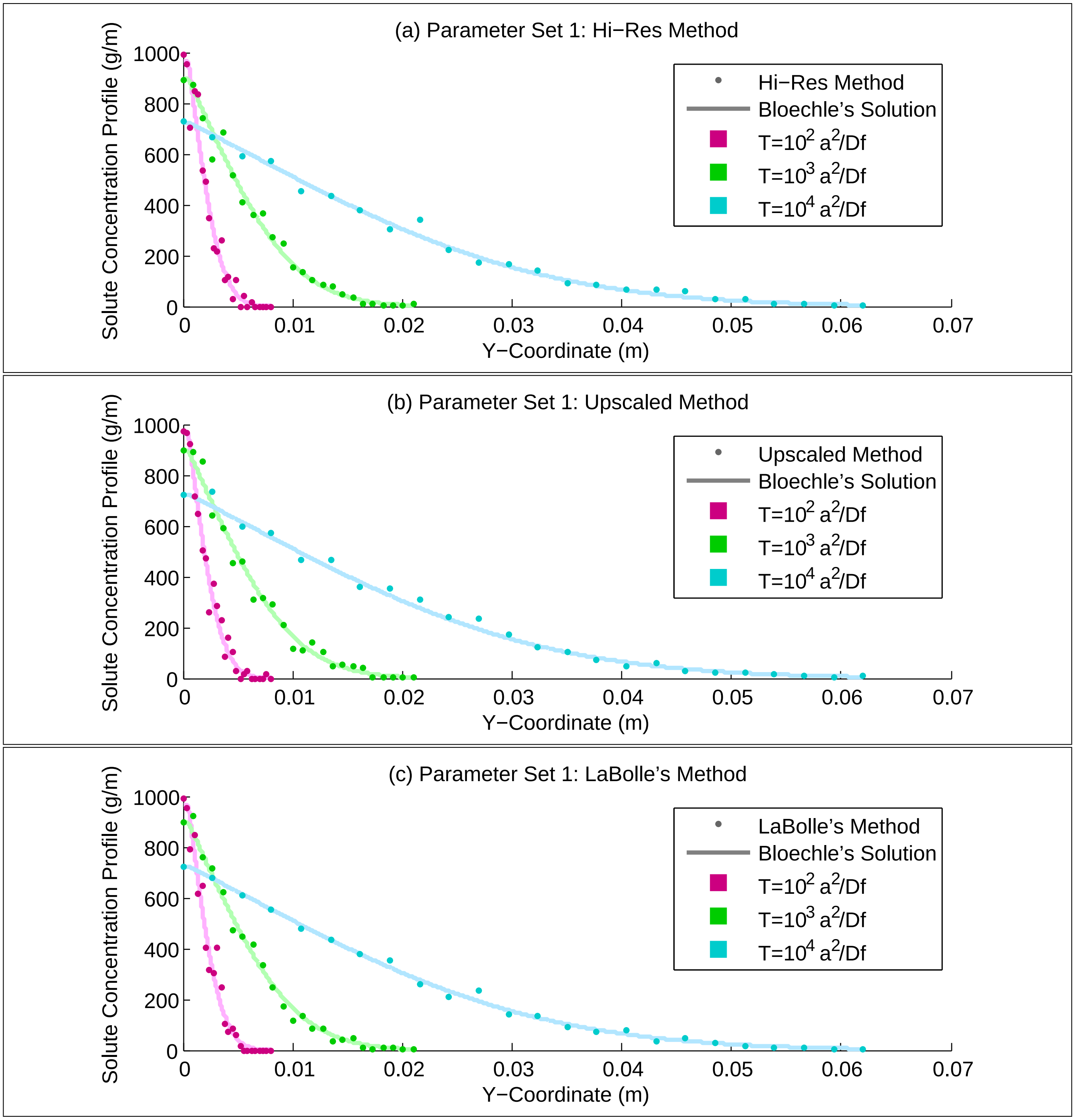


Figure 3: Simulated concentration profiles along the y-axis for parameter set 1 via (a) the hi-res method, (b) the upscaled method and (c) LaBolle's method.

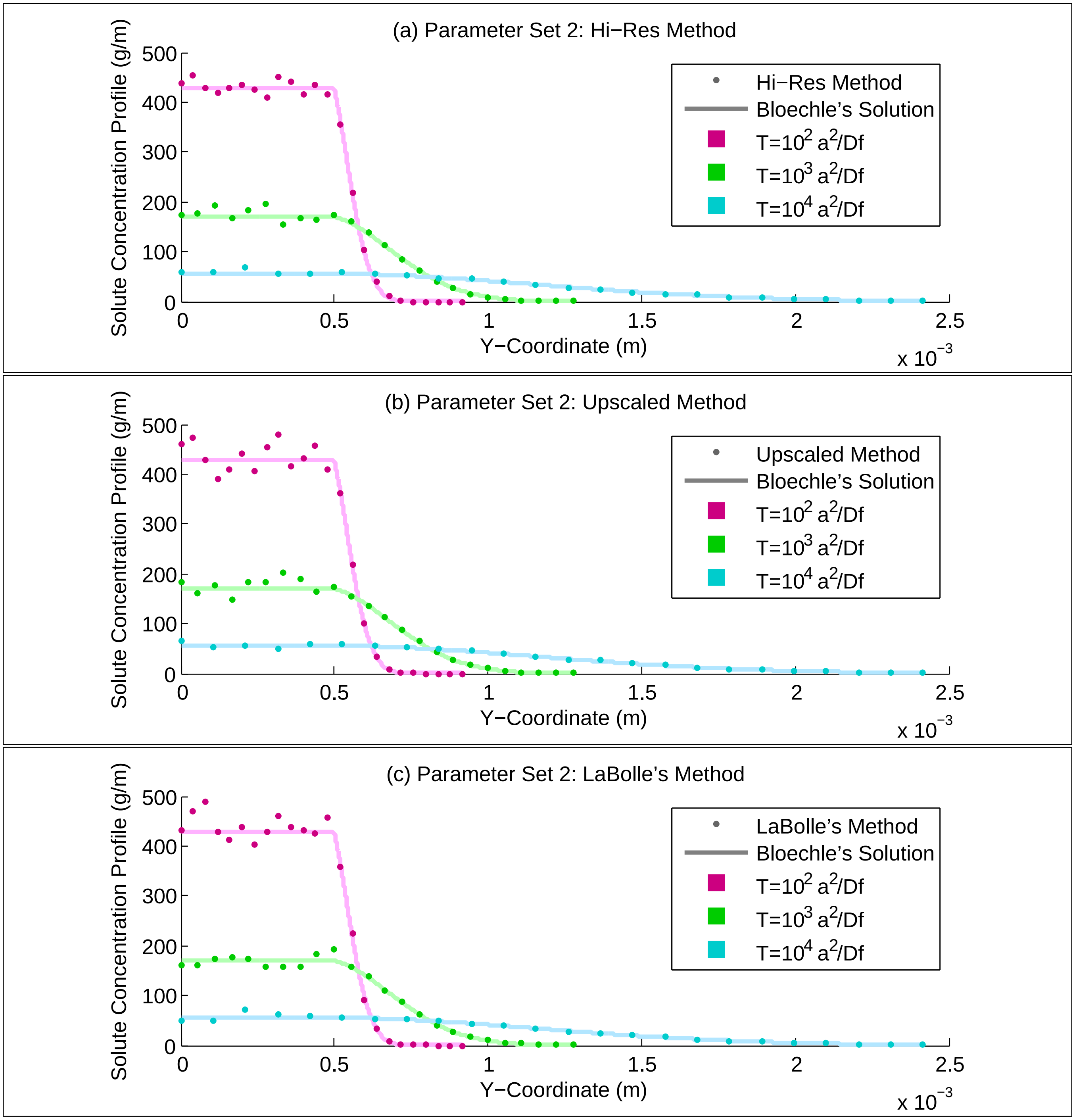
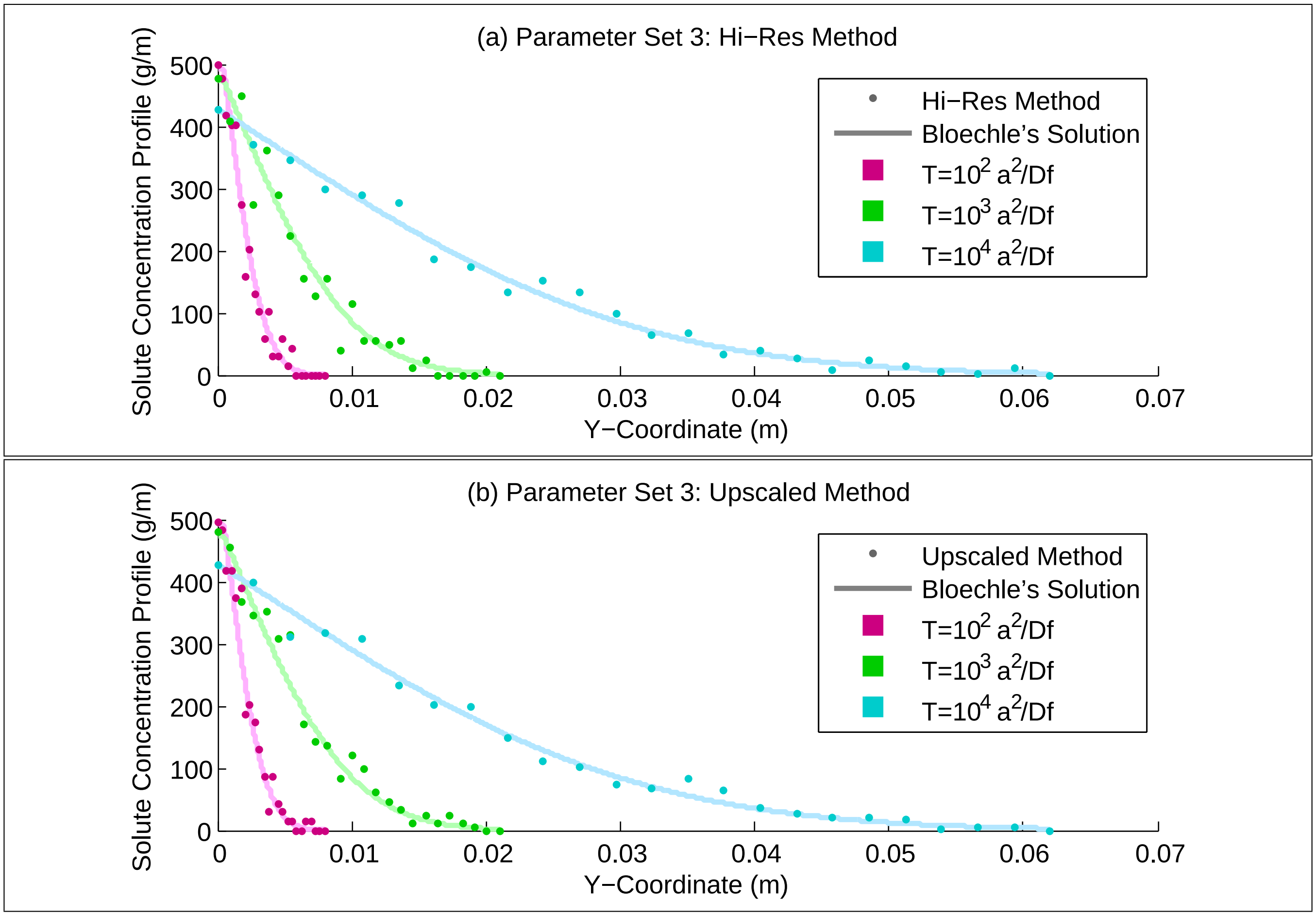
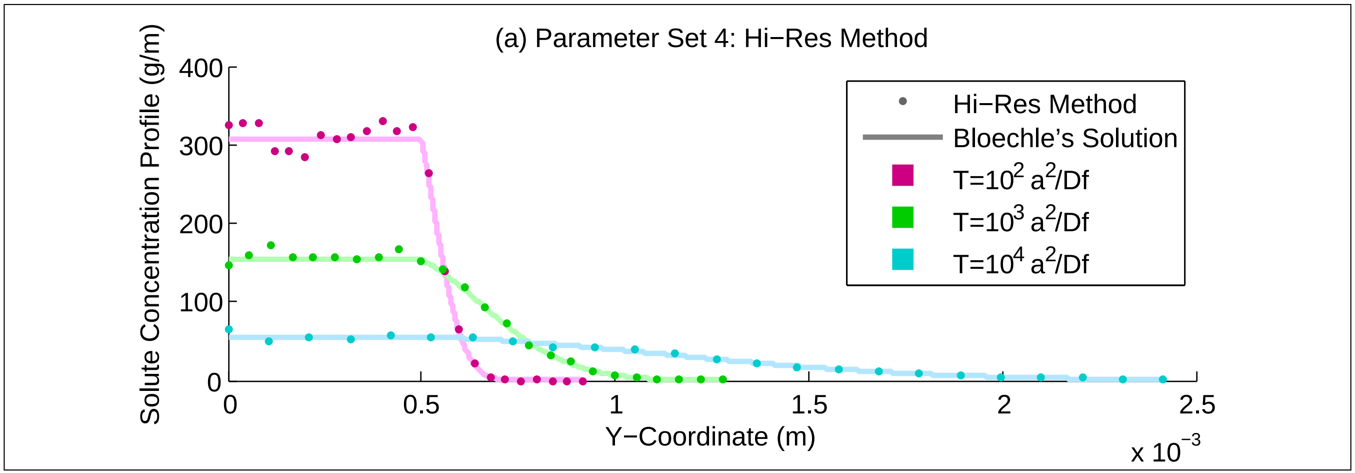


Figure 4: Simulated concentration profiles along the y-axis for parameter set 2 via (a) the hi-res method, (b) the upscaled method and (c) LaBolle's method.


(a) Parameter Set 3: Hi−Res Method

(b) Parameter Set 3: Upscaled Method

Figure 5: Simulated concentration profiles along the y-axis for parameter set 3 via (a) the hi-res method and (b) the upscaled method.


(a) Parameter Set 4: Hi−Res Method

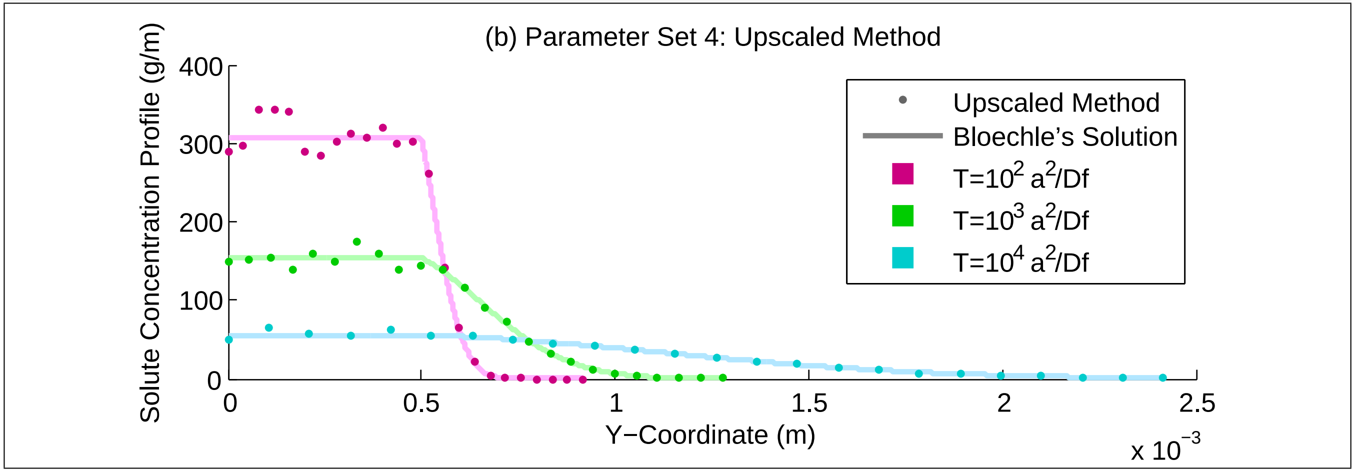

(b) Parameter Set 4: Upscaled Method

Figure 6: Simulated concentration profiles along the y-axis for parameter set 4 via (a) the hi-res method and (b) the upscaled method.

Simulation		Hi-res Method	Upscaled Method	LaBolle's Method
Parameter Set 1	$T = 10^2 \frac{a^2}{D_f}$	$3.28 \cdot 10^{-1}$	$2.54 \cdot 10^{-1}$	$3.78 \cdot 10^{-1}$
	$T = 10^3 \frac{a^2}{D_f}$	$4.33 \cdot 10^{-1}$	$5.25 \cdot 10^{-1}$	$4.00 \cdot 10^{-1}$
	$T = 10^4 \frac{a^2}{D_f}$	$8.61 \cdot 10^{-1}$	$9.84 \cdot 10^{-1}$	$6.30 \cdot 10^{-1}$
Parameter Set 2	$T = 10^2 \frac{a^2}{D_f}$	$6.86 \cdot 10^{-3}$	$1.26 \cdot 10^{-2}$	$1.03 \cdot 10^{-3}$
	$T = 10^3 \frac{a^2}{D_f}$	$6.72 \cdot 10^{-3}$	$8.14 \cdot 10^{-3}$	$6.73 \cdot 10^{-3}$
	$T = 10^4 \frac{a^2}{D_f}$	$3.66 \cdot 10^{-3}$	$3.35 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$
Parameter Set 3	$T = 10^2 \frac{a^2}{D_f}$	$1.57 \cdot 10^{-1}$	$1.51 \cdot 10^{-1}$	-
	$T = 10^3 \frac{a^2}{D_f}$	$4.96 \cdot 10^{-1}$	$3.73 \cdot 10^{-1}$	-
	$T = 10^4 \frac{a^2}{D_f}$	$7.53 \cdot 10^{-1}$	$6.50 \cdot 10^{-1}$	-
Parameter Set 4	$T = 10^2 \frac{a^2}{D_f}$	$7.26 \cdot 10^{-3}$	$8.95 \cdot 10^{-3}$	-
	$T = 10^3 \frac{a^2}{D_f}$	$4.34 \cdot 10^{-3}$	$5.47 \cdot 10^{-3}$	-
	$T = 10^4 \frac{a^2}{D_f}$	$4.03 \cdot 10^{-3}$	$3.91 \cdot 10^{-3}$	-

Table 4: The error measure for the simulated concentration profiles along the y-axis. The error measure is the summation of the absolute difference between Bloechle's analytical solution and the simulated distribution at 24 sample points multiplied by the distance between sample points.

Simulation		Hi-res Method	Upscaled Method	LaBolle's Method
Parameter Set 1	$T = 10^2 \frac{a^2}{D_f}$	$0.0393 \cdot 10^{-2}$	$0.0398 \cdot 10^{-2}$	$0.0412 \cdot 10^{-2}$
	$T = 10^3 \frac{a^2}{D_f}$	$4.11 \cdot 10^{-2}$	$1.05 \cdot 10^{-1}$	$3.42 \cdot 10^{-2}$
	$T = 10^4 \frac{a^2}{D_f}$	$3.97 \cdot 10^{-2}$	$1.20 \cdot 10^{-1}$	$3.97 \cdot 10^{-2}$
Parameter Set 2	$T = 10^2 \frac{a^2}{D_f}$	$3.40 \cdot 10^{-2}$	$3.96 \cdot 10^{-2}$	$4.97 \cdot 10^{-2}$
	$T = 10^3 \frac{a^2}{D_f}$	$1.69 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$	$1.92 \cdot 10^{-2}$
	$T = 10^4 \frac{a^2}{D_f}$	$1.28 \cdot 10^{-2}$	$1.20 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$
Parameter Set 3	$T = 10^2 \frac{a^2}{D_f}$	$3.95 \cdot 10^{-2}$	$4.74 \cdot 10^{-2}$	-
	$T = 10^3 \frac{a^2}{D_f}$	$3.78 \cdot 10^{-2}$	$2.65 \cdot 10^{-2}$	-
	$T = 10^4 \frac{a^2}{D_f}$	$3.70 \cdot 10^{-2}$	$7.67 \cdot 10^{-2}$	-
Parameter Set 4	$T = 10^2 \frac{a^2}{D_f}$	$2.81 \cdot 10^{-2}$	$3.01 \cdot 10^{-2}$	-
	$T = 10^3 \frac{a^2}{D_f}$	$2.02 \cdot 10^{-2}$	$2.06 \cdot 10^{-2}$	-
	$T = 10^4 \frac{a^2}{D_f}$	$1.18 \cdot 10^{-2}$	$8.85 \cdot 10^{-3}$	-

Table 5: The error measure for the simulated fracture concentration functions along the x-axis. The error measure is the summation of the absolute difference between Bloechle's analytical solution and the simulated distribution at 24 sample points multiplied by the distance between sample points.

Maximum Step Size	Computational Cost (MegaFLOP)	Fracture Concentration Error Measure
$\triangle t_{max} = 10^{-1} \frac{a^2}{Df}$	$9.33 \cdot 10^4$	$3.60 \cdot 10^{-2}$
$\triangle t_{max} = 10^{0} \frac{a^2}{Df}$	$9.70. \cdot 10^3$	$3.88 \cdot 10^{-2}$
$\triangle t_{max} = 10^{1} \frac{a^2}{Df}$	$9.44 \cdot 10^2$	$3.75 \cdot 10^{-2}$
$\triangle t_{max} = 10^{2} \frac{a^2}{Df}$	$1.00 \cdot 10^2$	$1.29 \cdot 10^{-1}$

Table 6: The simulation's FLOP count and the error measure of the produced fracture concentration function along the x-axis. The error measure is the summation of the absolute difference between Bloechle's analytical solution and the simulated distribution at 24 sample points multiplied by the distance between sample points.

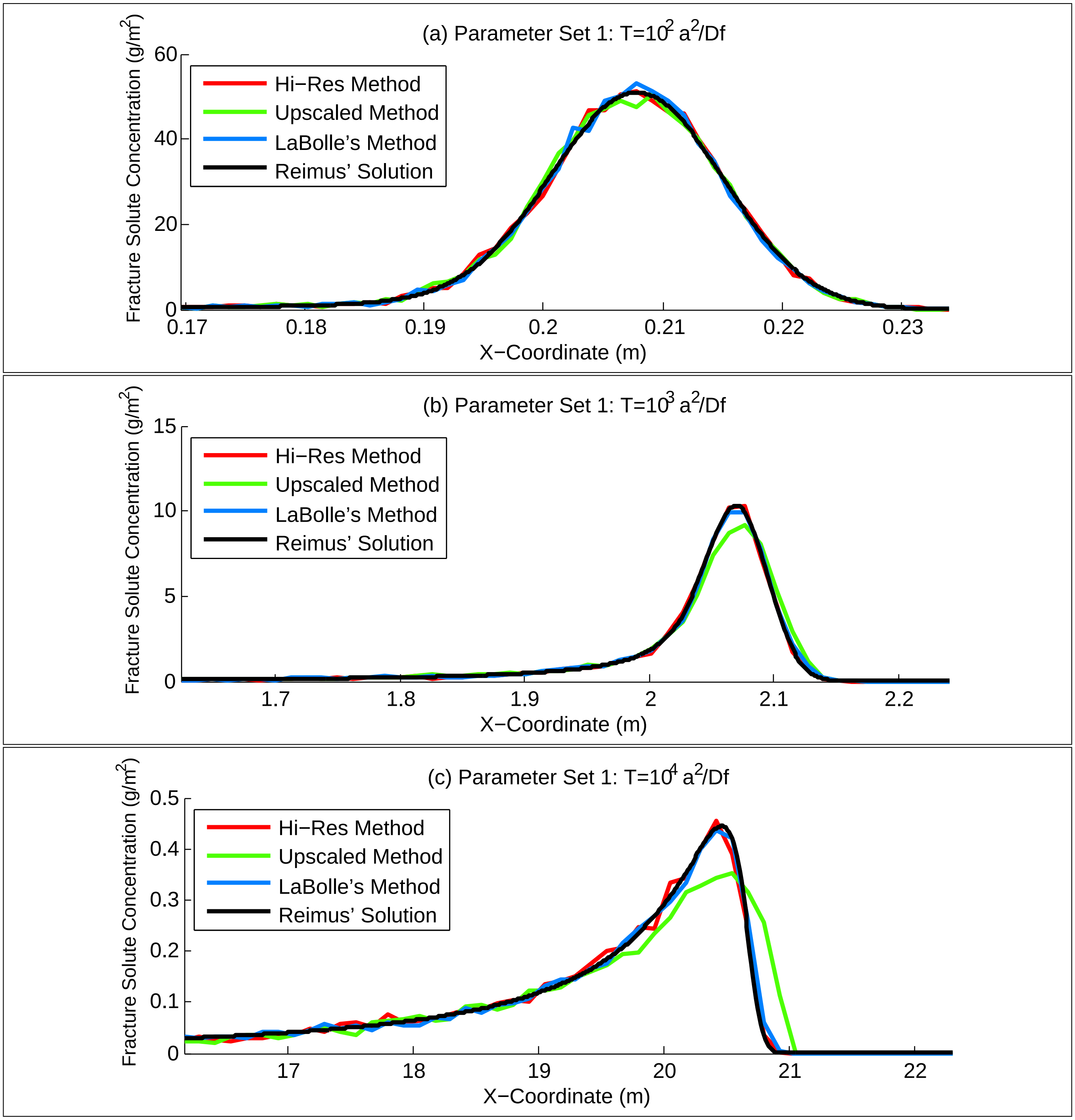


Figure 7: Simulated fracture concentrations along the x-axis for parameter set 1 at times (a) $T = 10^2 \frac{a^2}{D_f}$, (b) $T = 10^3 \frac{a^2}{D_f}$ and (c) $T = 10^4 \frac{a^2}{D_f}$.

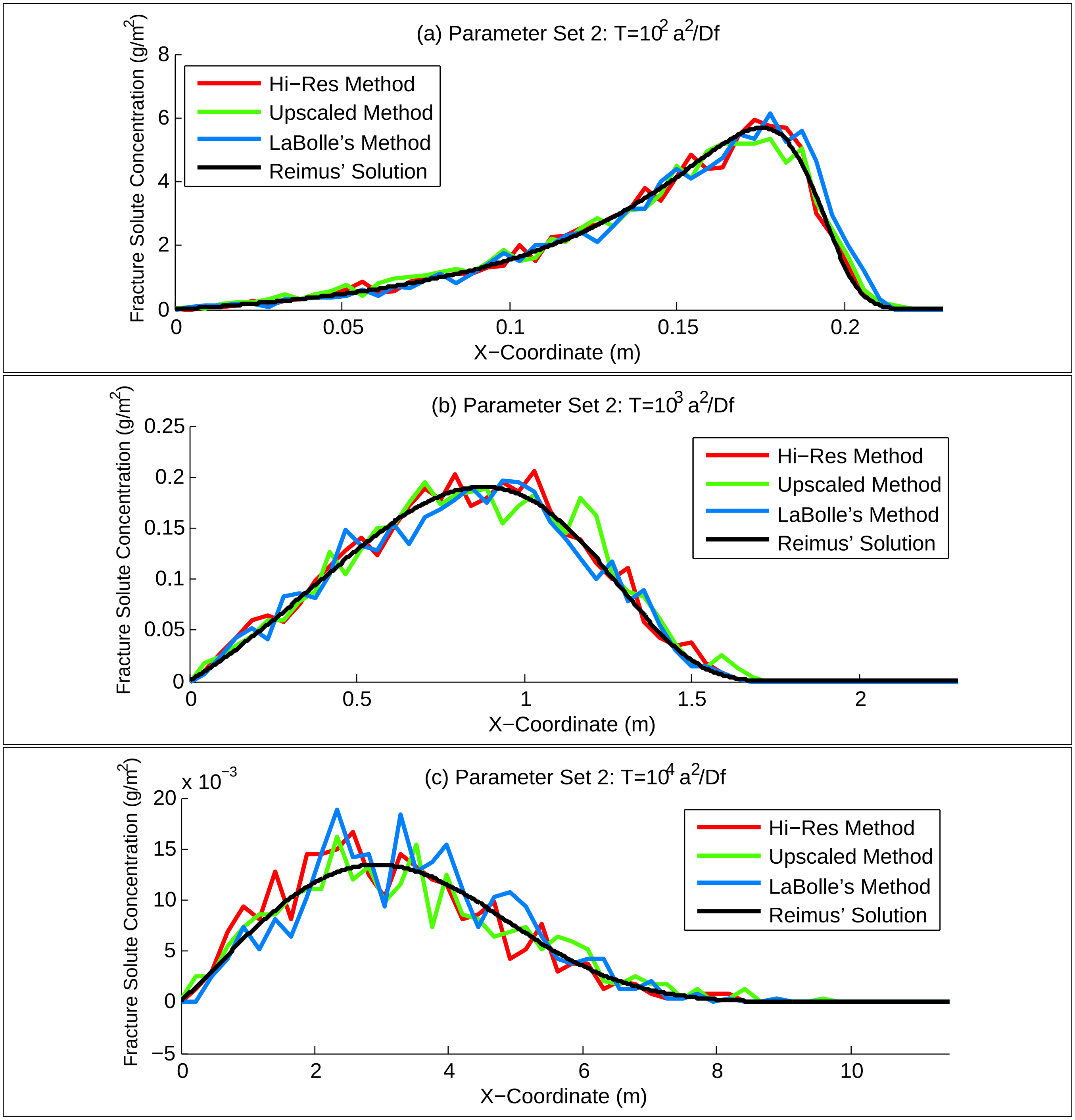


Figure 8: Simulated fracture concentrations along the x-axis for parameter set 2 at times (a) $T = 10^2 \frac{a^2}{D_f}$, (b) $T = 10^3 \frac{a^2}{D_f}$ and (c) $T = 10^4 \frac{a^2}{D_f}$.

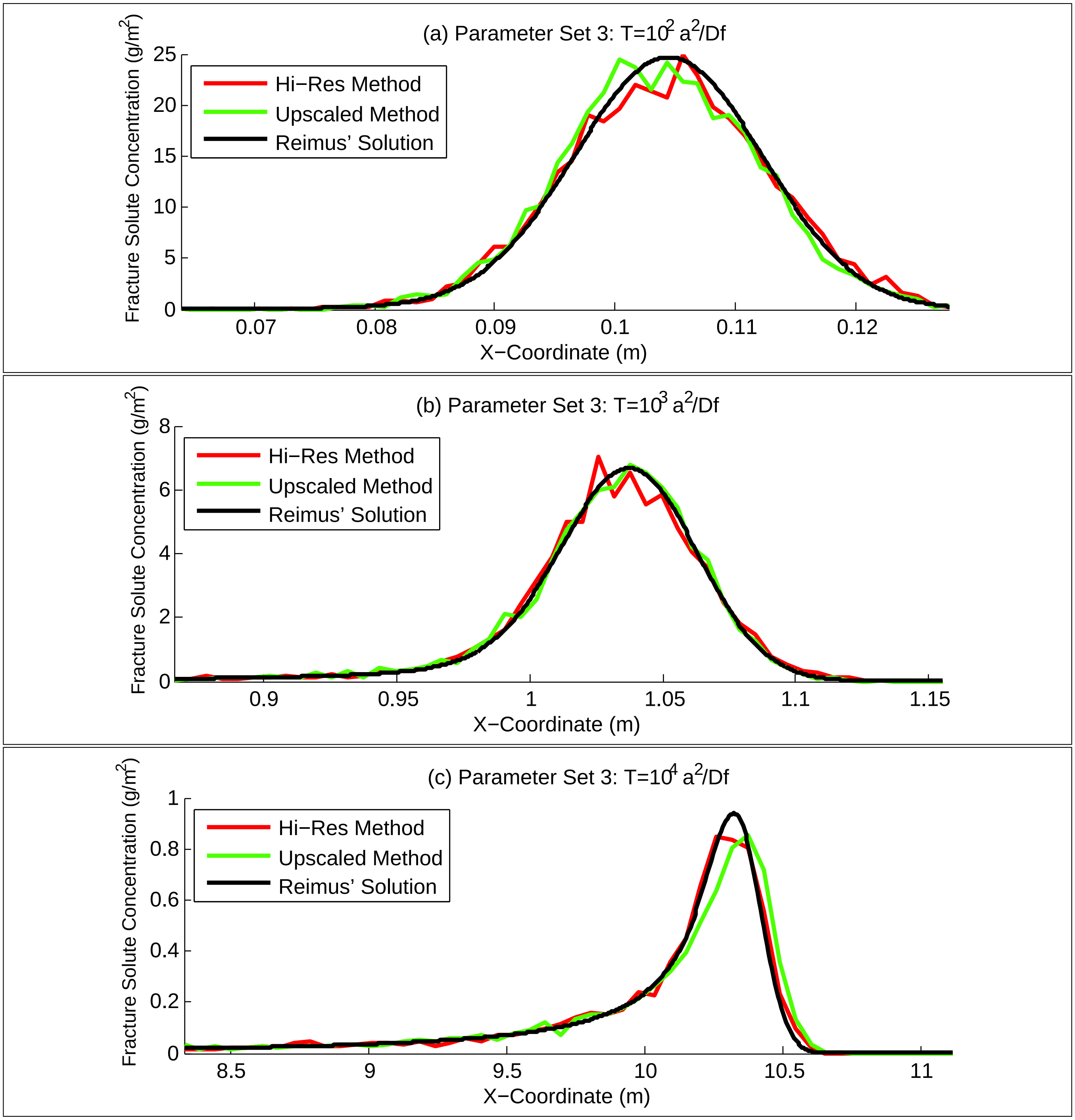


Figure 9: Simulated fracture concentrations along the x-axis for parameter set 3 at times (a) $T = 10^2 \frac{a^2}{D_f}$, (b) $T = 10^3 \frac{a^2}{D_f}$ and (c) $T = 10^4 \frac{a^2}{D_f}$.

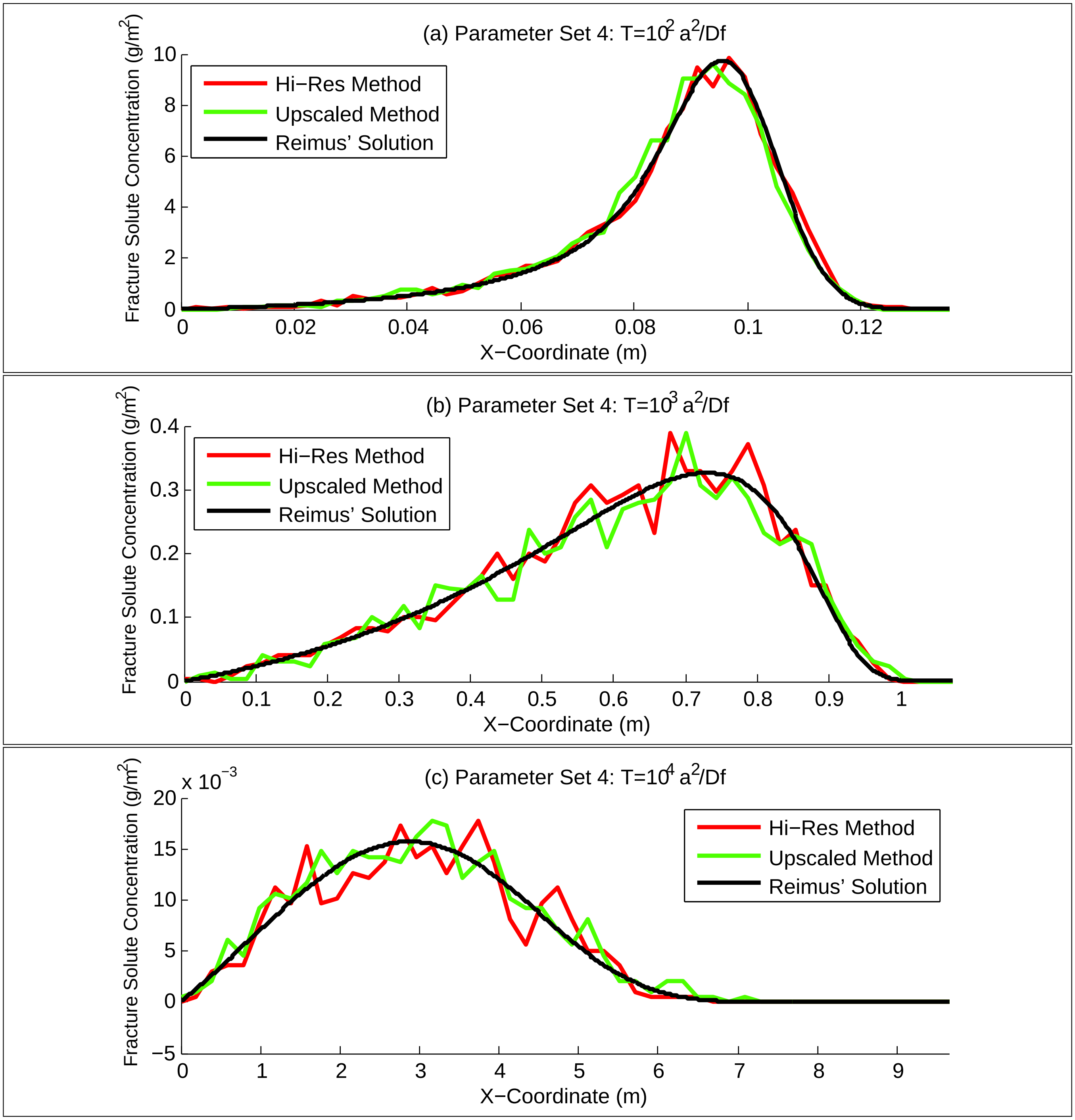


Figure 10: Simulated fracture concentrations along the x-axis for parameter set 4 simulated at times (a) $T = 10^2 \frac{a^2}{D_f}$, (b) $T = 10^3 \frac{a^2}{D_f}$ and (c) $T = 10^4 \frac{a^2}{D_f}$.

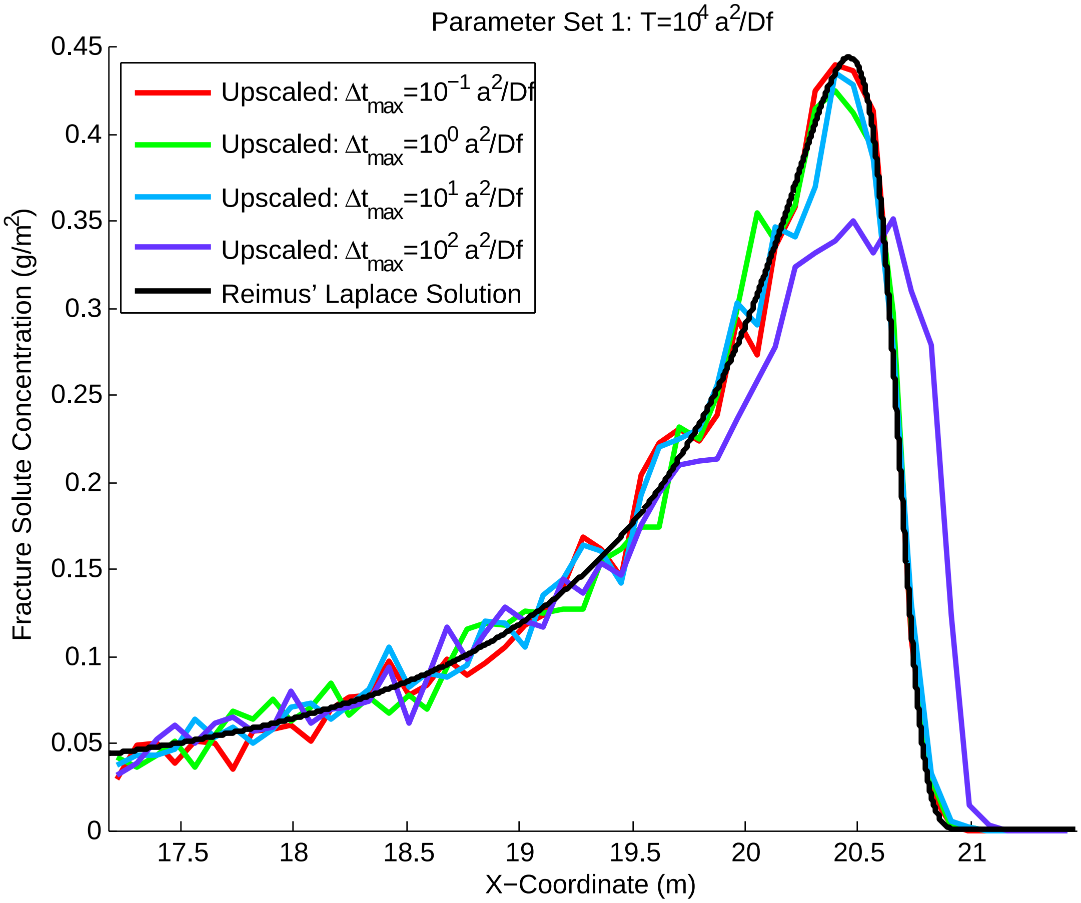


Figure 11: The fracture concentration along the x-axis for parameter set 1 at time $T = 10^4 \, \frac{a^2}{D_f}$ as simulated by the upscaled method using different maximum time steps.

6.3 Large Time-Scale Upscaled Simulations

Due to the computational efficiency of the upscaled method, we are able to test the method for larger end times. We here test the upscaled method with an end time of ten years. We test all four parameter sets presented in Table 2. For these simulations, we use a maximum time step length of one day. The number of particles simulated for each parameter set was chosen to roughly equalize the simulations' computational cost, as shown in Table 7.

We show the simulated concentration profiles along the y-axis produced in Figure 12 and Figure 13. In these figures, the upscaled method has near-perfect agreement with Bloechle's solution. We show the simulated fracture concentration functions in Figure 14 and Figure 15. Looking at these plots, the upscaled method again does quite well. There is, however, more random error for the fracture concentration functions than the concentration y-axis profiles. This is due to the number of particles in the fracture decreasing as the end time increases, leading to more random variance.

Simulation	Number of Particles Simulated	Computational Cost (MegaFLOP)
Parameter Set 1	10^5	$1.26 \cdot 10^4$
Parameter Set 2	10^6	$2.14 \cdot 10^4$
Parameter Set 3	10^5	$1.86 \cdot 10^4$
Parameter Set 4	10^6	$2.53 \cdot 10^4$

Table 7: Number of particles simulated and total floating point operations (FLOPs) used for upscaled simulations of a ten-year period.

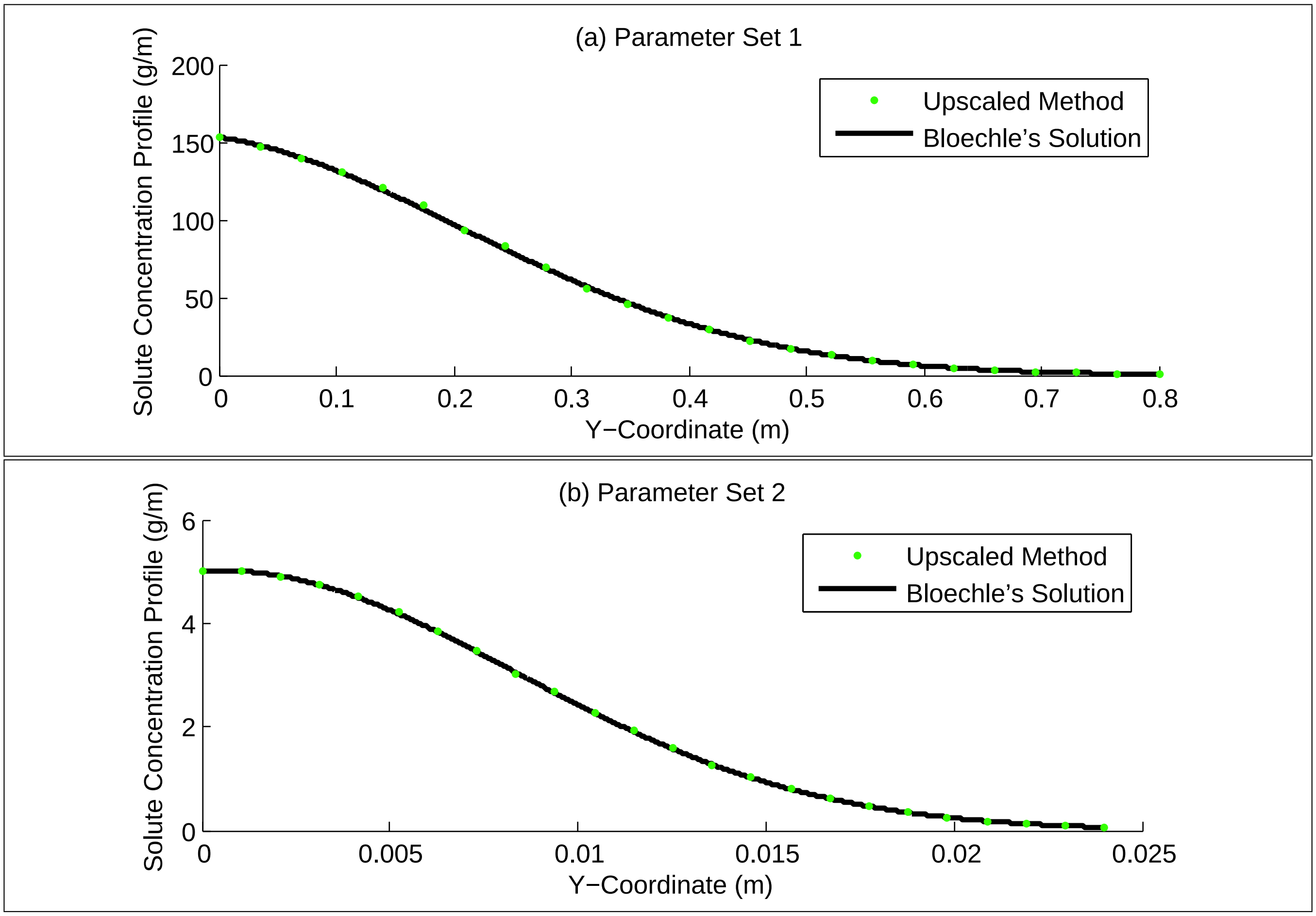
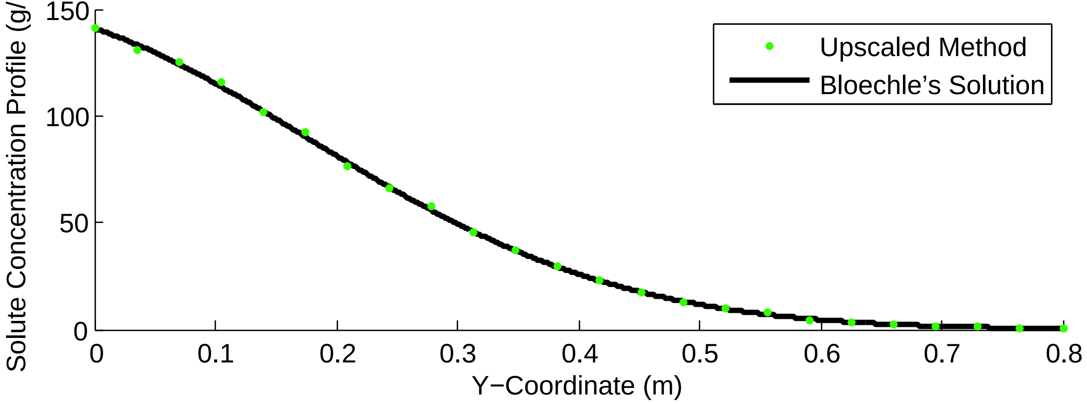


Figure 12: Concentration profiles along y-axis at time $T = 10$ years simulated via the upscaled method for parameter sets (a) 1 and (b) 2.

(a) Parameter Set 3



(b) Parameter Set 4

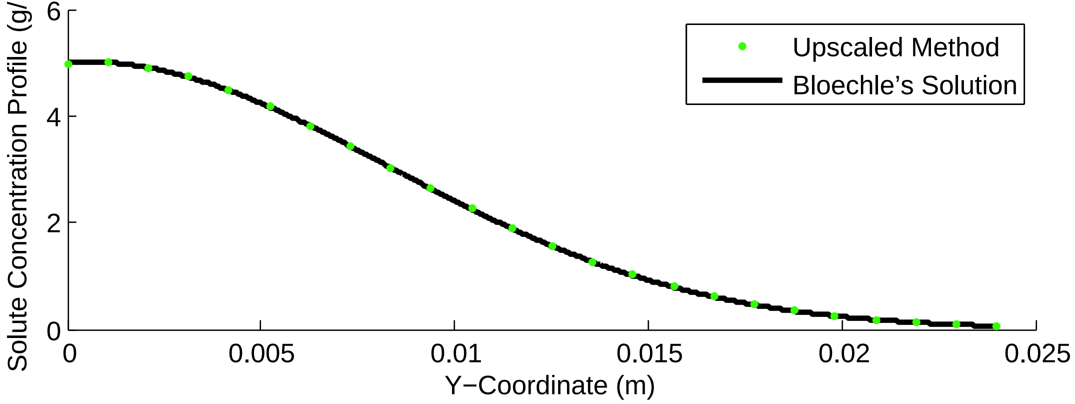


Figure 13: Concentration profiles along y-axis at time $T = 10$ years simulated via the upscaled method for parameter sets (a) 3 and (b) 4

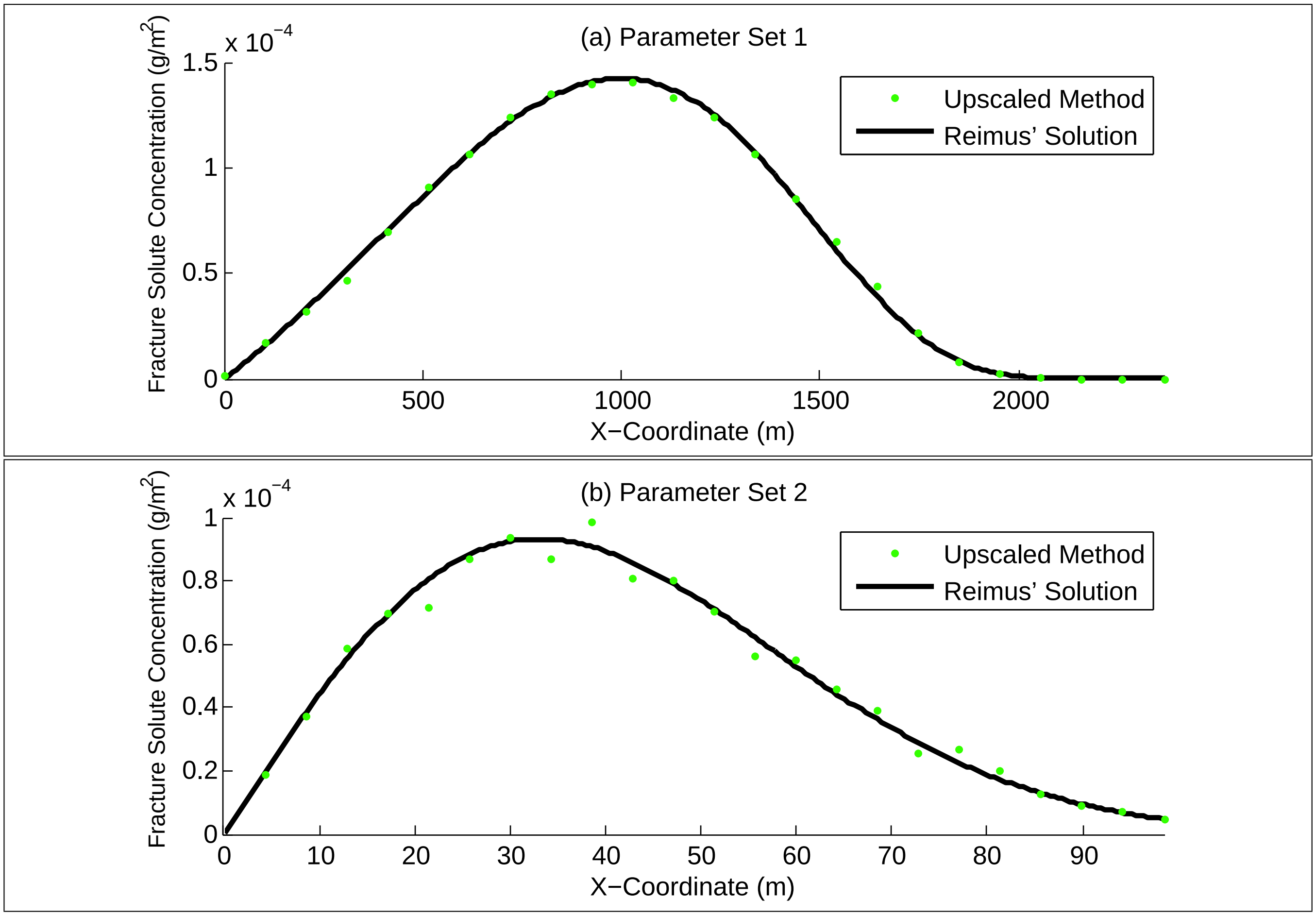


Figure 14: Fracture concentrations along x-axis at time $T = 10$ years simulated via the upscaled method for parameter sets (a) 1 and (b) 2.

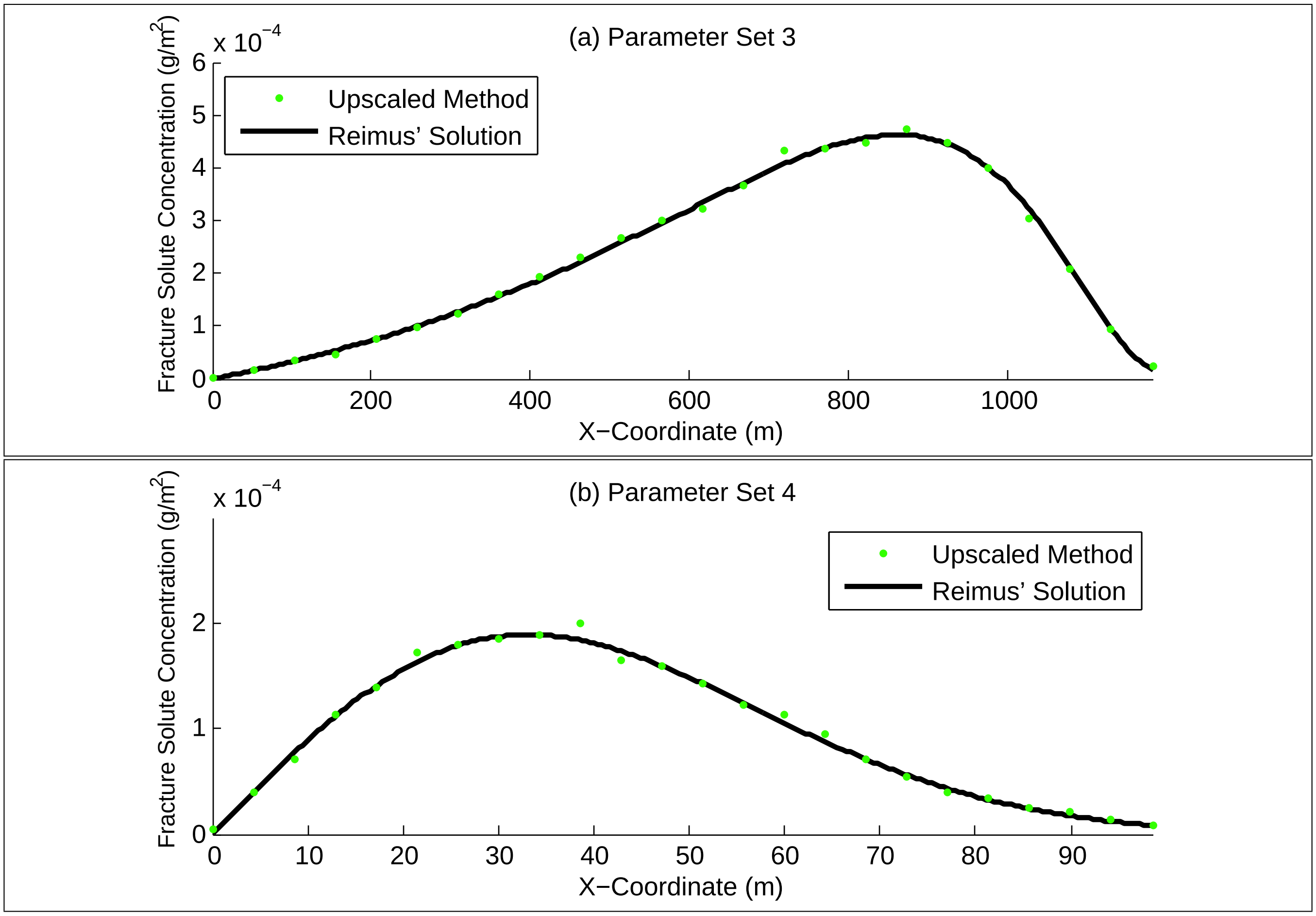


Figure 15: Fracture concentrations along x-axis at time $T = 10$ years simulated via the upscaled method for parameter sets (a) 3 and (b) 4.

7 Summary & Conclusions

In this thesis, we developed two particle tracking methods for a model of a fracture and the surrounding rock matrix that includes interface absorption: a hi-res method and an upscaled method.

The hi-res method models the movement of solute at a fine scale. In section 4.1, we developed an efficient method to advance particles in the fracture and matrix to a fixed time or until the particle hits an interface, whichever comes first. In section 4.2, we proved that the effect of interface absorption on a particle starting on the interface is to reduce the distance traveled by an exponentially distributed value. So that the effects of each interface can be handled separately, this method requires that the time steps be constrained so that particles cannot hit both interfaces in a single time step.

The upscaled method attempts to capture the particle's dominant behavior while ignoring some of the small-scale dynamics. In section 5.2, we demonstrated the key result that the fracture and adjoining interfaces together act much like a single absorptive interface in terms of solute transport. This result depends on D_f being much larger than D_m, a reasonable assumption for most real world fractures. This allows us to utilize the techniques already developed for the hi-res method at a macro-scale. The upscaled method is consequently capable of taking much larger time steps than the hi-res method.

To test the hi-res and upscaled method, we compare their results to some numerical results from the literature for a variety of end times and parameters. We also include results produced the particle tracking method presented by LaBolle, Fogg, and Tompson [1996] as a reference point in terms of accuracy and computational cost for cases with no interface absorption. In terms of computational cost, the upscaled method consistently used orders of magnitude less FLOPs due to the larger time steps while still generating results of equal accuracy to the other methods. The hi-res method typically used several times more FLOPs than LaBolle's method, due to increased computational cost of accounting for interface absorption. As for accuracy, all three particle tracking methods produced concentration profiles across the fracture that agreed with analytical series solutions for the concentration profiles presented by Bloechle [2001]. The particle tracking methods results also demonstrated agreement with the fracture concentration solution presented by Reimus et al. [2003]. It was, however, sometimes necessary to adjust the upscaled method's maximum step size to obtain this agreement. Finally, the upscaled method proved to be capable of efficiently carrying out decade-long simulations while still producing results consistent with the solutions from the literature.

We have thus succeeded in developing particle tracking methods that account for both interface absorption and matrix diffusion. In particular, the upscaled method appears to be a very efficient and powerful method for simulating solute transport in fractures.

References

[1] Bloechle, B. W., On the Taylor Dispersion of Reactive Solutes in a Parallel-Plate Fracture-Matrix System. PhD thesis, University of Colorado at Boulder-Department of Applied Mathematics, 2001.

[2] Chang, J., "Stochastic Processes." Yale University Department of Statistics . Yale. Web. 23 Nov 2013. <http://www.stat.yale.edu/~jtc5/251/stochastic-processes.pdf>.

[3] Chambers, J. M., C. L. Mallows, and B. W. Stuck, "A method for simulating stable random variables," J. Am. Stat. Soc., vol. 71, no. 354, pp. 340-344, Jun. 1976.

[4] Einstein, A., Investigations of the Theory of Brownian Movement. Dover, 1956.

[5] Jacobs, K., Stochastic Processes for Physicists. Cambridge University Press. pp. 57–59. ISBN 9781139486798. 2010.

[6] LaBolle, E. M., G. E. Fogg, and A. F. B. Tompson, Random-walk simulation of transport in heterogeneous porous media: Local mass-conservation problem and implementation methods. Water Resour. Res., 32(3), 583-593, 0043-1397/96/95WR-03528, 1996.

[7] Maloszewski, P., A. Zuber, Mathematical Modeling of Tracer Behavior in Short-Term Experiments in Fissured Rocks. Water Resour. Res., 26(7), 1517-1528, 0043-1397/90/90WR-003525, July 1990.

[8] Reimus, P., G. Pohll, T. Mihevc, J. Chapman, M. Haga, B. Lyles, S. Kosinski, R. Niswonger, and P. Sanders, Testing and parameterizing a conceptual model for solute transport in a fractured granite using multiple tracers in a forced-gradient test. Water Resour. Res., 39(12), 1356, 0010-1029/2002WR-001597, 2003.

Appendix A

In this appendix, we shall prove equation (4.1.12), a statement that two random variables are equal in distribution. For completeness, we reproduce the equation here:

$$\left[\frac{|Y_{max}|+|y_0|}{\sqrt{2D_f\cdot\triangle t_{max}}}\middle|\, Hitting\right] \underset{dist.}{=} \left[Z\,\middle|\, Z\geq\frac{|y_0|}{\sqrt{2D_f\cdot\triangle t_{max}}}\right] \tag{A.1}$$

Many of these terms in this equation were defined specifically in the course of developing the one-dimensional single-interface particle tracking method for the fracture and we also reproduce those definitions here. The initial y-coordinate of the particle, y_0, must be less than zero because the particle is assumed to start the time step in the fracture. In equation (4.1.5), the pdf of Y_{max} was defined as:

$$f_{Y_{max}}(y)=\frac{1}{\sqrt{4D_f\cdot\triangle t_{max}}}e^{-\frac{(y-y_0)^2}{4D_f\cdot\triangle t_{max}}} \tag{A.2}$$

The event "*Hitting*" was defined the particle hitting the interface within the maximum time step. Equation (4.1.6) gave the probability of "*Hitting*" given Y_{max}:

$$P(Hitting|Y_{max}=y)=\begin{cases}e^{-\frac{y_0\cdot y}{D_f\cdot\triangle t_{max}}} & y<0\\ 1 & y\geq 0\end{cases} \tag{A.3}$$

With the relevant terms defined, we now turn to proving equation (A.1) by determining two random variables' pdf up to a constant. We shall first determine the pdf of the the right-hand side of equation (A.1) up to a constant. We use the pdf form of Bayes' theorem to reformulate the random variable's pdf:

$$\begin{aligned} f_{Z\middle|Z>\frac{|y_0|}{\sqrt{2D_f\cdot\triangle t_{mac}}}}(y) &= \frac{f_Z(y)\cdot P\left(Z\geq\frac{|y_0|}{\sqrt{2D_f\cdot\triangle t_{macx}}}\middle|\, Z=y\right)}{P\left(Z\geq\frac{|y_0|}{\sqrt{2D_f\cdot\triangle t_{max}}}\right)}\\ &\propto f_Z(y)\cdot P\left(Z\geq\frac{|y_0|}{\sqrt{2D_f\cdot\triangle t_{max}}}\middle|\, Z=y\right) \end{aligned} \tag{A.4}$$

Both of the terms in this last expression are well known. The pdf of Z, a standard normal random variable, is:

$$f_Z(y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} \tag{A.5}$$

The probability of $Z \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}}$ being true given that $Z = y$ is trivially:

$$P\left(Z \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \middle| Z = y \right) = \begin{cases} 0 & if\; y < \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \\ 1 & if\; y \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \end{cases} \tag{A.6}$$

We substitute these terms into equation (A.4) to get:

$$f_{Z \middle| Z > \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{mac}}}}(y) \propto \begin{cases} 0 & y < \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \\ \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} & y \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \end{cases} \tag{A.7}$$

We have thus found the pdf of the right-hand side up to a constant. We now turn to finding the pdf of the left-hand side of equation (A.1) up to a constant. As before, we use the pdf form of Bayes' theorem to expand the variable's pdf:

$$\begin{aligned} f_{Y_{max}|Hitting}(y) =& \frac{f_{Y_{max}}(y) \cdot P(Hitting\,|Y_{max} = y)}{P(Hitting)} & \text{(A.8)} \\ \propto& f_{Y_{max}}(y) \cdot P(Hitting\,|Y_{max} = y) & \text{(A.9)} \end{aligned}$$

The two terms in the last expression are already given by equations (A.2) and (A.3). We substitute these expressions into equation (A.8) to get the pdf of Y_{max}:

$$\begin{aligned} f_{Y_{max}|Hitting}(y) \propto& f_{Y_{max}}(y) \cdot P(Hitting\,|Y_{max} = y) \\ \propto& \begin{cases} \frac{1}{\sqrt{4\pi D_f \cdot \triangle t_{max}}} e^{-\frac{(y-y_0)^2}{4D_f \cdot \triangle t_{max}}} \cdot e^{-\frac{y_0 \cdot y}{D_f \cdot \triangle t_{max}}} & y < 0 \\ \frac{1}{\sqrt{4\pi D_f \cdot \triangle t_{max}}} e^{-\frac{(y-y_0)^2}{4D_f \cdot \triangle t_{max}}} \cdot 1 & y \geq 0 \end{cases} \\ \propto& \begin{cases} \frac{1}{\sqrt{4\pi D_f \cdot \triangle t_{max}}} e^{-\frac{(y+y_0)^2}{4D_f \cdot \triangle t_{max}}} & y < 0 \\ \frac{1}{\sqrt{4\pi D_f \cdot \triangle t_{max}}} e^{-\frac{(y-y_0)^2}{4D_f \cdot \triangle t_{max}}} & y \geq 0 \end{cases} \end{aligned} \tag{A.10}$$

The pdf of $|Y_{max}|$ given the particle hits the interface, the absolute value of the above, is then:

$$f_{|Y_{max}||Hitting}(y) \propto \begin{cases} 0 & y < 0 \\ \frac{1}{\sqrt{2\pi D_f \cdot \triangle t_{max}}} e^{-\frac{(y-y_0)^2}{4D_f \cdot \triangle t_{max}}} & y \geq 0 \end{cases} \tag{A.11}$$

Finally $\frac{|Y_{max}|+|y_0|}{\sqrt{2D_f \cdot \triangle t}}$ given the particle has hit the interface is just the above random variable shifted and scaled. Its pdf is:

$$f_{\frac{|Y_{max}|+|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}}\Big| Hitting}(y) \propto \begin{cases} 0 & y < \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \\ \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} & y \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}} \end{cases} \tag{A.12}$$

This is exactly what we found $Z \Big| Z \geq \frac{|y_0|}{\sqrt{2D_f \cdot \triangle t_{max}}}$ to be proportional to. The pdfs of the two random variables are therefore equal up to a constant. As two functions cannot differ by a constant factor and still both be normalized pdfs, the two pdfs must be equal. We have thus proved that the two random variables are equal in distribution.

Appendix B

In this appendix, we solve for a point mass starting on the interface of the one-dimensional single-interface system, expressed by equations (3.3.1), (3.3.2), (3.3.3) and (4.2.1). For completeness, we reproduce those equations here. The mass transport equations are:

$$\begin{aligned}
\frac{\partial \rho_f}{\partial t} &= D_f \frac{\partial^2 \rho_f}{\partial y^2} && for\, y < 0 && \text{(B.1a)}\\
\frac{\partial \rho_m}{\partial t} &= D_m \frac{\partial^2 \rho_m}{\partial y^2} && for\, y > 0 && \text{(B.1b)}\\
\rho_f &= \frac{1}{\varphi} \rho_m && for\, y = 0 && \text{(B.1c)}\\
k_a \frac{\partial \rho_f}{\partial t} &= D_m \frac{\partial \rho_m}{\partial y} - D_f \frac{\partial \rho_f}{\partial y} && for\, y = 0 && \text{(B.1d)}
\end{aligned}$$

The interface mass density is defined by:

$$\rho_i(y,t) = k_a \cdot \rho_f(y,t) \cdot \delta_0(y) \tag{B.2}$$

The global mass density is:

$$\rho(x,y,t) = \begin{cases} \rho_f(y,t) & y < 0 \\ \rho_m(y,t) & y > 0 \\ \rho_i(y,t) & y = 0 \end{cases} \tag{B.3}$$

Finally, the initial mass density is simply a point mass at the origin:

$$\rho(y, t = 0) = \delta_0(y) \tag{B.4}$$

As we are placing the point mass right on the boundary, we must proceed with great care in translating the initial conditions from a global mass density function, ρ, to the regional mass density functions compatible with the mass transport equations: ρ_f, ρ_m, and ρ_i. Using the global mass density definition, equations (B.3), we may infer the regional mass densities at most places

$$\begin{aligned}
\rho_f(y, t = 0) &= 0 \; for\, y < 0 && \text{(B.5a)}\\
\rho_m(y, t = 0) &= 0 \; for\, y > 0 && \text{(B.5b)}\\
\rho_i(y, t = 0) &= \delta_0(y) \; for\, y = 0 && \text{(B.5c)}
\end{aligned}$$

However, ρ_f and ρ_m are also defined at $y = 0$; otherwise, equation (B.1c) would not make sense. Using equation (B.2), we can calculate the initial fracture mass density at the origin:

$$\rho_f(y = 0, t = 0) = \frac{1}{k_a} \tag{B.6}$$

Using the interface boundary condition, equation (B.1c), we can then find the initial matrix mass density at the origin:

$$\rho_m(y = 0, t = 0) = \frac{\varphi}{k_a} \tag{B.7}$$

Combining equations (B.5), (B.6) and (B.7), we get the full initial regional mass density functions:

$$\rho_f(y, t = 0) = \begin{cases} \frac{1}{k_a} & for\ y = 0 \\ 0 & for\ y < 0 \end{cases} \tag{B.8a}$$

$$\rho_m(y, t = 0) = \begin{cases} \frac{\varphi}{k_a} & for\ y = 0 \\ 0 & for\ y > 0 \end{cases} \tag{B.8b}$$

$$\rho_i(y, t = 0) = \delta_0(0) \quad for\ y = 0 \tag{B.8c}$$

We now take the Laplace transform of the governing system of PDEs with respect to t. We will denote the Laplace transform with respect to t with a tilde. The corresponding Laplace parameter is s. The system in Laplace space is then:

$$s\widetilde{\rho_f} - \rho_f(y, t = 0) = D_f \frac{\partial^2 \widetilde{\rho_f}}{\partial y^2} \qquad for\ y < 0 \tag{B.9a}$$

$$s\widetilde{\rho_m} - \rho_m(y, t = 0) = D_m \frac{\partial^2 \widetilde{\rho_m}}{\partial y^2} \qquad for\ y > 0 \tag{B.9b}$$

$$\widetilde{\rho_f} = \frac{1}{\varphi} \widetilde{\rho_m} \qquad for\ y = 0 \tag{B.9c}$$

$$k_a \cdot s\widetilde{\rho_f} - k_a \cdot \rho_f(y, t = 0) = D_m \frac{\partial \widetilde{\rho_m}}{\partial y} - D_f \frac{\partial \widetilde{\rho_f}}{\partial y} \qquad for\ y = 0 \tag{B.9d}$$

We now substitute in the reformulated set of initial conditions to obtain:

$$s\widetilde{\rho_f} = D_f \frac{\partial^2 \widetilde{\rho_f}}{\partial y^2} \qquad for\ y < 0 \tag{B.10a}$$

$$s\widetilde{\rho_m} = D_m \frac{\partial^2 \widetilde{\rho_m}}{\partial y^2} \qquad for\ y > 0 \tag{B.10b}$$

$$\widetilde{\rho_f} = \frac{1}{\varphi}\widetilde{\rho_m} \qquad for\ y = 0 \tag{B.10c}$$

$$k_a \cdot s\widetilde{\rho_f} - 1 = D_m \frac{\partial \widetilde{\rho_m}}{\partial y} - D_f \frac{\partial \widetilde{\rho_f}}{\partial y} \qquad for\ y = 0 \tag{B.10d}$$

The first two equations are now ordinary differential equations. Solving them yields that the Laplace-space solutions are of the following form:

$$\widetilde{\rho_f} = c_1 e^{\sqrt{\frac{s}{D_f}}y} + c_2 e^{-\sqrt{\frac{s}{D_f}}y} \qquad for\ y < 0 \tag{B.11a}$$

$$\widetilde{\rho_m} = c_3 e^{\sqrt{\frac{s}{D_m}}y} + c_4 e^{-\sqrt{\frac{s}{D_m}}y} \qquad for\ y > 0 \tag{B.11b}$$

where c_1, c_2, c_3, and c_4 are undetermined constants. We now reason that solute mass density must converge to zero as y goes to $\pm\infty$ by physical reasoning. It follows that the Laplace transform of the solute mass density must also go to zero as y goes to $\pm\infty$. This will only be true for our general solution if $c_2 = 0$ and $c_3 = 0$. We plug this into the general solution form above to get:

$$\widetilde{\rho_f} = c_1 e^{\sqrt{\frac{s}{D_f}}y} \qquad for\ y < 0 \tag{B.12a}$$

$$\widetilde{\rho_m} = c_4 e^{-\sqrt{\frac{s}{D_m}}y} \qquad for\ y > 0 \tag{B.12b}$$

To determine the last two constants, we plug these general solutions into the equations (B.9c) and (B.9d) and solve. The result is:

$$c_1 = \frac{1}{\varphi} c_4$$

$$k_a s \cdot c_1 - 1 = -\sqrt{\frac{s}{D_m}} D_m . c_4 - \sqrt{\frac{s}{D_f}} D_f \cdot c_1$$

Algebraically solving this linear system of equations gives the values of $c_1$and c_4:

$$\begin{aligned}
c_1 =& \frac{1}{\sqrt{s}\left(k_a\sqrt{s}+\sqrt{D_f}+\varphi\sqrt{D_m}\right)}\\
c_4 =& \frac{\varphi}{\sqrt{s}\left(k_a\sqrt{s}+\sqrt{D_f}+\varphi\sqrt{D_m}\right)}
\end{aligned}$$

With these last constants determined, we now have the solution in Laplace space:

$$\widetilde{\rho_f} = \frac{e^{\sqrt{\frac{s}{D_f}}y}}{\sqrt{s}\left(k_a\sqrt{s}+\sqrt{D_f}+\varphi\sqrt{D_m}\right)} \qquad for\; y < 0 \tag{B.15a}$$

$$\widetilde{\rho_m} = \frac{\varphi e^{-\sqrt{\frac{s}{D_m}}y}}{\sqrt{s}\left(k_a\sqrt{s}+\sqrt{D_f}+\varphi\sqrt{D_m}\right)} \qquad for\; y > 0 \tag{B.15b}$$

We define a new parameter at this point for convenience:

$$b = \sqrt{D_f} + \varphi\sqrt{D_m} \tag{B.16}$$

Using this parameter, we can express the Laplace space solutions more concisely as:

$$\widetilde{\rho_f} = \frac{e^{\sqrt{\frac{s}{D_f}}y}}{\sqrt{s}\left(k_a\sqrt{s}+b\right)} \qquad for\; y < 0 \tag{B.17a}$$

$$\widetilde{\rho_m} = \frac{\varphi e^{-\sqrt{\frac{s}{D_m}}y}}{\sqrt{s}\left(k_a\sqrt{s}+b\right)} \qquad for\; y > 0 \tag{B.17b}$$

It is difficult to just take the inverse Laplace transform of these functions. We shall instead take a more circuitous route and take a second Laplace transform of these functions, this time with respect to y. However, this Laplace transform is taken over the non-negative values of y while the the fracture mass density function is defined for non-positive y. We avoid this by defining $y' = -y$ and substituting this variable into equation (B.17a), the fracture's solution:

$$\begin{aligned}
\widetilde{\rho_f} =& \frac{e^{-\sqrt{\frac{s}{D_f}}y'}}{\sqrt{s}\left(k_a\sqrt{s}+b\right)} && for\; y' > 0\\
\widetilde{\rho_m} =& \frac{\varphi e^{-\sqrt{\frac{s}{D_m}}y}}{\sqrt{s}\left(k_a\sqrt{s}+b\right)} && for\; y > 0
\end{aligned}$$

We now take the Laplace transform of fracture and matrix mass density functions with respect to y' and y. We will denote these Laplace transforms with a bar. The Laplace parameter will be z. The doubly Laplace transformed functions then:

$$\overline{\widetilde{\rho_f}} = \frac{1}{\sqrt{s}\left(k_a\sqrt{s}+b\right)\left(z+\sqrt{\frac{s}{D_f}}\right)} \tag{B.18a}$$

$$\overline{\widetilde{\rho_m}} = \frac{\varphi}{\sqrt{s}\left(k_a\sqrt{s}+b\right)\left(z+\sqrt{\frac{s}{D_m}}\right)} \tag{B.18b}$$

As Laplace transforms are linear, we may interchange the order of the Laplace transforms:

$$\widetilde{\overline{\rho_f}} = \frac{1}{\sqrt{s}\left(k_a\sqrt{s}+b\right)\left(z+\sqrt{\frac{s}{D_f}}\right)} \tag{B.19a}$$

$$\widetilde{\overline{\rho_m}} = \frac{\varphi}{\sqrt{s}\left(k_a\sqrt{s}+b\right)\left(z+\sqrt{\frac{s}{D_m}}\right)} \tag{B.19b}$$

We now invert the original Laplace transform that was taken with respect to t:

$$\overline{\rho_f} = \frac{\sqrt{D_f}}{b}\cdot\frac{Erfcx\left(z\sqrt{D_f\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)}{-\frac{k}{b}\sqrt{D_f}\cdot z+1} \tag{B.20a}$$

$$\overline{\rho_m} = \frac{\varphi\sqrt{D_m}}{b}\cdot\frac{Erfcx\left(z\sqrt{D_m\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)}{-\frac{k}{b}\sqrt{D_m}\cdot z+1} \tag{B.20b}$$

We now factor these functions:

$$\overline{\rho_f} = \frac{\sqrt{D_f}}{b}\cdot\left[\frac{1}{-\frac{k}{b}\sqrt{D_f}\cdot z+1}\right]\cdot\left[Erfcx\left(z\sqrt{D_f\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)\right] \tag{B.21a}$$

$$\overline{\rho_m} = \frac{\varphi\sqrt{D_m}}{b}\cdot\left[\frac{1}{-\frac{k}{b}\sqrt{D_m}\cdot z+1}\right]\cdot\left[Erfcx\left(z\sqrt{D_m\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)\right] \tag{B.21b}$$

We will now use the following cross-correlation property of Laplace transforms:

$$\mathcal{L}^{-1}\left\{F(-s)\cdot G(s)\right\} = \mathcal{L}^{-1}\left\{F\right\}(t) \star \mathcal{L}^{-1}\left\{G\right\}(t) = \int_0^\infty \mathcal{L}^{-1}\left\{F\right\}(\tau)\cdot\mathcal{L}^{-1}\left\{G\right\}(t+\tau)\,d\tau \tag{B.22}$$

Using this property, we can take the inverse Laplace transform of equations (B.21) by taking the cross-correlation of the inverse Laplace transforms of the individual bracketed terms:

$$\rho_f = \frac{\sqrt{D_f}}{b}\cdot\mathcal{L}_z^{-1}\left\{\frac{1}{\frac{k}{b}\sqrt{D_f}\cdot z+1}\right\}\star\mathcal{L}_z^{-1}\left\{Erfcx\left(z\sqrt{D_f\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)\right\} \qquad for\, y'\geq 0 \tag{B.23a}$$

$$\rho_m = \frac{\varphi\sqrt{D_m}}{b}\cdot\mathcal{L}_z^{-1}\left\{\frac{1}{\frac{k}{b}\sqrt{D_m}\cdot z+1}\right\}\star\mathcal{L}_z^{-1}\left\{Erfcx\left(z\sqrt{D_m\cdot t}\right)-Erfcx\left(\frac{b}{k}\sqrt{t}\right)\right\} \qquad for\, y\geq 0 \tag{B.23b}$$

The inverse Laplace transforms in these expressions can be evaluated to get:

$$\rho_f = \frac{\sqrt{D_f}}{b}\cdot\left[\frac{b}{k\sqrt{D_f}}e^{-\frac{b}{k\sqrt{D_f}}y'}\cdot I_{y'\geq 0}\right]\star\left[2\cdot\frac{1}{\sqrt{4\pi D_f t}}e^{-\frac{y'^2}{4D_f t}}\cdot I_{y'\geq 0}\right] \qquad for\, y'\geq 0 \tag{B.24a}$$

$$\rho_m = \frac{\varphi\sqrt{D_m}}{b}\cdot\left[\frac{b}{k\sqrt{D_m}}e^{-\frac{b}{k\sqrt{D_m}}y}\cdot I_{y\geq 0}\right]\star\left[2\cdot\frac{1}{\sqrt{4\pi D_m t}}e^{-\frac{y^2}{4D_m t}}\cdot I_{y\geq 0}\right] \qquad for\, y\geq 0 \tag{B.24b}$$

We now substitute y back in to the fracture mass density function using the relation $y' = -y$ to get that the regional mass densities in normal coordinates:

$$\rho_f = \frac{\sqrt{D_f}}{b}\cdot\left[\frac{b}{k\sqrt{D_f}}e^{\frac{b}{k\sqrt{D_f}}y}\cdot I_{y\leq 0}\right]\star\left[2\cdot\frac{1}{\sqrt{4\pi D_f t}}e^{-\frac{y'^2}{4D_f t}}\cdot I_{y\leq 0}\right] \qquad for\, y\leq 0 \tag{B.25a}$$

$$\rho_m = \frac{\varphi\sqrt{D_m}}{b}\cdot\left[\frac{b}{k\sqrt{D_m}}e^{-\frac{b}{k\sqrt{D_m}}y}\cdot I_{y\geq 0}\right]\star\left[2\cdot\frac{1}{\sqrt{4\pi D_m t}}e^{-\frac{y^2}{4D_m t}}\cdot I_{y\geq 0}\right] \qquad for\, y\geq 0 \tag{B.25b}$$

We have thus found the mass density function for the initial conditions of a point mass starting on a single interface.

Appendix C

In this appendix, we take limit of the system of equations (5.2.1) as $D_w \to \infty$ and formulate the results in the form analogous to that of a single-interface system. We reproduce the equations here for completeness:

$$\frac{\partial \rho_f}{\partial t} = - u\frac{\partial \rho_f}{\partial x} + D_f\frac{\partial^2 \rho_f}{\partial x^2} + D_w\frac{\partial^2 \rho_f}{\partial y^2} \qquad for\ |y| < L \tag{C.1a}$$

$$\frac{\partial \rho_m}{\partial t} = D_m\left(\frac{\partial^2 \rho_m}{\partial x^2} + \frac{\partial^2 \rho_m}{\partial y^2}\right) \qquad for\ |y| > L \tag{C.1b}$$

$$\rho_f = \frac{1}{\varphi}\rho_m \qquad for\ y = \pm L \tag{C.1c}$$

$$\pm k_a\frac{\partial \rho_f}{\partial t} = D_m\frac{\partial \rho_m}{\partial y} - D_w\frac{\partial \rho_f}{\partial y} \qquad for\ y = \pm L \tag{C.1d}$$

We now take the limit of this system as as $D_w \to \infty$. As D_w goes to infinity, we reason that the coefficients involving D_w must immediately go to zero to keep the terms from blowing up:

$$\frac{\partial^2 \rho_f}{\partial y^2} = 0 \qquad for\ |y| < L \tag{C.2a}$$

$$\frac{\partial \rho_f}{\partial y} = 0 \qquad for\ y = \pm L \tag{C.2b}$$

From this, we can conclude that the mass density in the fracture is constant with respect to y:

$$\rho_f(x, y, t) = \rho_f(x, -L, t) for\ |y| \leq L \tag{C.3}$$

By combining this with equation (C.1c), we can express the fracture mass density in terms of the matrix mass density at $y = -L$:

$$\rho_f(x, y, t) = \frac{1}{\varphi}\rho_m(x, y = -L, t) for\ |y| \leq L \tag{C.4}$$

Using equation (C.1c), we can just as easily express the fracture mass density in terms matrix mass density at $y = L$ instead:

$$\rho_f(x, y, t) = \frac{1}{\varphi}\rho_m(x, y = -L, t) for\ |y| \leq L \tag{C.5}$$

By combining the two above equations we can conclude the matrix mass densities at $y = -L$ and $y = L$ are equal:

$$\rho_m(y = L) = \rho_m(y = -L) \tag{C.6}$$

We now turn to another line of reasoning. By equation (C.3), the following is true:

$$2k_a \frac{\partial \rho_f}{\partial t}(y = -L) = k_a \frac{\partial \rho_f}{\partial t}(y = L) + k_a \frac{\partial \rho_f}{\partial t}(y = -L) \tag{C.7}$$

We use equation (C.1d) to expand terms on the right-hand side of the above and get:

$$\begin{aligned} k_a \frac{\partial \rho_f}{\partial t}(y = L) + k_a \frac{\partial \rho_f}{\partial t}(y = -L) = &\left(D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_w \frac{\partial \rho_f}{\partial y}(y = L) \right) \\ &- \left(D_m \frac{\partial \rho_m}{\partial y}(y = -L) - D_w \frac{\partial \rho_f}{\partial y}(y = -L) \right) \end{aligned} \tag{C.8}$$

We now manipulate the fracture terms of right-hand side of the equation until they take the form of an integral over the interior of the fracture:

$$\begin{aligned} k_a \frac{\partial \rho_f}{\partial t}(y = L) + k_a \frac{\partial \rho_f}{\partial t}(y = -L) &= \left(D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_w \frac{\partial \rho_f}{\partial y}(y = L) \right) \\ &\quad - \left(D_m \frac{\partial \rho_m}{\partial y}(y = -L) - D_w \frac{\partial \rho_f}{\partial y}(y = -L) \right) \\ &= \left(D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_m \frac{\partial \rho_m}{\partial y}(y = -L) \right) \\ &\quad - \left(D_w \frac{\partial \rho_f}{\partial y}(y = L) - D_w \frac{\partial \rho_f}{\partial y}(y = -L) \right) \\ &= \left(D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_m \frac{\partial \rho_m}{\partial y}(y = -L) \right) \\ &\quad - \left(D_w \frac{\partial \rho_f}{\partial y}(y = L) - D_w \frac{\partial \rho_f}{\partial y}(y = -L) \right) \\ &= \left(D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_m \frac{\partial \rho_m}{\partial y}(y = -L) \right) \\ &\quad - \int_{-L}^{L} D_w \frac{\partial^2 \rho_f}{\partial y^2}(y = \hat{y})\, d\hat{y} \end{aligned} \tag{C.9}$$

If we take equation (C.1a) and solve for $D_w \frac{\partial^2 \rho_f}{\partial y^2}$, we get that $D_w \frac{\partial^2 \rho_f}{\partial y^2} = \frac{\partial \rho_f}{\partial t} + u \frac{\partial \rho_f}{\partial x} - D_f \frac{y \partial^2 \rho_f}{\partial x^2}$ for $|y| < a$. We substitute this into the integrand above:

$$\begin{aligned} k_a \frac{\partial \rho_f}{\partial t}(y=L) + k_a \frac{\partial \rho_f}{\partial t}(y=-L) = &\left(D_m \frac{\partial \rho_m}{\partial y}(y=L) - D_m \frac{\partial \rho_m}{\partial y}(y=-L) \right) \\ &- \int_{-L}^{L} \left(\frac{\partial \rho_f}{\partial t}(y=\hat{y}) + u \frac{\partial \rho_f}{\partial x}(y=\hat{y}) - D_f \frac{\partial^2 \rho_f}{\partial x^2}(y=\hat{y}) \right) d\hat{y} \end{aligned} \tag{C.10}$$

We use equation (C.3), which expresses that the mass density in the fracture is constant across its width for $t > 0$, to change the integral across the fracture's width into a simple product:

$$\begin{aligned} k_a \frac{\partial \rho_f}{\partial t}(y=L) + k_a \frac{\partial \rho_f}{\partial t}(y=-L) = &\left(D_m \frac{\partial \rho_m}{\partial y}(y=L) - D_m \frac{\partial \rho_m}{\partial y}(y=-L) \right) \\ &- \int_{-L}^{L} \left(\frac{\partial \rho_f}{\partial t}(y=\hat{y}) + u \frac{\partial \rho_f}{\partial x}(y=\hat{y}) - D_f \frac{\partial^2 \rho_f}{\partial x^2}(y=\hat{y}) \right) d\hat{y} \\ = &\left(D_m \frac{\partial \rho_m}{\partial y}(y=L) - D_m \frac{\partial \rho_m}{\partial y}(y=-L) \right) \\ &- \int_{-L}^{L} \left(\frac{\partial \rho_f}{\partial t}(y=-L) + u \frac{\partial \rho_f}{\partial x}(y=-L) - D_f \frac{\partial^2 \rho_f}{\partial x^2}(y=-L) \right) dy \\ = &\left(D_m \frac{\partial \rho_m}{\partial y}(y=L) - D_m \frac{\partial \rho_m}{\partial y}(y=-L) \right) \\ &- 2L \cdot \left(\frac{\partial \rho_f}{\partial t}(y=-L) + u \frac{\partial \rho_f}{\partial x}(y=-L) - D_f \frac{\partial^2 \rho_f}{\partial x^2}(y=-L) \right) \end{aligned} \tag{C.11}$$

We use equation (C.1c) to express the above in terms of the matrix mass density at $y = -L$, as opposed to the fracture mass density. We then collect terms containing $\frac{\partial \rho_m}{\partial t}(y = -L)$ on the left. The result is:

$$\begin{aligned} \frac{2\left(k_a + L\right)}{\varphi} \frac{\partial \rho_m}{\partial t}(y=-L) = &D_m \frac{\partial \rho_m}{\partial y}(y=L) - D_m \frac{\partial \rho_m}{\partial y}(y=-L) \\ &+ \frac{2L \cdot u}{\varphi} \frac{\partial \rho_m}{\partial x}(y=-L) - \frac{2L \cdot D_f}{\varphi} \frac{\partial^2 \rho_m}{\partial x^2}(y=-L) \end{aligned} \tag{C.12}$$

Equations (C.1b), (C.6) and (C.12) together form the following new system of PDEs:

$$\begin{aligned}
\frac{\partial \rho_m}{\partial t} =& D_m \left(\frac{\partial^2 \rho_m}{\partial x^2} + \frac{\partial^2 \rho_m}{\partial y^2} \right) \qquad for\ |y| > L && \text{(C.13a)} \\
\rho_m(y = L) =& \rho_m(y = -L) && \text{(C.13b)} \\
\frac{2\,(k_a + L)}{\varphi} \frac{\partial \rho_m}{\partial t}(y = -L) =& D_m \frac{\partial \rho_m}{\partial y}(y = L) - D_m \frac{\partial \rho_m}{\partial y}(y = -L) \\
& + \frac{2L \cdot u}{\varphi} \frac{\partial \rho_m}{\partial x}(y = -L) - \frac{2L \cdot D_f}{\varphi} \frac{\partial^2 \rho_m}{\partial x^2}(y = -L) && \text{(C.13c)}
\end{aligned}$$

This new upscale system of PDEs is of nearly the same form as the single-interface mass transport equations for a single interface. To make the match more explicit, we define three regional upscaled mass density functions:

$$\begin{aligned}
\rho_{m1}(x, y, t) =& \rho_m(x, y - L, t) && for\ y \leq 0 && \text{(C.14a)} \\
\rho_{m2}(x, y, t) =& \rho_m(x, y + L, t) && for\ y \geq 0 && \text{(C.14b)} \\
\rho_{f/i}(x, y, t) =& \delta_0(y) \cdot \int_{-\infty}^{\infty} \rho_f(x, \hat{y}, t) + \rho_i(x, \hat{y}, t)\, d\hat{y} && for\ y = 0 && \text{(C.14c)}
\end{aligned}$$

We substitute these regional upscaled mass density functions into equations (C.13) to obtain the upscaled mass transport equations:

$$\begin{aligned}
\frac{\partial \rho_{m1}}{\partial t} =& D_m \left(\frac{\partial^2 \rho_{m1}}{\partial x^2} + \frac{\partial^2 \rho_{m1}}{\partial y^2} \right) && for\ y < 0 && \text{(C.15a)} \\
\frac{\partial \rho_{m2}}{\partial t} =& D_m \left(\frac{\partial^2 \rho_{m2}}{\partial x^2} + \frac{\partial^2 \rho_{m2}}{\partial y^2} \right) && for\ y > 0 && \text{(C.15b)} \\
\rho_{m1} =& \rho_{m2} && for\ y = 0 && \text{(C.15c)} \\
\frac{2\,(k_a + L)}{\varphi} \frac{\partial \rho_{m1}}{\partial t} =& D_m \frac{\partial \rho_{m2}}{\partial y} - D_m \frac{\partial \rho_{m1}}{\partial y} + \frac{2L \cdot u}{\varphi} \frac{\partial \rho_m}{\partial x} - \frac{2L \cdot D_f}{\varphi} \frac{\partial^2 \rho_m}{\partial x^2} && for\ y = 0 && \text{(C.15d)}
\end{aligned}$$

This system of equations has a form analogous to that of the single-interface subsystem's, as desired.