Water Resources Research

METHOD
10.1029/2022WR032308

Key Points:

- A decoupled Finite Particle Method with normalized kernel (DFPM-NK) is developed to simulate groundwater solute transport
- DFPM is computationally more efficient than Finite Particle Method (FPM) due to using diagonal elements of correction matrix
- DFPM-NK is computationally more accurate than DFPM due to using normalized kernel

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

M. Ye and M. Jin,
mye@fsu.edu;
mgjin@cug.edu.cn

Citation:

Jiao, T., Ye, M., Jin, M., & Yang, J. (2022). Decoupled finite particle method with normalized kernel (DFPM-NK): A computationally efficient method for simulating solute transport in heterogeneous porous media. *Water Resources Research*, *58*, e2022WR032308. https://doi.org/10.1029/2022WR032308

Received 4 MAR 2022
Accepted 18 JUL 2022

Author Contributions:

Conceptualization: Tian Jiao, Ming Ye, Menggui Jin
Formal analysis: Tian Jiao, Jing Yang
Funding acquisition: Ming Ye, Menggui Jin
Investigation: Tian Jiao, Jing Yang
Methodology: Tian Jiao, Ming Ye, Menggui Jin
Resources: Ming Ye, Menggui Jin
Software: Tian Jiao, Jing Yang
Supervision: Ming Ye, Menggui Jin
Visualization: Tian Jiao, Jing Yang
Writing – original draft: Tian Jiao, Ming Ye, Menggui Jin
Writing – review & editing: Tian Jiao, Ming Ye, Menggui Jin, Jing Yang

Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK): A Computationally Efficient Method for Simulating Solute Transport in Heterogeneous Porous Media

Tian Jiao[1,2], Ming Ye[2], Menggui Jin[1], and Jing Yang[3]

[1]School of Environmental Studies, China University of Geosciences, Wuhan, China, [2]Department of Earth, Ocean, and Atmosphere Science and Department of Scientific Computing, Florida State University, Tallahassee, FL, USA, [3]College of Water Resources and Architectural Engineering, Northwest A&F University, Yangling, China

Abstract Previous studies found that Finite Particle Method (FPM), an improved Smoothed Particle Hydrodynamics (SPH) method, yields more accurate solutions of the advection-dispersion equation (ADE) than SPH method does when simulating solute transport in heterogeneous porous media. FPM however is computationally more expensive than SPH because FPM needs to solve a correction matrix equation for each particle. While decoupled FPM (DFPM) can reduce computational cost of FPM by using diagonal terms of the matrix equation, DFPM is less accurate than FPM due to discarding off-diagonal terms of the matrix equation. This study develops the Decoupled Finite Particle Method with Normalized Kernel (DFPM-NK) to improve computational accuracy of DFPM by using normalized kernels in DFPM. We evaluate computational performance of SPH, FPM, DFPM, and DFPM-NK using two numerical experiments of ADE with non-reactive and reactive solute transport in a heterogeneous aquifer. Results of the two experiments indicate that, among the four methods, SPH has the least computational time, but has the worst computational accuracy. DFPM-NK is substantially more efficient than FPM, and has similar accuracy of FPM. On the other hand, DFPM-NK and DFPM have similar computational cost, but DFMP-NK is significantly more accurate than DFPM, especially for heterogeneous hydraulic conductivity fields. We thus recommend to use DFPM-NK for computationally expensive ADE problems without sacrificing computational accuracy.

1. Introduction

When numerically solving advection-dominated solute transport problems using Lagrangian approaches, smoothed particle hydrodynamics (SPH) methods have been found to often perform better than traditional Eulerian schemes for solving the advection-dispersion equation (ADE) due to SPH's ability of eliminating numerical dispersion (Benson et al., 2017; Benson & Meerschaert, 2008; Bolster et al., 2016; Boso et al., 2013; de Barros et al., 2015; Herrera et al., 2009, 2017; Sole-Mari et al., 2017, 2019; Sole-Mari & Fernàndez-Garcia, 2018; Tartakovsky, Meakin, Scheibe, & Eichler West, 2007; Tartakovsky, Meakin, Scheibe, & Wood 2007; Tartakovsky et al., 2009, 2015). However, standard SPH methods are highly sensitive to spatial distributions of particles (Herrera and Beckie. 2013), and accuracy of SPH methods decreases when solving ADE for heterogeneous aquifers (Avesani et al., 2015; Herrera & Beckie, 2013; Herrera et al., 2009; Tartakovsky, 2010), where particles are irregularly distributed because particles move in non-uniform velocity fields. To resolve this problem, several studies have improved particle approximation of SPH for irregularly distributed particles (Alvarado-Rodríguez et al., 2019; Avesani et al., 2015, 2021; Batra & Zhang, 2004; Liu et al., 2005; Liu & Liu, 2006).

Jiao et al. (2021) used Finite Particle Method (FPM) to improve accuracy of SPH solutions with irregular particle distributions. In a FPM solution of ADE, a Taylor series expansion is used to correct kernel gradients in the construction of concentration gradients for improving particle approximation accuracy. However, since FPM requires solving a matrix equation for each particle, FPM is computationally more expensive than SPH. In addition, when solving the matrix equations for highly irregularly distributed particles, FPM may suffer from numerical instability or early termination due to ill-conditioned or even singular correction matrix.

There have been several studies that improve FPM's computational efficiency and stability in other research fields than groundwater modeling. Zhang and Batra (2009) presented a symmetric smoothed particle hydrodynamics method to model elastic problems, and the method makes the correction matrix inverted symmetric. The symmetric matrix has also been used in other research fields (Jiang et al., 2012; Xu & Deng, 2016). Zhang et al. (2019)

coupled FPM with a particle shifting technique with improved stability to model particulate flows with thermal convection. This particle shifting technique can achieve regular particle distribution by shifting disordered particles to improve FPM stability. However, since the aforementioned methods still need to solve the correction matrix equation, the problems of instability and low computational efficiency have not been adequately resolved. In order to avoid solving the correction matrix equation, a Decoupled Finite Particle Method (DFPM) was developed by Zhang and Liu (2018) to improve FPM stability and computational efficiency for modeling incompressible fows with free surfaces. This method negelects off-diagonal elements of the correction matrix, and only uses diagonal elements to modify kernel functions and/or kernel gradients. Because a full correction matrix is replaced by a diagonal correction matrix, DFPM is numerically more stable and efficient than FPM. However, DFPM is less accurate than FPM because DFPM's kernels and kernel gradients are modified independently using only diagonal elements of the correction matrix. The independent modification of DFPM is undesirable because approximation accuracy of a function affects approximation accuracy of the function's derivatives. In other words, the kernel and kernel gradients should be simultaneously modified so that simulation accuracy of DFPM can be improved.

The simultaneous modification of the kernels and kernel gradients is achieved in this study by integrating normalized kernels with DFPM. While using normalized kernels was originally proposed by Jiang et al. (2012) for corrected SPH to construct a correction matrix, and we use the normalized kernels for DFPM in this study, which leads to the Decoupled Finite Particle Method with Normalized Kernel (DFPM-NK). DFPM-NK first obtains a normalized kernel (Liu & Liu, 2010), and then applies the normalized kernel to DFPM for modifying kernel gradients. DFPM-NK is computationally as efficient as DFPM, but computationally more accurate than DFPM due to using the normalized kernel. In comparison with conventional FPM methods, DFPM-NK is computationally more efficient without sacrificing computational accuracy. This is demonstrated for the first time in this study to solve ADE for simulating solute transport in heterogeneous aquifers. Performance of DFPM-NK is investigated with two numerical experiments involving both non-reactive and reactive transport. With a numerically obtained reference solution, computational cost and efficiency of DFPM-NK are compared with those of SPH, FPM, and DFPM.

2. Mathematic Model and Numerical Schemes

The ADE in porous media with the Lagrangian coordinates is described as

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}, \tag{1}$$

$$\frac{dC}{dt} = \nabla \cdot (\mathbf{D}\nabla C), \tag{2}$$

where C is the solute concentration, $\mathbf{D}$ is the local-scale dispersion coefficient defined by Bear (1972), $\mathbf{v}$ is a vector of seepage velocity, and $\mathbf{x}$ is the position of a fluid particle. The substantial derivative, $dC/dt = \partial C/\partial t + \mathbf{v} \cdot \partial C/\partial \mathbf{x}$, represents time rate change of solute concentration along the pathline of a particle. In a component form, advection and dispersion terms are presented by Equations 1 and 2, respectively.

In SPH scheme, Equation 1 governs the particle movement, and the solution of Equation 2 is evaluated by using a kernel interpolation approximation. The kernel interpolation approximation of dispersion term in Equation 2 for particle i is given as (Alvarado-Rodríguez et al., 2019; Cleary & Monaghan, 1999; Herrera et al., 2009)

$$\frac{dC_i}{dt} = \frac{1}{2}\sum_{j=1}^{N_b} \frac{1}{n_{ij}} (C_i - C_j) D_{ij} \frac{\mathbf{x}_i - \mathbf{x}_j}{|\mathbf{x}_i - \mathbf{x}_j|^2} \cdot \nabla_i W_{ij}, \tag{3}$$

where $C_i = C(\mathbf{x}_i)$ and $C_j = C(\mathbf{x}_j)$ are the concentrations of the particles i and j, respectively, W is a kernel function, N_b is the number of neighbor particles within a the ith particle's with smoothing length h. Particle number density (i.e., the number of particles per unit volume), $n_j = \rho_j/m_j = 1/V_j$, is expressed as $n_j = \sum_k W(\mathbf{x}_j - \mathbf{x}_k, h)$ (Tartakovsky & Meakin, 2005), where $\mathbf{x}_k$ is the location of particle k in the neighborhood of particle j. The particle number density (and the associated hydrodynamic volume, V_j) may vary, depending on the spatial distribution and the number of particles within the simulation domain. The $n_{ij} = 2n_i n_j/(n_i + n_j)$ term in Equation 3 is the harmonic mean of n_i and n_j that are particle density number at position $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. D_{ij}

is set as $D_{ij} = 2\left(D^i + D^j\right)$ for isotropic dispersion problem (Herrera et al., 2009; Jubelgas et al., 2004) and $D_{ij} = \sum_\alpha \sum_\beta \left(D^i_{\alpha\beta} + D^j_{\alpha\beta}\right)\left(\Gamma\left(\mathbf{x}_i - \mathbf{x}_j\right)_\alpha \left(\mathbf{x}_i - \mathbf{x}_j\right)_\beta / |\mathbf{x}_i - \mathbf{x}_j|^2 - \delta_{\alpha\beta}\right)$ for anisotropic dispersion problem, where $\Gamma = 4$ and 5 in a two- and three-dimensional space, respectively (Alvarado-Rodríguez et al., 2019; Espanol & Revenga, 2003; Yildiz et al., 2009), α and β are the dimension indices ranging from 1 to 3 (i.e., from x to z), $D^i_{\alpha\beta}$ is a component of the dispersion coefficient tensor related to particle i, and $\delta_{\alpha\beta}$ is the Kronecker delta function. This study is focused on isotropic dispersion problems.

The fundamentals of SPH methods are given in Liu and Liu (2010), and various applications of the SPH methods in other research fields than groundwater modeling are referred to Amicarelli et al. (2020). For an irregular particle distribution, the low accuracy of SPH solution is related to the normalized and symmetric properties of kernel function (i.e., $\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}\left(x_i - x_j, h\right) = 1$, $\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}\left(x_i - x_j, h\right)\left(x_i - x_j\right) = 0$) and anti-symmetric property of kernel gradients (i.e., $\sum_{j=1}^{N_b} \frac{m_j}{\rho_j}\left(\mathbf{x}_j - \mathbf{x}_i\right)\partial_{\mathbf{x}_i} W_{ij}\left(\mathbf{x}_i - \mathbf{x}_j, h\right) = 1$ or 0). These properties are not satisfied when particles are irregularly distributed, or support domain is not completed as illustrated in Jiao et al. (2021). This will affect the SPH particle approximation accuracy of C_i and its first-order derivatives, and accuracy of Equation 3 depends on particle distributions. This was demonstrated in Jiao et al. (2021), who also showed that FPM solutions of ADE are more accurate than SPH solutions for irregular particle distributions.

2.1. Finite Particle Method (FPM) and Decoupled FPM (DFPM)

The derivation of FPM starts with first applying the Taylor series expansion to $C_j = C(\mathbf{x}_j)$ at particle location $\mathbf{x}_i = \{x_i, y_i, z_i\}$, and then retaining the first-order derivatives, which leads to

$$C_j \approx C_i + \left(\mathbf{x}_j^\alpha - \mathbf{x}_i^\alpha\right)\frac{\partial C_i}{\partial \mathbf{x}_i^\alpha} = C_i + \left(x_j - x_i\right)\frac{\partial C_i}{\partial x_i} + \left(y_j - y_i\right)\frac{\partial C_i}{\partial y_i} + \left(z_j - z_i\right)\frac{\partial C_i}{\partial z_i}, \tag{4}$$

In FPM, the both sides of Equation 4 are multiplied by the kernel function W_{ij} and its three first-order derivatives, $\partial W_{ij}/\partial x_i$, $\partial W_{ij}/\partial y_i$ and $\partial W_{ij}/\partial z_i$, and the results are integrated over the support domain. Subsequently, replacing the integrations with the particle approximations leads to

$$\begin{cases} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j W_{ij} = C_i \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} + \frac{\partial C_i}{\partial x_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} + \frac{\partial C_i}{\partial y_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} + \frac{\partial C_i}{\partial z_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} W_{ij} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial x_i} = C_i \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i} + \frac{\partial C_i}{\partial x_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial x_i} + \frac{\partial C_i}{\partial y_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial x_i} + \frac{\partial C_i}{\partial z_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial W_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial y_i} = C_i \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i} + \frac{\partial C_i}{\partial x_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial y_i} + \frac{\partial C_i}{\partial y_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial y_i} + \frac{\partial C_i}{\partial z_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial W_{ij}}{\partial y_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial z_i} = C_i \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial z_i} + \frac{\partial C_i}{\partial x_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial z_i} + \frac{\partial C_i}{\partial y_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial z_i} + \frac{\partial C_i}{\partial z_i} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial W_{ij}}{\partial z_i}, \end{cases} \tag{5}$$

where $x_{ji} = x_j - x_i$, $y_{ji} = y_j - y_i$ and $z_{ji} = z_j - z_i$. In Equation 5, C_i, $\frac{\partial C_i}{\partial x_i}$, $\frac{\partial C_i}{\partial y_i}$, and $\frac{\partial C_i}{\partial z_i}$ are the four unknows for this system of equations, and the four unknows are simultaneously obtained by solving Equation 5. Expressions of particle approximations the C_i and its three first-order derivatives in a matrix form are given by

$$\begin{bmatrix} C_i \\ \frac{\partial C_i}{\partial x_i} \\ \frac{\partial C_i}{\partial y_i} \\ \frac{\partial C_i}{\partial z_i} \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j W_{ij} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial y_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} C_j \frac{\partial W_{ij}}{\partial z_i} \end{bmatrix}, \tag{6}$$

Where

$$\mathbf{M}=\begin{bmatrix} \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}W_{ij} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}W_{ij} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}z_{ji}W_{ij} \\ \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}\frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}\frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}z_{ji}\frac{\partial W_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}\frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}\frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}z_{ji}\frac{\partial W_{ij}}{\partial y_i} \\ \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial z_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}\frac{\partial W_{ij}}{\partial z_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}\frac{\partial W_{ij}}{\partial z_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}z_{ji}\frac{\partial W_{ij}}{\partial z_i} \end{bmatrix}, \tag{7}$$

By solving the matrix equation of Equation 6 and using modified particle approximations of concentration and its first-order derivatives (i.e., $C_i=\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}C_jW_{ij}^M$, and $\frac{\partial C_i}{\partial \mathbf{x}_i}=\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}C_j\frac{\partial^M W_{ij}}{\partial \mathbf{x}_i}$), the modified kernel W_{ij}^M and its first-order derivatives are given by

$$\begin{bmatrix} W_{ij}^M \\ \frac{\partial^M W_{ij}}{\partial x_i} \\ \frac{\partial^M W_{ij}}{\partial y_i} \\ \frac{\partial^M W_{ij}}{\partial z_i} \end{bmatrix}=\mathbf{M}^{-1}\begin{bmatrix} W_{ij} \\ \frac{\partial W_{ij}}{\partial x_i} \\ \frac{\partial W_{ij}}{\partial y_i} \\ \frac{\partial W_{ij}}{\partial z_i} \end{bmatrix}, \tag{8}$$

With this modified kernel and kernel gradients, in FPM, the concentration and its gradient are approximated by solving matrix equation, $\mathbf{M}\begin{bmatrix} W_{ij}^M & \frac{\partial^M W_{ij}}{\partial x_i} & \frac{\partial^M W_{ij}}{\partial y_i} & \frac{\partial^M W_{ij}}{\partial z_i} \end{bmatrix}^T=\begin{bmatrix} W_{ij} & \frac{\partial W_{ij}}{\partial x_i} & \frac{\partial W_{ij}}{\partial y_i} & \frac{\partial W_{ij}}{\partial z_i} \end{bmatrix}^T$. In the FPM solution of the ADE, we only need to replace kernel gradient $\nabla_i W_{ij}$ in Equation 3 with the modified kernel gradient, $\nabla_i^M W_{ij}$, modified by using matrix $\mathbf{M}$. When $\mathbf{M}$ is a unity matrix, the FPM implementation is identical to the SPH implementation. It should be noted that, while solute mass is conserved in SPH because of $\nabla_i W_{ij}=-\nabla_j W_{ji}$ in SPH solutions (Herrera et el. 2009; Avesani et al., 2015; Boso et al., 2013), solute mass is not conserved in FPM, DFPM, and DFPM-NK, because the modified kernel gradients $\nabla_i W_{ij}^M$ and $\nabla_j W_{ji}^M$ at particles i and j, respectively, are modified with different correction matrix $\mathbf{M}$, that is, $\nabla_i W_{ij}^M \neq -\nabla_j W_{ji}^M$. This is illustrated in Figure S1 in Supporting Information S1 file associated with the paper.

FPM has two computational problems. One is that FPM is computationally more expensive than SPH, because FPM needs to solve pointwise matrix equations at every time step when particles move with groundwater flow. The other problem is that FPM may suffer from numerical instability and even early termination due to ill-conditioned or singular correction matrix $\mathbf{M}$ for cases with highly irregular particles. It is thus necessary to improve computational efficiency and stability of FPM. This can be done by using DFPM (Zhang & Liu, 2018), and Jiao et al. (2021) applied DFPM to groundwater solute transport modeling. In DFPM, the correction matrix of FPM is replaced by a diagonal matrix $\overline{\mathbf{M}}$ that only contains the diagonal elements of $\mathbf{M}$ in Equation 7 to improve stability and computational efficiency of FPM solution. Hence, the modified kernel and kernel gradient are obtained as

$$\begin{cases} W_{ij}^{\overline{M}} = \dfrac{W_{ij}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} W_{ij}} \\ \dfrac{\partial^{\overline{M}} W_{ij}}{\partial x_i} = \dfrac{\dfrac{\partial W_{ij}}{\partial x_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \dfrac{\partial W_{ij}}{\partial x_i}} \\ \dfrac{\partial^{\overline{M}} W_{ij}}{\partial y_i}_{ij} = \dfrac{\dfrac{\partial W_{ij}}{\partial y_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \dfrac{\partial W_{ij}}{\partial y_i}} \\ \dfrac{\partial^{\overline{M}} W_{ij}}{\partial z_i} = \dfrac{\dfrac{\partial W_{ij}}{\partial z_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \dfrac{\partial W_{ij}}{\partial z_i}} \end{cases}. \tag{9}$$

Equation 9 of DFPM indicates that the kernel function and its gradient are modified independently in that only diagonal elements of the correction matrix are used. This is clearly different from Equation 8 of FPM, which indicates that the kernel function and its gradient of FPM are modified by using all elements of the correction matrix. This is demonstrated in the appendix for a two-dimensional problem for which analytical expressions of the modified kernel gradient are derived. Equations A6 and A7 show that, in FPM, the modified kernel gradients are a function of kernel. Therefore, without using the kernel to evaluate the modified kernel gradient as in Equation 9, numerical results of DFPM are less accurate than those of FPM, and the numerical error may be significant.

2.2. Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK)

We note that the modified kernel gradient of FPM can be expressed by using the so-called normalized kernel that is defined below, as shown in Equations A6 and A7 for the two-dimensional problem. It is therefore possible to use the kernel function to modify kernel gradient even in a decoupled manner. This is the motivation of developing DFPM-NK.

We first develop FPM-NK, and then simplify it to DFPM-NK. If we normalize the kernel function in first row of Equation 9 as $\hat{W}_{ij} = W_{ij}^{\overline{M}} = W_{ij} / \sum_{j=1}^{N_b} \left(\frac{m_j}{\rho_j} W_{ij} \right)$ (Jiang et al., 2012), then the kernel gradient of normalized kernel function is derived as

$$\frac{\partial \hat{W}_{ij}}{\partial \mathbf{x}_i} = \frac{\frac{\partial W_{ij}}{\partial \mathbf{x}_i} \eta_i - W_{ij} \frac{\partial \eta_i}{\partial \mathbf{x}_i}}{\eta_i^2} = \frac{\frac{\partial W_{ij}}{\partial \mathbf{x}_i} \eta_i - W_{ij} \sum\limits_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial \mathbf{x}_i}}{\eta_i^2}, \tag{10}$$

where $\eta_i = \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}$. Similar to the derivation of FPM, we multiply the both sides of the Taylor series expansion in Equation 4 by the three first-order derivatives of the normalized kernel, $\partial \hat{W}_{ij}/\partial x_i$, $\partial \hat{W}_{ij}/\partial y_i$ and $\partial \hat{W}_{ij}/\partial z_i$, and subsequently integrate the results over the support domain. Replacing the integrations with the particle approximations and moving the first term on the right-hand side to the left-hand side gives the three-dimensional system equations as

$$\begin{cases} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} (C_j - C_i) \dfrac{\partial \hat{W}_{ij}}{\partial x_i} = \dfrac{\partial C_i}{\partial x_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial x_i} + \dfrac{\partial C_i}{\partial y_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial x_i} + \dfrac{\partial C_i}{\partial z_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial x_i} \\ \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} (C_j - C_i) \dfrac{\partial \hat{W}_{ij}}{\partial y_i} = \dfrac{\partial C_i}{\partial x_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial y_i} + \dfrac{\partial C_i}{\partial y_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial y_i} + \dfrac{\partial C_i}{\partial z_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial y_i} \\ \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} (C_j - C_i) \dfrac{\partial \hat{W}_{ij}}{\partial z_i} = \dfrac{\partial C_i}{\partial x_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial z_i} + \dfrac{\partial C_i}{\partial y_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial z_i} + \dfrac{\partial C_i}{\partial z_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial z_i} \end{cases}, \tag{11}$$

Rewriting Equation 11 into the matrix form, we have the particle approximations as

$$\begin{bmatrix} \frac{\partial C_i}{\partial x_i} \\ \frac{\partial C_i}{\partial y_i} \\ \frac{\partial C_i}{\partial z_i} \end{bmatrix} = \left(\mathbf{M}^N\right)^{-1} \begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} (C_j - C_i) \frac{\partial \hat{W}_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} (C_j - C_i) \frac{\partial \hat{W}_{ij}}{\partial y_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} (C_j - C_i) \frac{\partial \hat{W}_{ij}}{\partial z_i} \end{bmatrix}, \tag{12}$$

where

$$\mathbf{M}^N = \begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial \hat{W}_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial \hat{W}_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial \hat{W}_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial \hat{W}_{ij}}{\partial y_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial \hat{W}_{ij}}{\partial y_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial \hat{W}_{ij}}{\partial y_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial \hat{W}_{ij}}{\partial z_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial \hat{W}_{ij}}{\partial z_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} z_{ji} \frac{\partial \hat{W}_{ij}}{\partial z_i} \end{bmatrix}, \tag{13}$$

Using the asymmetric form of the first-order derivatives of C_i, $\frac{\partial C_i}{\partial \mathbf{x}_i} = \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} (C_j - C_i) \frac{\partial^M W_{ij}}{\partial \mathbf{x}_i}$, we derive the modified kernel gradients, that is,

$$\begin{bmatrix} \frac{\partial^M W_{ij}}{\partial x_i} \\ \frac{\partial^M W_{ij}}{\partial y_i} \\ \frac{\partial^M W_{ij}}{\partial z_i} \end{bmatrix} = \left(\mathbf{M}^N\right)^{-1} \begin{bmatrix} \frac{\partial \hat{W}_{ij}}{\partial x_i} \\ \frac{\partial \hat{W}_{ij}}{\partial y_i} \\ \frac{\partial \hat{W}_{ij}}{\partial z_i} \end{bmatrix}, \tag{14}$$

With normalized kernel function used in Equation 14, the particle approximation of modified kernel gradient in Equation 14 is as accurate as the modified kernel gradients in Equation 8 of FPM solution. This is demonstrated in the appendix for the two-dimensional problem that Equations A8 and A9 based on the normalized kernel are the same as Equation A6 and A7 based on the original kernel.

DFPM-NK is derived by simply neglecting the off-diagonal terms of the correction matrix $\mathbf{M}^N$, and the diagonal matrix $\overline{\mathbf{M}}^N$ is

$$\overline{\mathbf{M}}^N = \begin{bmatrix} \sum_j \frac{m_j}{\rho_j} x_{ji} \frac{\partial \hat{W}_{ij}}{\partial x_i} & 0 & 0 \\ 0 & \sum_j \frac{m_j}{\rho_j} y_{ji} \frac{\partial \hat{W}_{ij}}{\partial y_i} & 0 \\ 0 & 0 & \sum_j \frac{m_j}{\rho_j} z_{ji} \frac{\partial \hat{W}_{ij}}{\partial z_i} \end{bmatrix}, \tag{15}$$

The first-order derivatives of the kernel function are obtained by replacing W_{ij} in the right-hand side of the last three rows of Equation 9 with $W_{ij}^{\overline{M}}\left(\hat{W}_{ij}\right)$

$$\begin{cases} \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial x_i} = \dfrac{\dfrac{\partial \hat{W}_{ij}}{\partial x_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial x_i}} \\ \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial y_i} = \dfrac{\dfrac{\partial \hat{W}_{ij}}{\partial y_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial y_i}} \\ \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial z_i} = \dfrac{\dfrac{\partial \hat{W}_{ij}}{\partial z_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \dfrac{\partial \hat{W}_{ij}}{\partial z_i}} \end{cases}, \tag{16}$$

Substituting the first-order derivatives of the normalized kernel $\frac{\partial \hat{W}_{ij}}{\partial \mathbf{x}_i}$ of Equations 10 to 16, the modified kernel gradients are given as

$$\begin{cases} \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial x_i} = \dfrac{\dfrac{\partial W_{ij}}{\partial x_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial x_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} x_{ji} \left(\dfrac{\partial W_{ij}}{\partial x_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial x_i} \right)} \\ \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial y_i} = \dfrac{\dfrac{\partial W_{ij}}{\partial y_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial y_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} y_{ji} \left(\dfrac{\partial W_{ij}}{\partial y_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial y_i} \right)} \\ \dfrac{\partial^{\overline{M}^N} W_{ij}}{\partial z_i} = \dfrac{\dfrac{\partial W_{ij}}{\partial z_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial z_i}}{\sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} z_{ji} \left(\dfrac{\partial W_{ij}}{\partial z_i} - \dfrac{W_{ij}}{\eta_i} \sum\limits_{j=1}^{N_b} \dfrac{m_j}{\rho_j} \dfrac{\partial W_{ij}}{\partial z_i} \right)} \end{cases}, \tag{17}$$

Comparing Equation 17 of DFPM-NK with Equation 9 of DFPM indicates that the modified kernel gradient of DFPM-NK is a function of the kernel. DFPM-NK thus improves computational accuracy of DFPM, and still has computational efficiency of DFPM.

For solving ADE, replacing kernel gradient $\nabla_i W_{ij}$ in Equation 3 with modified kernel gradient $\nabla_i^{\overline{M}^N} W_{ij}$ based on matrix $\overline{\mathbf{M}}^N$ gives the DFPM-NK solution of the ADE as

$$\frac{dC_i}{dt} = \frac{1}{2} \sum_{j=1}^{N_b} \frac{1}{n_{ij}} (C_j - C_i) D_{ij} \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|^2} \cdot \nabla_i^{\overline{M}^N} W_{ij}, \tag{18}$$

The only difference between Equations 3 and 18 is the use of the modified kernel gradient corrected by using diagonal matrix $\overline{\mathbf{M}}^N$. This is an advantage of DFPM-NK, because the DFPM-NK solution can be implemented in a straightforward way based on the SPH and FPM implementations without significantly changing computer codes of SPH and FPM. The problem of mass conservation in FPM solutions also occurs in DFPM and DFPM-NK solutions due to the correction of kernel gradients.

3. Numerical Experiment Setup

Computational performance of DFPM-NK is evaluated by using two numerical experiments of ADE-based groundwater solute transport modeling with conservative and reactive solutes. The two experiments consider solute transport in a two-dimensional, confined domain with the size of 40 × 15 m. It is assumed that the natural logarithm of hydraulic conductivity, ln$K$, follows the normal distribution with mean of $<\ln K>=1$ and variance

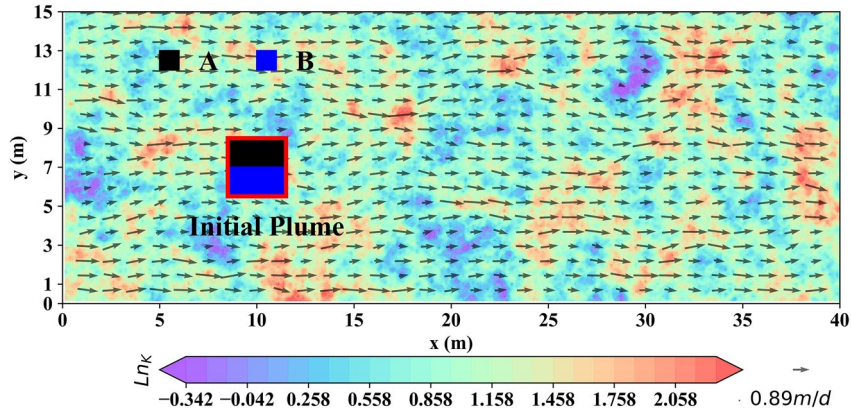

Figure 1. Groundwater seepage velocity for a heterogeneous field of hydraulic conductivity with $\sigma^2_{\ln K} = 0.2$. The red box represents the area of initial concentration domain for the transport problem of conservative solute. The black and blue areas represent the initial concentration domain of species A and B, respectively, for the transport problem of reactive solute.

of $\sigma^2_{\ln K} = 0.2$. An exponential covariance function with the correlation length of $I_{\ln K} = 1$m is considered. A field heterogeneous hydraulic conductivity (Figure 1) is generated by using the "gstat" geostatistical package (Pebesma, 2004). The top and bottom boundaries are set as no-flow boundaries and the two lateral boundaries are assumed to be constant head boundaries, producing steady flow from left to right. MODFLOW 2000 (Harbaugh, 2000) is used to solve the flow problem with a uniform block size of 0.125 × 0.125 m. Seepage velocity is evaluated by using a constant porosity of 0.3 over the simulation domain. The velocity field is shown in Figure 1, and used to simulate solute transport.

3.1. Transport of Conservative Solute

The first numerical experiment considers a conservative solute transport problem. The initial condition of the advection-dispersion problem has an instantaneous concentration $C_0 = 1$ mol/m^3 that is a constant over a square with the size of $L = 3$m centered at the location of (10m, 7m). The dispersion coefficient is isotropic with $D = D_{xx} = D_{yy} = 1 \times 10^{-6}$ m^2/s. The total simulation time is set as $t = 1{,}200\Delta t$ with a uniform time step of $\Delta t = 1{,}800$ s. This time step satisfies the relation of $\Delta t \le \varepsilon h^2 / (D_{xx} + D_{yy})$ that has been used in literature (Alvarado-Rodríguez et al., 2019; Herrera & Beckie, 2013; Herrera et al., 2009), where ε is an empirical coefficient used in this study. Previous studies reported that $\varepsilon = 0.1$ is a good choice for SPH solutions (Alvarado-Rodríguez et al., 2019; Herrera & Beckie, 2013; Herrera et al., 2009). This study uses a smaller time step considering the highly irregular particle distribution used in the two numerical experiments, because using a smaller time step helps capture spatial variation of solute concentrations for the irregular particle distribution. To be consistent, for all particle irregular distributions in the advection dispersion transport experiment, we use the same ε value of 0.02. Initial particles at time $t = 0$ are evenly placed over the domain with the spatial resolution of $\Delta x = \Delta y = 0.125$ m, and the kernel smoothing length is set as $h = 1.5\Delta x = 0.1875$ m. When the particles start moving with groundwater flow, the particle distribution becomes irregular. To estimate the irregularity, the standard deviation of unity index $Q_i = \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}$ (Jiao et al., 2021; Zhu et al., 2015) is used in this study. While the value of Q_i varies between particles, for regularly distributed particles and accurately estimated particle number density, the histogram of Q_i for all the particles has a peak around 1 (Zhu et al., 2015). When the particle distribution becomes more irregular and particle number becomes less, Q_i deviates further from 1, and the standard deviation of Q_i increases. After Q_i is evaluated for all the particles, the standard deviation of Q_i is estimated and used to quantify the degree of particle irregularity. A larger standard deviation indicates a higher degree of particle irregularity.

3.2. Transport of Reactive Solutes

The second numerical experiment is the same as the first one, except that the former experiment considers two chemical species undergoing an irreversible bimolecular reaction of type A + B → C. The transport is described by ADE with a degradation term, and the initial solute distribution is divided into two halves, each containing one reactant (A or B) of the concentration of 1 mol/m^3, that is, $C_A = C_B = 1$ mol/m^3. Initially, the concentration

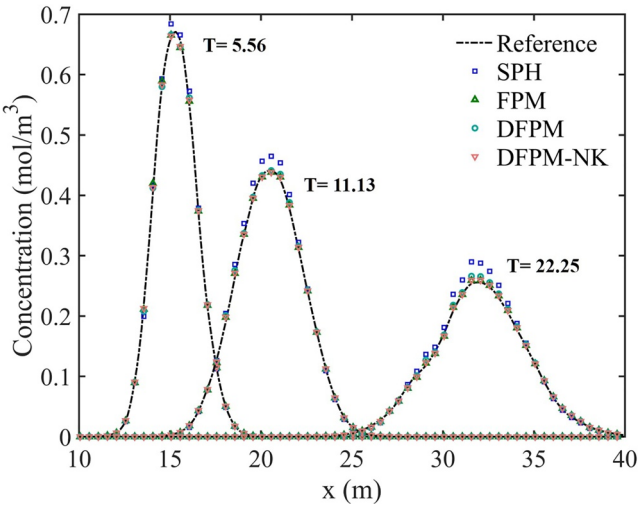

Figure 2. Profiles of concentration of reference, Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM), and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) solutions along the line of $y = 7.5$ m at different dimensionless time $T = Ut/I_{\ln K} = 5.56$, 11.13, and 22.25 for $\sigma^2_{\ln K} = 0.2$.

of product C is 0, and the distributions of reactants A and B are shown in Figure 1. The figure shows that the initial concentration distributions of species A and B are not mixed, and the corresponding simulation results are discussed below. Figure S2 in Supporting Information S1 includes simulation results of two additional conditions of partial and total mixing for the initial distributions of species A and B. The reaction rate of the irreversible reaction follows a double Monod kinetic model with limited conditions caused by simultaneously high concentrations of A and B,

$$r = k_f \frac{C_A}{C_A + K_A} \frac{C_B}{C_B + K_B}, \tag{19}$$

where the kinetic constants are set as $k_f = 7.2 \times 10^{-3}$ mol·m^{-3}·h^{-1}, $K_A = 1.67$ mol/m^3, and $K_B = 0.016$ mol/m^3. The kinetic constant values are taken from Sole-Mari et al. (2017).

3.3. Reference Solutions

To evaluate computational performance of SPH, FPM, DFPM and DFPM-NK, a reference solution is obtained numerically, because there is no analytical solution for solute transport in heterogeneous porous media. The reference solution used in this study is given by FPM using a fine particle spacing of $\Delta x = \Delta y = 0.03125$ m and an adequate smoothing length of $h = 0.3$ m. These values are chosen based on the general principles that accurate SPH solutions can be obtained by using an adequate value of smoothing length, h, to ensure a large number of particles within a support domain (Herrera et al., 2009; Zhu et al., 2015). On the other hand, Jiao et al. (2021) showed that FPM solutions are more accurate than SPH solutions for irregular particle distributions. With the reference solution, the error of scaled L_2 norm error is used to quantify the error between the reference solution and the numerical solutions of SPH, FPM, DFPM and DFPM-NK. This error of L_2 norm scaled by particle number is defined as error $= \|C_N - C_R\|_2/N$, where C_N is a numerical solution, C_R is the reference solution, and N is the total number of particles. Since the particle spacings used for the four numerical solutions are different from the particle spacing used for the reference solution, to compare the SPH, FPM, DFPM and DFPM-NK solutions with the reference solution, the numerical and reference solutions are interpolated to a grid with a uniform particle spacing. The interpolation is conducted by using the MATLAB scatter interpolation function, *ScatteredInterpolant*, with the "*natural*" option of the function to facilitate the comparison between the numerical solutions and the reference solution. We assume that the interpolation error is not expected to affect the conclusions of this study.

4. Results

4.1. Transport of Conservative Solution

The numerical results plotted in Figures 2 and 3 are used to evaluate computational accuracy of SPH, FPM, DFPM, and DFPM-NK. Figure 2 plots the concentration profiles of the reference and the four numerical solutions along the line of $y = 7.5$ m at three dimensionless times of $T = Ut/I\ln k = 5.56$, 11.13, and 22.25 for $\sigma^2_{\ln K} = 0.2$, where $U = 0.89$ m/day is the mean seepage velocity in the x direction. Figure 2 shows that the SPH solution deviates from the reference solution when simulation time increases. This is also the case for the DFPM solution, but the DFPM solution is more accurate than the SPH solution. The FPM solution is closer to the reference solution than the SPH and DFPM solutions, and visually speaking, the DFPM-NK and FPM solutions have similar accuracy.

To investigate SPH, FPM, DFPM, and DFPM-NK for different levels of particle irregularity in different fields of heterogeneous hydraulic conductivity, Figure 3 plots the error of the four methods over the simulation domain for three $\sigma^2_{\ln K}$ values of 0.2, 0.4, and 0.8. Figure 3a shows that, as expected, the standard deviation of Q_i increases when $\sigma^2_{\ln K}$ increases, and that the standard deviation increases with time for each $\sigma^2_{\ln K}$ value. The particle irregularity for $\sigma^2_{\ln K} = 0.8$ is already large enough. As shown in Figure S3 in Supporting Information S1, the standard deviation of Q_i for $\sigma^2_{\ln K} = 0.8$ is similar to that for $\sigma^2_{\ln K} = 2.0$, 4.0, and 8.0 at late simulation time. Figures 3b–3d

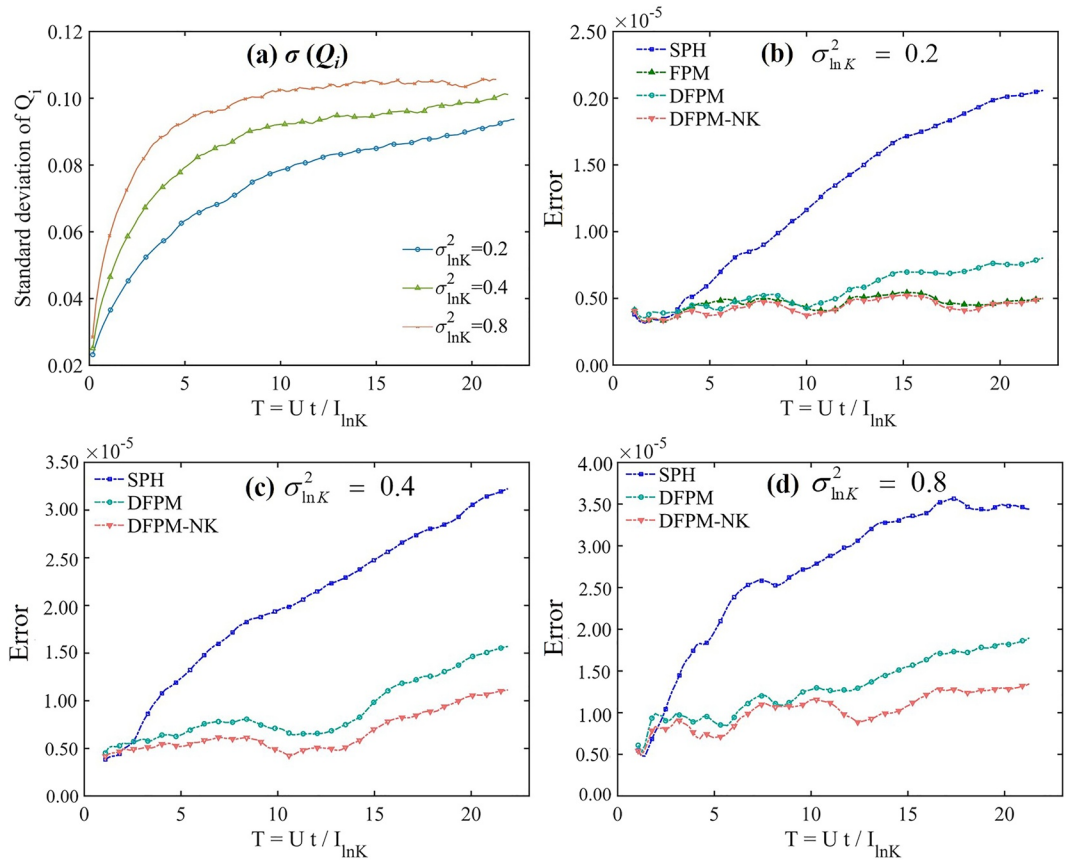

Figure 3. (a) Variation of the standard deviation of unity index, Q_i, as a function of dimensionless time for three heterogeneous fields of hydraulic conductivity with different $\sigma^2_{\ln K}$ values. The integral scale, $I_{\ln K}$, of heterogeneous hydraulic conductivity field is set as $I_{\ln K} = 8\Delta x = 1$m. (b–d) Variation of error of concentration of Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM), and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) solutions, as a function of dimensionless time for the three heterogeneous fields of hydraulic conductivity.

indicate that the SPH results have the least accuracy. Figure 3b indicates that DFPM is more accurate than SPH but less accurate than FPM and DFPM-NK especially for late simulation time.

For Figures 3c and 3d with $\sigma^2_{\ln K} = 0.4$ and $\sigma^2_{\ln K} = 0.8$, respectively, they only plot the results of SPH, DFPM, and DFPM-NK, because FPM fails for the two $\sigma^2_{\ln K}$ values due to highly irregular particle distribution. While FPM results may be obtained by adjusting Δx, h, and Δt values, the failure of FPM demonstrates numerical instability of FPM and the better stability of DFPM and DFPM-NK. On the other hand, Figures 3c and 3d indicate that DFPM-NK significantly outperforms DFPM in terms of computational accuracy for more heterogeneous fields of hydraulic conductivity and later simulation times. This is not surprising because the two factors control the particle irregularity in heterogenous hydraulic conductivity. The supporting information file compares accuracy of SPH, FPM, DFPM, and DFPM-NK for a numerical example of diffusion transport in a field with homogeneous hydraulic conductivity. For this example, an analytical solution of the ADE exists, and it is used as the reference solution. Figure S4 in Supporting Information S1 plots the error for SPH, FPM, DFPM, and DFPM-NK solutions over the simulation domain. The figure also shows that FPM and DFPM-NK visually have the same computational accuracy, and they are substantially more accurate than DPFM, which is substantially more accurate than SPH.

Figure 4 plots the relative L_2 norm errors of concentration (i.e., $\|C_N - C_R\|_2/\|C_R\|_2$) of SPH, FPM, DFPM, and DFPM-NK as a function of the initial particle spacing Δx and CPU time (seconds) for $\sigma^2_{\ln K} = 0.2$. Figure 4a shows that for all the Δx values, the SPH solutions are least accurate, followed by the DFPM solutions. The results indicate that FPM and DFPM-NK have similar accuracy. Figure 4b indicates that, generally speaking, DFPM-NK has the highest computational efficiency. These results demonstrate that the computational efficiency of DFPM-NK

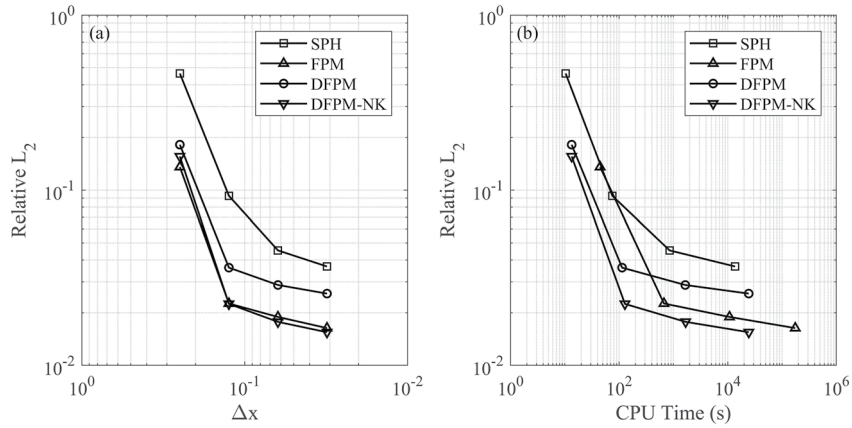

Figure 4. Relative L_2 norm errors of concentration of the Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM), and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) solutions over the simulation domain as a function of (a) particle spacing (Δx) and (b) CPU time (seconds) in a heterogeneous field of hydraulic conductivity with $\sigma^2_{\ln K} = 0.2$.

is similar to (slightly higher than) that of DFPM, and that the computational accuracy of DFPM-NK is similar to (slightly less accurate than) that of FPM. The computational efficiency of DFPM-NK is attributed to using the diagonal matrix in Equation 15, which avoids solving the correction matrix and thus improves the computational efficiency. The computational accuracy of DFPM-NK is due to using the modified kernel gradient in Equation 17 to simultaneously modify both the kernels and the kernel gradients.

Table 1 lists the error and CPU time of the four numerical solutions with four Δx values of 0.25, 0.125, 0.0625, and 0.03125 m. The CPU times are measured on a computer with Intel(R) Core (TM) i5-10400 CPU of 2.90 GHz. When Δx reduces from 0.25 to 0.03125 m, the computational time of FPM becomes substantially larger than that of SPH, and the ratio of computation times of FPM and SPH increases from 4.31 (44.61s/10.35s) to 12.81 (174,962.36s/13,656.85s). Both DFPM and DFPM-NK are computationally more efficient than FPM, and the efficiency increases when particle spacing becomes finer. In comparison with FPM, DFPM-NK saves 70% ((44.61–13.38)/44.61) CPU time for $\Delta x = 0.25$ m and 86% ((174,962.36–24,409.04)/174,962.36) for $\Delta x = 0.03125$ m. The reason is that, when the particle spacing decreases, the particle number increases, and more correction matrices need to be constructed for each particle as indicated by Equation 8. Figure 4 and Table 1 show that, while DFPM is slightly more efficient than DFPM-NK, DFPM-NK is slightly more accurate than DFPM,

Table 1
Relative L_2 Norm Error and CPU Time (Second) of Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM), and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) Solutions With Four Different Δx Values for $\sigma^2_{\ln K} = 0.2$

	0.25		0.125		0.0625		0.03125	
	Error	Time	Error	Time	Error	Time	Error	Time
Δx (m)	Non-reactive transport							
SPH	0.46	10.35	0.09	74.99	0.05	851.26	0.04	13656.85
FPM	0.14	44.61	0.02	669.78	0.02	10761.49	0.02	174962.36
DFPM	0.18	13.36	0.04	113.42	0.03	1651.69	0.03	24327.13
DFPM-NK	0.16	13.38	0.02	127.88	0.02	1678.11	0.02	24409.04
	Reactive transport							
SPH	0.40	11.81	0.09	82.11	0.05	1014.99	0.04	14850.42
FPM	0.25	49.01	0.03	699.45	0.02	11930.36	0.02	196362.26
DFPM	0.25	14.32	0.04	122.67	0.03	1817.39	0.03	25341.36
DFPM-NK	0.24	14.48	0.03	134.21	0.02	1845.18	0.02	25405.48

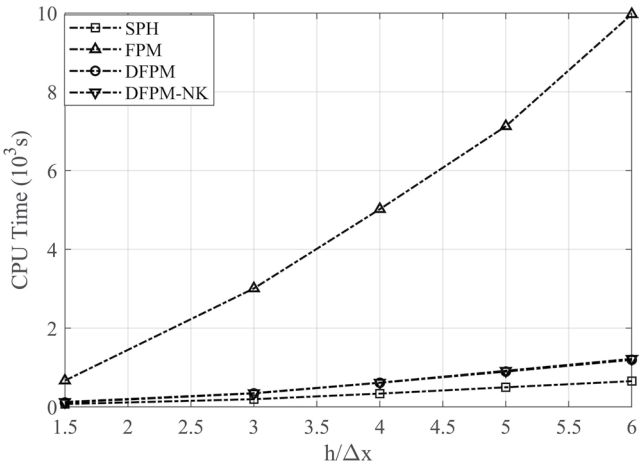

Figure 5. CPU time of Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM) and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) solutions as a function of $h/\Delta x$ in a heterogeneous field of hydraulic conductivity with $\sigma^2_{\ln K} = 0.2$.

which is expected. Table 1 indicates that FPM, DFPM, and DFPM-NK have similar errors, and the reason is that, when FPM does not converge for certain particles, DFPM is used for the particles, as explained in Jiao et al. (2021).

Figure 5 plots the CPU times of the four numerical solutions with different smoothing length of $h = 1.5\Delta x$, $3\Delta x$, $4\Delta x$, $5\Delta x$, and $6\Delta x$ for $\Delta x = 0.125$ m and $\sigma^2_{\ln K} = 0.2$. When the smoothing length increases, the CUP times increase, because more neighbor particles are included in the support domain when h increases. Among the four methods, the CPU time of FPM is significantly larger than those of the other three methods, because FPM needs to solve full-matrix equations given in Equation 8 for each neighbor particle. SPH does not need to solve the equations, and the computational cost of solving diagonal-matrix equations is negligible for DFPM and DFPM-NK. When h increases from $1.5\Delta x$ to $6\Delta x$, the reduction of computational time of DFPM-NK relative to FPM increases from 80.91% ((669.78–127.88)/669.78) (Table 1) to 87.74% ((9967.92–1222.13)/9967.92) (results not shown).

4.2. Transport of Reactive Solute

Figure 6 plots the concentration plumes of species A, B, and C simulated by DFPM-NK for the heterogeneous field of hydraulic conductivity generated with $\sigma^2_{\ln K} = 0.2$. The three rows of Figure 6 are for the results at dimensionless times of $T = 5.56$, 11.13, and 18.54, respectively, and the three columns are for the concentrations of three species A, B, and C, respectively.

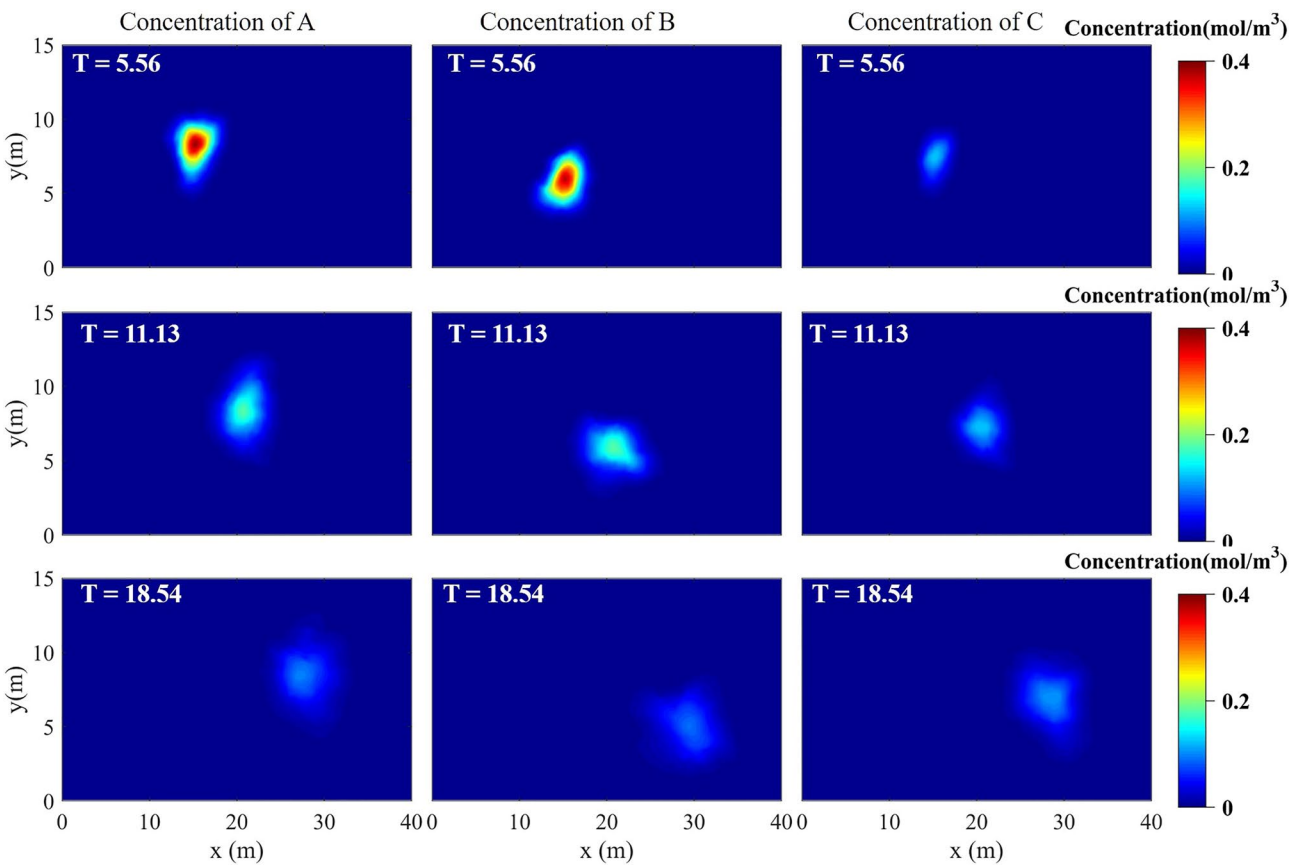

Figure 6. Plumes of Decoupled Finite Particle Method With Normalized Kernel solutions of concentration of species A, B, and C at dimensionless time $T = Ut/I_{\ln K} = 5.56$, 11.13, and 18.54 used in the reactive transport problem for $\sigma^2_{\ln K} = 0.2$.

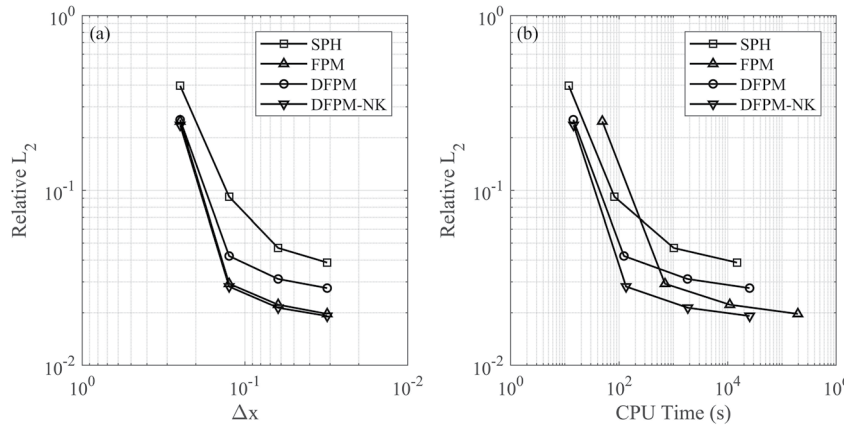

Figure 7. Relative L_2 norm errors of concentration of specie C of the Smoothed Particle Hydrodynamics (SPH), Finite Particle Method (FPM), Decoupled Finite Particle Method (DFPM), and Decoupled Finite Particle Method With Normalized Kernel (DFPM-NK) solutions over the simulation domain as a function of (a) particle spacing(Δx) and (b) elapsed CPU time (seconds) in a heterogeneous field of hydraulic conductivity with $\sigma^2_{\ln K} = 0.2$.

Species A and B are depleted with simulation time, and specie C is produced with simulation time. Computational performance of SPH, FPM, DFPM, and DFPM-NK is evaluated for specie C.

Figure 7 plots the relative L_2 norm of concentration of specie C over the simulation domain as a function of particle spacing Δx and CPU time for $\sigma^2_{\ln K} = 0.2$. Figure 7a indicates that DFPM-NK and FPM have similar computational accuracy, and the accuracy is higher than that of SPH and DFPM. SPH has the worst computational accuracy. Figure 7b indicates that, generally speaking, DFPM-NK is computationally efficient and accurate. Table 1 lists the CPU times of the four numerical solutions with different particle spacings. The CPU time of DFPM-NK is substantially smaller than that of FPM. For example, the CPU time of DFPM-NK is reduced by 87.1% for $\Delta x = 0.03125$ m. For the transport problem of reactive solute, the computational efficient DFPM-NK is of particular importance because reactive tranport modeling is always computationally expensive (Herrera et al., 2010).

We also discuss the effects Peclect number ($Pe = UI_Y/D$) and Damkohler number ($Da = k_d C_{A,0} I_Y/D$) (Tartakovsky et al., 2009) on the performance of the numerical solutions. We have $Pe = 10.3$ based on the values of $U = 0.89$ m/day, $I_Y = 1$m, and $D = 1 \times 10^{-6}$ m^2/s given in Section 3.1, and $Da = 0.74$ based on $C_{A,0} = C_{B,0} = 1$ mol/m^3, $k_f = 7.2 \times 10^{-3}$ mol·m^{-3}·h^{-1} $= 2 \times 10^{-6}$ mol·m^{-3}·s^{-1}, $K_A = 1.67$ mol/m^3, and $K_B = 0.016$ mol/m^3 given in Section 3.2. Note that $k_d = k_f \frac{1}{C_A + K_A} \frac{1}{C_B + K_B}$ due to Equation 19 and $r = k_d C_A C_B$. We consider three values of Peclet number (5.15, 10.3, and 20.6 for $D = 2 \times 10^{-6}$, 1×10^{-6}, and 0.5×10^{-6} m^2/s, respectively) and three Damkohler numbers (0.37, 0.74, and 1.48 for $k_f = 1 \times 10^{-6}$, 2×10^{-6}, and 4×10^{-6} mol·m^{-3}·s^{-1}, respectively). For the nine combinations of three Peclet numbers and three Damkohler numbers, Figure 8 plots the Error in concentration of species C over the simulation domain as a function of dimensionless time with different Peclect and Damkohler numbers for $\sigma^2_{\ln K} = 0.2$. Figure 8 suggests that the transport regimes affect the performance of the four methods. FPM and DFPM-NK behave similarly when the Pe and Da numbers vary. The two methods are less sensitive to the Pe and Da numbers than SPH and DFPM are. SPH is the most sensitive to the two numbers. For all the four methods, their errors increase with the increase of the Da number, also increase with the decrease of the Pe number.

5. Conclusions

This study develops the DFPM-NK method for improving computational accuracy of the DFPM method, so that the computationally efficient DFPM method can be used for solving ADE of groundwater solute transport in heterogeneous porous media. DFPM-NK outperforms DFPM in terms of computational accuracy, because DFPM-NK uses normalized kernels for the modification of kernel gradients. Computational accuracy and efficiency of SPH, FPM, DFPM and DFPM-NK are evaluated using two numerical experiments of advection-dispersion transport of conservative and reactive solutes in heterogeneous porous media. For the numerical experiments of this study, DFPM-NK gives more accurate ADE solutions than SPH and DFPM espe-

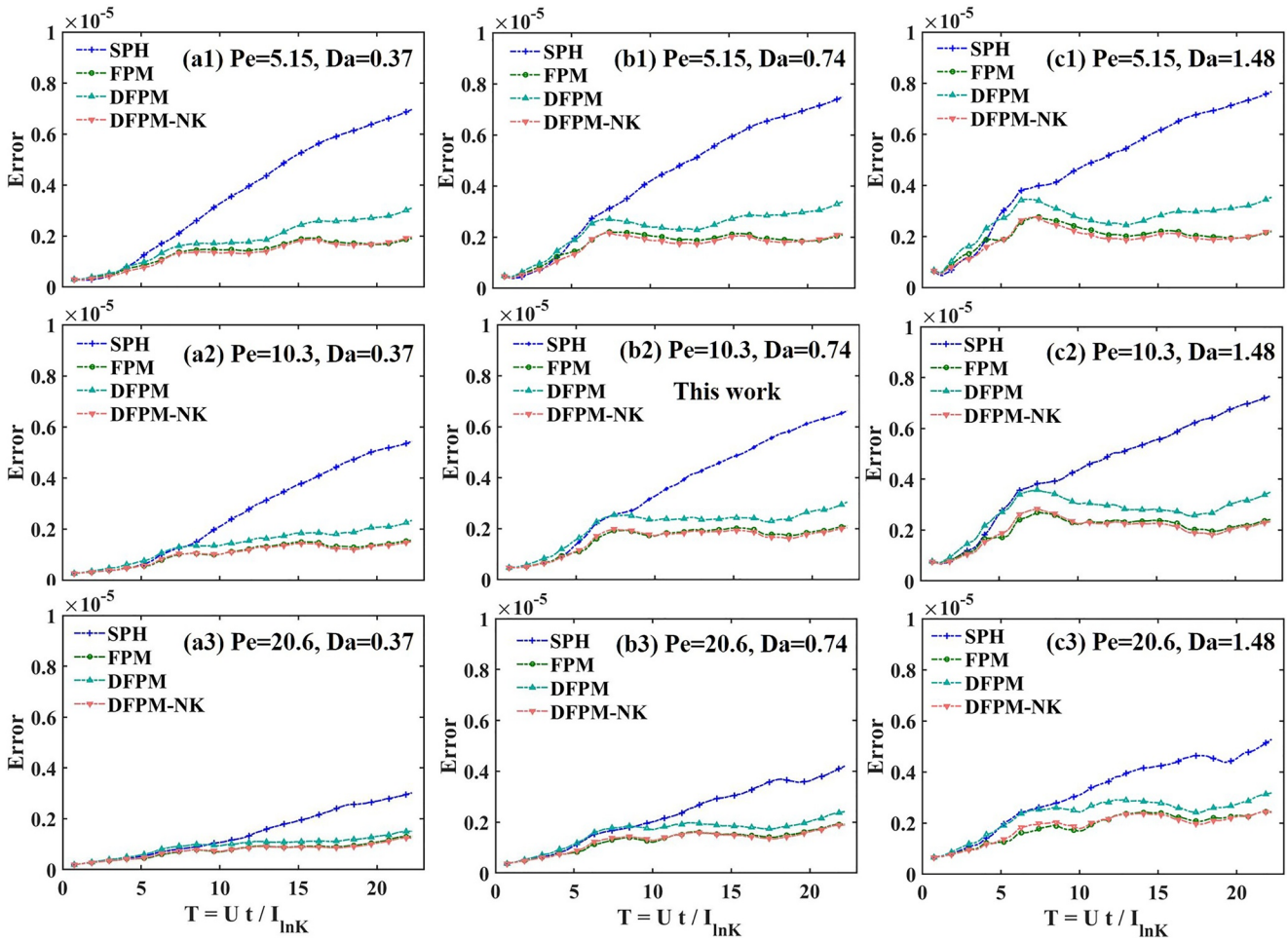

Figure 8. Variation of the error in concentration of specie C over the simulation domain as a function of dimensionless time with different Peclect and Damkohler numbers for $\sigma^2_{\ln K} = 0.2$.

cially for irregular particle distributions generated for heterogeneous hydraulic conductivity fields. DFPM-NK is computationally more stable than FPM, and the two methods have similar computational accuracy, although DFPM-NK is slightly less accurate than FPM. In addition, DFPM-NK is computationally more efficient than FPM, especially for fine spatial resolutions and large smoothing lengths. For the numerical experiments of this study, DFPM-NK reduces the CPU times of FPM by 70% $\sim$ 87%. The computationally efficient and accurate DFPM-NK method is useful for computationally expensive problems of groundwater solute transport, especially reactive transport, in heterogeneous porous media. For the Lagrangian-based SPH and FPM methods, they could be computationally expensive, and computational efficiency is always a concern for practical applications. The computationally efficient and accurate DFPM-NK method is potentially useful for simulating computationally demanding problems of groundwater solute transport, especially reactive transport, in heterogeneous porous media. Applications of DFPM-NK to real-world problem of groundwater reactive transport modeling are warranted in a future study.

Appendix A: Equivalence of FPM and FPM-NK for a Two-Dimensional Problem

This appendix demonstrates that, for a two-dimensional problem, the 3×3 matrix equation for the approximation of FPM modified kernel gradients is mathematically equivalent to the 2×2 matrix equation for the approximation of FPM-NK modified kernel gradient. Following the matrix equation of Equation 8 in FPM, the modified

kernel W_{ij}^{M} and kernel gradient $\frac{\partial^{M} W_{ij}}{\partial \mathbf{x}_i}$ for a two-dimensional problem are approximated simultaneously by solving a matrix equation, that is,

$$\overbrace{\begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial z_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial z_i} \end{bmatrix}}^{\mathbf{M}} \begin{bmatrix} W_{ij}^{M} \\ \frac{\partial^{M} W_{ij}}{\partial x_i} \\ \frac{\partial^{M} W_{ij}}{\partial y_i} \end{bmatrix} = \overbrace{\begin{bmatrix} W_{ij} \\ \frac{\partial W_{ij}}{\partial x_i} \\ \frac{\partial W_{ij}}{\partial y_i} \end{bmatrix}}^{\mathbf{b}}. \tag{A1}$$

To compare Equation A1 with Equation A10 of FPM-NK, the augmented matrix of Equation A1 with an elementary row transformation is given as

$$[\mathbf{M} \mid \mathbf{b}] = \left[\begin{array}{ccc|c} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} & W_{ij} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial x_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial x_i} & \frac{\partial W_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial y_i} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial y_i} & \frac{\partial W_{ij}}{\partial y_i} \end{array}\right] \xrightarrow[\text{elementary row transformation}]{\begin{array}{l} row2 + row1 \times \left(-\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i} \Big/ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} \right) \\ row3 + row1 \times \left(-\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i} \Big/ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} \right) \end{array}}$$

$$\left[\begin{array}{ccc|c} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} & W_{ij} \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial x_i} - \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial x_i} - \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \frac{\partial W_{ij}}{\partial x_i} - W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \frac{\partial W_{ij}}{\partial y_i} - \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \frac{\partial W_{ij}}{\partial y_i} - \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \frac{\partial W_{ij}}{\partial y_i} - W_{ij} \left(\frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) \end{array}\right]. \tag{A2}$$

$$= \left[\begin{array}{ccc|c} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} & W_{ij} \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\partial W_{ij}}{\partial x_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\partial W_{ij}}{\partial x_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \frac{\partial W_{ij}}{\partial x_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\partial W_{ij}}{\partial y_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\partial W_{ij}}{\partial y_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \frac{\partial W_{ij}}{\partial y_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \end{array}\right]$$

To simplify the matrix in Equation A2, $\mathbf{M}$ is expressed as

$$\mathbf{M} = \begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} W_{ij} & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} W_{ij} \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\partial W_{ij}}{\partial x_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\partial W_{ij}}{\partial x_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) \\ 0 & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\partial W_{ij}}{\partial y_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\partial W_{ij}}{\partial y_i} - W_{ij} \frac{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b} \frac{m_j}{\rho_j} W_{ij}} \right) \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix}, \tag{A3}$$

and the inverse matrix of $\mathbf{M}$ is given as

$$\mathbf{M}^{-1}=\frac{1}{a_1\left(b_2c_3-b_3c_2\right)}\begin{bmatrix} a_1 & c_1b_3-b_1c_3 & b_1c_2-c_1b_2 \\ 0 & a_1c_3 & -a_1c_2 \\ 0 & -a_1b_3 & a_1b_2 \end{bmatrix}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\begin{bmatrix} 1 & \frac{c_1b_3-b_1c_3}{a_1} & \frac{b_1c_2-c_1b_2}{a_1} \\ 0 & c_3 & -c_2 \\ 0 & -b_3 & b_2 \end{bmatrix}. \quad \text{(A4)}$$

Based on Equations A1 and A2, the modified kernel and kernel gradient are simultaneously approximated by using $\mathbf{M}^{-1}$, that is,

$$\begin{bmatrix} W_{ij}^{M} \\ \frac{\partial^M W_{ij}}{\partial x_i} \\ \frac{\partial^M W_{ij}}{\partial y_i} \end{bmatrix}=\mathbf{M}^{-1}\begin{bmatrix} W_{ij} \\ \frac{\partial W_{ij}}{\partial x_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}} \\ \frac{\partial W_{ij}}{\partial y_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}} \end{bmatrix}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\begin{bmatrix} 1 & \frac{c_1b_3-b_1c_3}{a_1} & \frac{b_1c_2-c_1b_2}{a_1} \\ 0 & c_3 & -c_2 \\ 0 & -b_3 & b_2 \end{bmatrix}\begin{bmatrix} W_{ij} \\ \frac{\partial W_{ij}}{\partial x_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}} \\ \frac{\partial W_{ij}}{\partial y_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}} \end{bmatrix}. \quad \text{(A5)}$$

Then, the modified first-order derivative of the kernel function in the x and y directions are given as

$$\frac{\partial^M W_{ij}}{\partial x_i}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\left(c_3\left(\frac{\partial W_{ij}}{\partial x_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}}\right)-c_2\left(\frac{\partial W_{ij}}{\partial y_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}}\right)\right), \quad \text{(A6)}$$

$$\frac{\partial^M W_{ij}}{\partial y_i}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\left(-b_3\left(\frac{\partial W_{ij}}{\partial x_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial x_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}}\right)+b_2\left(\frac{\partial W_{ij}}{\partial y_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial y_i}}{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}}\right)\right). \quad \text{(A7)}$$

Because $\frac{\partial\hat{W}_{ij}}{\partial\mathbf{x}_i}=\frac{\frac{\partial W_{ij}}{\partial\mathbf{x}_i}\eta_i-W_{ij}\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial\mathbf{x}_i}}{\eta_i^2}=\frac{\frac{\partial W_{ij}}{\partial\mathbf{x}_i}}{\eta_i}-W_{ij}\frac{\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}\frac{\partial W_{ij}}{\partial\mathbf{x}_i}}{\eta_i^2}$, where $\eta_i=\sum_{j=1}^{N_b}\frac{m_j}{\rho_j}W_{ij}$ is defined in Section 2.2, the modified first-order derivatives of kernel function in the x and y directions of Equations A6 and A7 are simplified to

$$\frac{\partial^M W_{ij}}{\partial x_i}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\left(c_3\eta_i\frac{\partial\hat{W}_{ij}}{\partial x_i}-c_2\eta_i\frac{\partial\hat{W}_{ij}}{\partial y_i}\right), \quad \text{(A8)}$$

$$\frac{\partial^M W_{ij}}{\partial y_i}=\frac{1}{\left(b_2c_3-b_3c_2\right)}\left(-b_3\eta_i\frac{\partial\hat{W}_{ij}}{\partial x_i}+b_2\eta_i\frac{\partial\hat{W}_{ij}}{\partial y_i}\right). \quad \text{(A9)}$$

Following Equation 14, the modified kernel gradient of FPM-NK is approximated by solving the following matrix equation

$$\overbrace{\begin{bmatrix} \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}\frac{\partial\hat{W}_{ij}}{\partial x_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}\frac{\partial\hat{W}_{ij}}{\partial x_i} \\ \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}x_{ji}\frac{\partial\hat{W}_{ij}}{\partial y_i} & \sum_{j=1}^{N_b}\frac{m_j}{\rho_j}y_{ji}\frac{\partial\hat{W}_{ij}}{\partial y_i} \end{bmatrix}}^{\mathbf{M}^N}\begin{bmatrix} \frac{\partial^{M^N}W_{ij}}{\partial x_i} \\ \frac{\partial^{M^N}W_{ij}}{\partial y_i} \end{bmatrix}=\overbrace{\begin{bmatrix} \frac{\partial\hat{W}_{ij}}{\partial x_i} \\ \frac{\partial\hat{W}_{ij}}{\partial y_i} \end{bmatrix}}^{\mathbf{b}}. \quad \text{(A10)}$$

Substituting $\frac{\partial\hat{W}_{ij}}{\partial\mathbf{x}_i}$ to the elements of matrix $\mathbf{M}^N$ in Equation A10 gives

$$\mathbf{M}^N = \begin{bmatrix} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\frac{\partial W_{ij}}{\partial x_i}}{\eta_i} - \frac{W_{ij} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\eta_i^2} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\frac{\partial W_{ij}}{\partial x_i}}{\eta_i} - \frac{W_{ij} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial x_i}}{\eta_i^2} \right) \\ \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} x_{ji} \left(\frac{\frac{\partial W_{ij}}{\partial y_i}}{\eta_i} - \frac{W_{ij} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\eta_i^2} \right) & \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} y_{ji} \left(\frac{\frac{\partial W_{ij}}{\partial y_i}}{\eta_i} - \frac{W_{ij} \sum_{j=1}^{N_b} \frac{m_j}{\rho_j} \frac{\partial W_{ij}}{\partial y_i}}{\eta_i^2} \right) \end{bmatrix} = \begin{bmatrix} \frac{b_2}{\eta_i} & \frac{c_2}{\eta_i} \\ \frac{b_3}{\eta_i} & \frac{c_3}{\eta_i} \end{bmatrix}. \tag{A11}$$

Subsequentially, the inverse matrix of $\mathbf{M^N}$ is given as

$$\left(\mathbf{M}^N\right)^{-1} = \frac{\eta_i^2}{b_2 c_3 - b_3 c_2} \begin{bmatrix} \frac{c_3}{\eta_i} & -\frac{c_2}{\eta_i} \\ -\frac{b_3}{\eta_i} & \frac{b_2}{\eta_i} \end{bmatrix} = \frac{1}{b_2 c_3 - b_3 c_2} \begin{bmatrix} c_3 \eta_i & -c_2 \eta_i \\ -b_3 \eta_i & b_2 \eta_i \end{bmatrix}. \tag{A12}$$

The modified kernel gradients are then derived by using $\left(\mathbf{M}^N\right)^{-1}$, that is,

$$\begin{bmatrix} \frac{\partial^{M^N} W_{ij}}{\partial x_i} \\ \frac{\partial^{M^N} W_{ij}}{\partial y_i} \end{bmatrix} = \left(\mathbf{M}^N\right)^{-1} \begin{bmatrix} \frac{\partial \hat{W}_{ij}}{\partial x_i} \\ \frac{\partial \hat{W}_{ij}}{\partial y_i} \end{bmatrix} = \frac{1}{(b_2 c_3 - b_3 c_2)} \begin{bmatrix} c_3 \eta_i & -c_2 \eta_i \\ -b_3 \eta_i & b_2 \eta_i \end{bmatrix} \begin{bmatrix} \frac{\partial \hat{W}_{ij}}{\partial x_i} \\ \frac{\partial \hat{W}_{ij}}{\partial y_i} \end{bmatrix}. \tag{A13}$$

We obtain the modified first-order derivatives of the kernel function in the x and y directions as

$$\frac{\partial^{M^N} W_{ij}}{\partial x_i} = \frac{1}{(b_2 c_3 - b_3 c_2)} \left(c_3 \eta_i \frac{\partial \hat{W}_{ij}}{\partial x_i} - c_2 \eta_i \frac{\partial \hat{W}_{ij}}{\partial y_i} \right), \tag{A14}$$

$$\frac{\partial^{M^N} W_{ij}}{\partial y_i} = \frac{1}{(b_2 c_3 - b_3 c_2)} \left(-b_3 \eta_i \frac{\partial \hat{W}_{ij}}{\partial x_i} + b_2 \eta_i \frac{\partial \hat{W}_{ij}}{\partial y_i} \right), \tag{A15}$$

Equations A14 and A15 are the same as Equations A8 and A9, which indicates that the kernel gradients approximations of FPM and FPM-NK are equivalent. Therefore, solving the 2×2 matrix in FPM-NK gives the same results as solving the 3×3 matrix in FPM.

Data Availability Statement

Data and codes for reproducing the results of this study are available online at figshare via https://doi.org/10.6084/m9.figshare.19307393.v1.

Acknowledgments
This study was supported by the National Natural Science Foundation of China (No. 41877192). Part of the research was conducted when the first author was a visiting student at the Florida State University.

References

Alvarado-Rodríguez, C. E., Sigalotti, L. D. G., & Klapp, J. (2019). Anisotropic dispersion with a consistent smoothed particle hydrodynamics scheme. *Advances in Water Resources*, *131*, 103374. https://doi.org/10.1016/j.advwatres.2019.07.004

Amicarelli, A., Manenti, S., Albano, R., Agate, G., Paggi, M., Longoni, L., et al. (2020). SPHERA v.9.0.0: A computational fluid dynamics research code, based on the smoothed particle hydrodynamics mesh-less method. *Computer Physics Communications*, *250*, 107157. https://doi.org/10.1016/j.cpc.2020.107157

Avesani, D., Dumbser, M., Vacondio, R., & Righetti, M. (2021). An alternative SPH formulation: ADER-WENO-SPH. *Computer Methods in Applied Mechanics and Engineering*, *382*, 113871. https://doi.org/10.1016/j.cma.2021.113871

Avesani, D., Herrera, P., Chiogna, G., Bellin, A., & Dumbser, M. (2015). Smooth Particle Hydrodynamics with nonlinear moving-least-squares WENO reconstruction to model anisotropic dispersion in porous media. *Advances in Water Resources*, *80*, 43–59. https://doi.org/10.1016/j.advwatres.2015.03.007

Batra, R. C., & Zhang, G. M. (2004). Analysis of adiabatic shear bands in Elasto-thermo-Viscoplastic materials by modified smoothed-particle hydrodynamics (MSPH) method. *Journal of Computational Physics*, *201*(1), 172–190. https://doi.org/10.1016/j.jcp.2004.05.007

Bear, J. (1972). *Dynamics of fluids in porous materials*, American Elsevier.

Benson, D. A., & Meerschaert, M. M. (2008). Simulation of chemical reaction via particle tracking: Diffusion-limited versus thermodynamic rate-limited regimes. *Water Resources Research*, *44*(12), W12201. https://doi.org/10.1029/2008WR007111

Benson, D. A., Tomás, A., Diogo, B., Engdahl, N., Henri, C. V., & Fernàndez-Garcia, D. (2017). A comparison of Eulerian and Lagrangian transport and non-linear reaction algorithms. *Advances in Water Resources*, *99*, 15–37. https://doi.org/10.1016/j.advwatres.2016.11.003

Bolster, D., Paster, A., & Benson, D. A. (2016). A particle number conserving Lagrangian method for mixing-driven reactive transport. *Water Resources Research*, *52*(2), 1518–1527. https://doi.org/10.1002/2015WR018310

Boso, F., Bellin, A., & Dumbser, M. (2013). Numerical simulations of solute transport in highly heterogeneous formations: A comparison of alternative numerical schemes. *Advances in Water Resources*, *52*, 178–189. https://doi.org/10.1016/j.advwatres.2012.08.006

Cleary, P. W., & Monaghan, J. J. (1999). Conduction modelling using smoothed particle hydrodynamics. *Journal of Computational Physics*, *148*(1), 227–264. https://doi.org/10.1006/jcph.1998.6118

de Barros, F. P. J., Fiori, A., Boso, F., & Bellin, A. (2015). A theoretical framework for modeling dilution enhancement of non-reactive solutes in heterogeneous porous media. *Journal of Contaminant Hydrology*, *175–176*, 72–83. https://doi.org/10.1016/j.jconhyd.2015.01.004

Espanol, P., & Revenga, M. (2003). Smoothed dissipative particle dynamics. *Physical Review A*, *67*(2), 026705. https://doi.org/10.1103/PhysRevE.67.026705

Harbaugh, A. (2000). *MODFLOW-2000, the US geological survey modular groundwater model: User guide to modularization concepts and the ground-water flow process*. US Geological Survey.

Herrera, P. A., & Beckie, R. D. (2013). An assessment of particle methods for approximating anisotropic dispersion. *International Journal for Numerical Methods in Fluids*, *71*(5), 634–651. https://doi.org/10.1002/fld.3676

Herrera, P. A., Cortínez, J. M., & Valocchi, A. J. (2017). Lagrangian scheme to model subgrid-scale mixing and spreading in heterogeneous porous media. *Water Resources Research*, *53*(4), 3302–3318. https://doi.org/10.1002/2016wr019994

Herrera, P. A., Massabó, M., & Beckie, R. D. (2009). A meshless method to simulate solute transport in heterogeneous porous media. *Advances in Water Resources*, *32*(3), 413–429. https://doi.org/10.1016/j.advwatres.2008.12.005

Herrera, P. A., Valocchi, A. J., & Beckie, R. D. (2010). A multidimensional streamline-based method to simulate reactive solute transport in heterogeneous porous media. *Advances in Water Resources*, *33*(7), 711–727. https://doi.org/10.1016/j.advwatres.2010.03.001

Jiang, T., Ouyang, J., Ren, J. L., Yang, B., & Xu, X. (2012). A mixed corrected symmetric SPH (MC-SSPH) method for computational dynamic problems. *Computer Physics Communications*, *183*(1), 50–62. https://doi.org/10.1016/j.cpc.2011.08.016

Jiao, T., Ye, M., Jin, M. G., & Yang, J. (2021). A finite particle method (FPM) for Lagrangian simulation of conservative solute transport in heterogeneous porous media. *Advances in Water Resources*, *156*, 104043. https://doi.org/10.1016/j.advwatres.2021.104043

Jubelgas, M., Springel, V., & Dolag, K. (2004). Thermal conduction in cosmological SPH simulations. *Monthly Notices of the Royal Astronomical Society*, *351*(2), 423–435. https://doi.org/10.1111/j.1365-2966.2004.07801.x

Liu, M. B., & Liu, G. R. (2006). Restoring particle consistency in smoothed particle hydrodynamics. *Applied Numerical Mathematics*, *56*(1), 19–36. https://doi.org/10.1016/j.apnum.2005.02.012

Liu, M. B., & Liu, G. R. (2010). Smoothed particle hydrodynamics (SPH): An overview and recent developments. *Archives of Computational Methods in Engineering*, *17*(1), 25–76. https://doi.org/10.1007/s11831-010-9040-7

Liu, M. B., Xie, W. P., & Liu, G. R. (2005). Modeling incompressible flows using a finite particle method. *Applied Mathematical Modelling*, *29*(12), 1252–1270. https://doi.org/10.1016/j.apm.2005.05.003

Pebesma, E. J. (2004). Multivariable geostatistics in S: The Gstat package. *Computers & Geosciences*, *30*(7), 683–691. https://doi.org/10.1016/j.cageo.2004.03.012

Sole-Mari, G., & Fernàndez-Garcia, D. (2018). Lagrangian modeling of reactive transport in heterogeneous porous media with an automatic locally adaptive particle support volume. *Water Resources Research*, *54*(10), 8309–8331. https://doi.org/10.1029/2018wr023033

Sole-Mari, G., Fernàndez-Garcia, D., Rodriguez-Escales, P., & Xavier Sanchez-Vila, X. (2017). A KDE-based random walk method for modeling reactive transport with complex kinetics in porous media. *Water Resources Research*, *53*(11), 9019–9039. https://doi.org/10.1002/2017WR021064

Sole-Mari, G., Schmidt, M. J., Pankavich, S. D., & Benson, D. A. (2019). Numerical equivalence between SPH and probabilistic mass transfer methods for Lagrangian simulation of dispersion. *Advances in Water Resources*, *126*, 108–115. https://doi.org/10.1016/j.advwatres.2019.02.009

Tartakovsky, A. M. (2010). Lagrangian simulations of unstable gravity-driven flow of fluids with variable density in randomly heterogeneous porous media. *Stochastic Environmental Research and Risk Assessment*, *24*(7), 993–1002. https://doi.org/10.1007/s00477-010-0402-3

Tartakovsky, A. M., & Meakin, P. (2005). A smoothed particle hydrodynamics model for miscible flow in three-dimensional fractures and the two-dimensional Rayleigh–Taylor instability. *Journal of Computational Physics*, *207*(2), 610–624. https://doi.org/10.1016/j.jcp.2005.02.001

Tartakovsky, A. M., Meakin, P., Scheibe, T. D., & Eichler West, R. M. (2007). Simulations of reactive transport and precipitation with smoothed particle hydrodynamics. *Journal of Computational Physics*, *222*(2), 654–672. https://doi.org/10.1016/j.jcp.2006.08.013

Tartakovsky, A. M., Meakin, P., Scheibe, T. D., & Wood, B. D. (2007). A smoothed particle hydrodynamics model for reactive transport and mineral precipitation in porous and fractured porous media. *Water Resources Research*, *43*(5), W05437. https://doi.org/10.1029/2005wr004770

Tartakovsky, A. M., Tartakovsky, G. D., & Scheibe, T. D. (2009). Effects of incomplete mixing on multicomponent reactive transport. *Advances in Water Resources*, *32*(11), 1674–1679. https://doi.org/10.1016/j.advwatres.2009.08.012

Tartakovsky, A. M., Trask, N., Pan, K., Jones, B., Pan, W., & Williams, J. R. (2015). Smoothed particle hydrodynamics and its applications for multiphase flow and reactive transport in porous media. *Computational Geosciences*, *20*(4), 807–834. https://doi.org/10.1007/s10596-015-9468-9

Xu, X., & Deng, X. L. (2016). An improved weakly compressible SPH method for simulating free surface flows of viscous and viscoelastic fluids. *Computer Physics Communications*, *201*, 43–62. https://doi.org/10.1016/j.cpc.2015.12.016

Yildiz, M., Rook, R. A., & Suleman, A. (2009). SPH with multiple boundary tangent method. *International Journal for Numerical Methods in Engineering*, *77*(10), 1416–1438. https://doi.org/10.1002/nme.2458

Zhang, G. M., & Batra, R. C. (2009). Symmetric smoothed particle hydrodynamics (SSPH) method and its application to elastic problems. *Computational Mechanics*, *43*(3), 321–340. https://doi.org/10.1007/s00466-008-0308-9

Zhang, Z. L., & Liu, M. B. (2018). A decoupled finite particle method for modeling incompressible flows with free surfaces. *Applied Mathematical Modelling*, *60*, 606–633. https://doi.org/10.1016/j.apm.2018.03.043

Zhang, Z. L., Walayat, K., Huang, C., Chang, J. Z., & Liu, M. B. (2019). A finite particle method with particle shifting technique for modeling particulate flows with thermal convection. *International Journal of Heat and Mass Transfer*, *128*, 1245–1262. https://doi.org/10.1016/j.ijheatmasstransfer.2018.09.074

Zhu, Q., Hernquist, L., & Li, Y. (2015). Numerical convergence in smoothed particle Hydrodynamics. *The Astrophysical Journal*, *800*(1), 6. https://doi.org/10.1088/0004-637x/800/1/6

References From the Supporting Information

Alvarado-Rodríguez, C. E., Sigalotti, L. D. G., & Klapp, J. (2019). Anisotropic dispersion with a consistent smoothed particle hydrodynamics scheme. *Advances in Water Resources*, *131*, 103374. https://doi.org/10.1016/j.advwatres.2019.07.004

Zimmermann, S., Koumoutsakos, P., & Kinzelbach, W. (2001). Simulation of pollutant transport using a particle method. *Journal of Computational Physics*, *173*(1), 322–347. https://doi.org/10.1006/jcph.2001.6879