Advances in Water Resources 140 (2020) 103577

Contents lists available at ScienceDirect

Advances in Water Resources

journal homepage: www.elsevier.com/locate/advwatres

A mass-transfer particle-tracking method for simulating transport with discontinuous diffusion coefficients☆

Michael J. Schmidt[a,*], Nicholas B. Engdahl[c], Stephen D. Pankavich[b], Diogo Bolster[a]

[a] Department of Civil and Environmental Engineering and Earth Sciences, University of Notre Dame, Notre Dame, IN 46556, USA
[b] Department of Applied Mathematics and Statistics, Colorado School of Mines, Golden, CO 80401, USA
[c] Department of Civil and Environmental Engineering, Washington State University, Pullman, WA 99164, USA

ARTICLE INFO

Keywords:
Lagrangian modeling
Particle methods
Mass-transfer particle-tracking
Imperfect mixing
Diffusion-reaction equation
Composite porous media
Discontinuous diffusion coefficients

ABSTRACT

The problem of a spatially discontinuous diffusion coefficient ($D(\boldsymbol{x})$) is one that may be encountered in hydrogeologic systems due to natural geological features or as a consequence of numerical discretization of flow properties. To date, mass-transfer particle-tracking (MTPT) methods, a family of Lagrangian methods in which diffusion is jointly simulated by random walk and diffusive mass transfers, have been unable to solve this problem. This manuscript presents a new mass-transfer (MT) algorithm that enables MTPT methods to accurately solve the problem of discontinuous $D(\boldsymbol{x})$. To achieve this, we derive a semi-analytical solution to the discontinuous $D(\boldsymbol{x})$ problem by employing a predictor-corrector approach, and we use this semi-analytical solution as the weighting function in a reformulated MT algorithm. This semi-analytical solution is generalized for cases with multiple 1D interfaces as well as for 2D cases, including a 2×2 tiling of 4 subdomains that corresponds to a numerically-generated diffusion field. The solutions generated by this new mass-transfer algorithm closely agree with an analytical 1D solution or, in more complicated cases, trusted numerical results, demonstrating the success of our proposed approach.

1. Introduction

Simulating diffusive transport under the condition of a spatially discontinuous diffusion coefficient is a challenging problem that is frequently encountered in hydrogeological contexts (Uffink, 1983; LaBolle et al., 2000; Appuhamillage et al., 2010; Semra et al., 1970; Oukili et al., 2019). Physically, this can occur wherever there is an abrupt change in the material properties of a medium, like the sharp interfaces between different depositional units. Sharp discontinuities can also be seen in, for example: fractured or composite media, local compaction zones, or at the interface between a saturated and unsaturated zone. From a numerical perspective, any non-constant hydraulic conductivity field that is discretized will generate a diffusion/dispersion field containing numerous discontinuities. Interpolation or averaging methods have been used in the past to smooth these discontinuities, and these can be effective as long as the differences in magnitude of the parameter(s) across the interface is sufficiently small (in general, less than an order of magnitude). However, when the difference in diffusion coefficients between cells, or regions of a domain, becomes sufficiently large, the simplest versions of these methods can fail, and overcoming this challenge requires a more nuanced approach.

Random-walk particle-tracking (RWPT) methods are a class of stochastic Lagrangian (mesh-free or gridless) methods that are commonly used to simulate advective-diffusive transport. These methods were originally formulated in the context of conservative (non-chemically reactive) transport or cases of simple, linear reactions, such as sorbing solutes or first-order decay (LaBolle et al., 1996; Salamon et al., 2006). They are popular because they introduce no numerical diffusion into the simulation of the advection (hyperbolic) operator, and they also escape the burden imposed by restrictive stability conditions in Eulerian (grid-based) methods, resulting in lower run times than corresponding Eulerian methods (Benson et al., 2017). Further, because RWPT is a stochastic algorithm, statistics of concentrations can be readily generated instead of expected values (point estimates). In this context, the problem of discontinuous diffusion coefficients has received much attention, resulting in various methods for overcoming the difficulties of simulating such a system (e.g., Uffink, 1983; Appuhamillage et al., 2010; Semra et al., 1970; Hoteit et al., 2002; Bechtold et al., 2011; Oukili et al., 2019; Bagtzoglou et al., 1992; LaBolle et al., 1996; 2000), each with their own merits and drawbacks.

One of the major advantages of classical RWPT is its speed, due to the fact that every particle is completely independent of its neighbors.

☆ This work was partially supported by the US Army Research Office under Contract/Grant number W911NF-18-1-0338; the National Science Foundation under awards EAR-1417145 and DMS-1614586; and the DOE Office of Science under award DE-SC0019123.

* Corresponding author.
E-mail addresses: mschmi23@nd.edu (M.J. Schmidt), nick.engdahl@wsu.edu (N.B. Engdahl), pankavic@mines.edu (S.D. Pankavich), bolster@nd.edu (D. Bolster).

https://doi.org/10.1016/j.advwatres.2020.103577
Received 1 November 2019; Received in revised form 17 March 2020; Accepted 21 March 2020
Available online 25 March 2020

However, this also means that complex reactions cannot be simulated since particle interactions are not allowed. Recent developments in the field of RWPT have enabled methods to simulate complex and nonlinear chemical reactions in the presence of transport using either collision-based reactions between particles of opposite species (Benson and Meerschaert, 2008; Paster et al., 2014; Bolster et al., 2016; Schmidt et al., 2017; Sole-Mari et al., 2017; Sole-Mari and Fernàndez-Garcia, 2018), or by treating individual particles as reaction volumes that communicate via diffusive mass transfers (Benson and Bolster, 2016). The latter, referred to as mass-transfer particle-tracking (MTPT) algorithms, offer the increased flexibility of being able to model arbitrarily complex chemical reactions at relatively low computational cost (Engdahl et al., 2017; Schmidt et al., 2019b), including generalized "reactions" such as the aging of water parcels (Benson et al., 2019b). The mass-transfer (MT) portion of these MTPT methods has been demonstrated to solve the diffusion equation to $\mathcal{O}(\Delta t)$ (Schmidt et al., 2018) and exhibit superlinear convergence as particle numbers grow large (Schmidt et al., 2019a). Additionally, a method for parallelizing the MTPT method via domain decomposition has recently been developed and achieves linear speedup up to hundreds of computational cores/subdomains (Engdahl et al., 2019). MTPT methods have also been shown to be related to smoothed-particle hydrodynamics (SPH) methods (e.g., Herrera et al., 2009; Herrera and Beckie, 2013; Gingold and Monaghan, 1977; Monaghan, 2012) under specific modeling choices, including the use of a Gaussian spatial kernel (Sole-Mari et al., 2019b). Despite these advances, past work on MTPT methods has neither addressed the impact of a discrete parameter field on the mass transfer operations nor accounted for the possible errors that may be incurred.

All previously-mentioned random-walk methods may be employed for diffusion coefficients with spatial discontinuities because they are capable of simulating small-scale mixing and non-mixed spreading of solute separately (Benson et al., 2019a). In other words, spreading may be simulated by a random walk and mixing as a mass transfer. However, accuracy of the mass-transfer step is only preserved for a smoothly varying field (i.e., one in which interpolation may be reasonably performed), and the current MTPT schemes incur error when there is a sharp discontinuity. This is similar to the problems identified by LaBolle et al. (2000) for classical RWPT. MTPT has clear applications for highly accurate simulations of mixing-limited reactive transport, but this issue undermines its accuracy. Thus, the purpose of the current paper is to address this deficiency and ensure that MTPT methods remain accurate even in such a case.

In Section 2, we outline the specific mathematical problem on which we will focus, and introduce the methods used to solve the problem in Section 3. In Section 3.1, we provide a brief overview of RWPT methods and discuss how the problem of discontinuous diffusion coefficients is typically handled, with specific focus on a particular predictor-corrector technique (LaBolle et al., 2000) that we extend to MTPT. In Section 3.2, we outline our approach to solving the discontinuous diffusion coefficient problem with an MTPT method by employing an alternative mass-transfer kernel. Section 4 is devoted to discussing the results of applying the new MTPT method. Finally, Section 5 presents the conclusions drawn from our work.

2. Analytic model

We consider a chemically-conservative, single species, purely diffusive system that may be described by the (heterogeneous) diffusion equation

$$\frac{\partial C}{\partial t} = \nabla \cdot (D(\boldsymbol{x})\nabla C), \qquad \boldsymbol{x} \in \Omega \subseteq \mathbb{R}^d, \qquad t > 0, \tag{1}$$

where $C(t, \boldsymbol{x})$ [mol L^{-d}] is the concentration of the single species, $D(\boldsymbol{x})$ [L^2 T^{-1}] is the scalar diffusion coefficient, which, for our purposes, may be a function of space. For this work, we concern ourselves with the condition where D may be discontinuous. This case leads to infinite spatial derivatives at all discontinuities, so the question is how best to numerically evaluate the $\nabla \cdot D$ term within the chosen method to minimize artifacts of the discontinuity. The discontinuity we use for this study is created by partitioning the domain, Ω, into N_Ω subdomains such that $\Omega = \Omega_1 \cup \Omega_2 \cup \ldots \cup \Omega_{N_\Omega}$, where each subdomain has its own constant-valued diffusion coefficient, D_i, $i = 1, \ldots, N_\Omega$, and the interface between subdomains i and j is denoted γ_{ij}.

3. Computational methods

3.1. Random-walk particle-tracking method

The classical Lagrangian method for simulating the system of interest is standard random-walk particle-tracking (RWPT) (Thomson, 1987; LaBolle et al., 1996). In these methods, masses are divided among particles that simulate diffusion via the Langevin equation (formulated for homogeneous D)

$$\boldsymbol{X}_i(t + \Delta t) = \boldsymbol{X}_i(t) + \xi_i \sqrt{2D\Delta t}, \tag{2}$$

where $\boldsymbol{X}_i(t)$ is the position of particle i at time t, Δt is the chosen simulated timestep, and ξ_i is a d-dimensional vector of random numbers drawn from a standard normal, $\mathcal{N}(0, 1)$, distribution. In this basic form, RWPT methods are unable to simulate the problem of discontinuous diffusion coefficients ($D(\boldsymbol{x})$), described in Section 2. Conceptually, the problem is that, during the course of a single-step random walk, a particle may "see" diffusion at the rates on both sides of the discontinuity in $D(\boldsymbol{x})$; however, there are well-documented strategies for overcoming this.

The first general group of strategies are reflection methods (Uffink, 1983; Appuhamillage et al., 2010; Semra et al., 1970; Hoteit et al., 2002), which may include a nonlinearly decomposed time step (Bechtold et al., 2011), interpolation methods (Bagtzoglou et al., 1992), or a combination thereof (LaBolle et al., 1996). A selection of these are reviewed and compared in LaBolle et al. (1998), and the conclusion reached therein is that, among the methods considered, those of Uffink (1983); Semra et al. (1970) provide the best accuracy. A benchmark comparison of various methods is also conducted by Lejay and Pichot (2016) who distinguish between methods that preserve or lose important physical and numerical properties, and recent work of the same authors presents a method that employs skew Brownian motion densities with exponential timestepping to capture the dynamics of the discontinuous $D(\boldsymbol{x})$ problem (Lejay et al., 2019). Another recent approach (Oukili et al., 2019) employs negative-mass particles in a partial reflection scheme, so as to keep the total mass in a system constant and maintain particle independence.

To demonstrate how discontinuous $D(\boldsymbol{x})$ is handled with RWPT, and because we later use this method to generate reference solutions, we briefly discuss the work of LaBolle et al. (2000). We consider this method because it bears resemblance to the algorithm we present in Section 3.2. Also, it is relatively simple to implement, and the extension to greater than one spatial dimension is straightforward, unlike some other approaches. This method may be thought of as a predictor-corrector approach, and is formulated as

$$\boldsymbol{\mathcal{X}}_i = \boldsymbol{X}_i(t) + \xi_i \sqrt{2D(\boldsymbol{X}_i)\Delta t}. \tag{3}$$

$$\boldsymbol{X}_i(t + \Delta t) = \boldsymbol{X}_i(t) + \xi_i \sqrt{2D(\boldsymbol{\mathcal{X}}_i)\Delta t}, \tag{4}$$

In words, a "predictor" random walk is first taken from $\boldsymbol{X}_i(t)$ to $\boldsymbol{\mathcal{X}}_i$ in (3) to determine the diffusion coefficient that is then used in the "corrected" random walk from $\boldsymbol{X}_i(t)$ to $\boldsymbol{X}_i(t + \Delta t)$ in (4). A subtle but important point is that the same random number, ξ_i must be used in (4) that was generated for (3).

3.2. Mass-transfer particle-tracking method

Another family of Lagrangian methods that has gained attention recently are the mass-transfer particle-tracking (MTPT) methods, which

are the focus of this work (Benson and Bolster 2016; Engdahl et al., 2017; Schmidt et al., 2018, 2019a, 2019b). These methods are quite similar to RWPT methods in that diffusion is typically still simulated, in part, by random walks. However, the important distinction is that particle masses are no longer fixed and can be transferred among particles according to an algorithm that also simulates diffusion. The MT algorithm may be given as

$$m_i(t+\Delta t) = m_i(t) + \sum_{j=1}^{N} \mathcal{W}_{ij}\left[m_j(t) - m_i(t)\right], \tag{5}$$

where $m_i(t)$ is the mass carried by particle i at time t, N is the number of particles, and

$$\mathcal{W}_{ij} := \frac{W(\boldsymbol{X}_i, \boldsymbol{X}_j; h)}{\rho_{ij}}. \tag{6}$$

Note that this formulation is equivalent to choosing $\beta = 1$, in the context of Sole-Mari et al. (2019b). Above, W is a Gaussian weighting function that determines the amount of mass transferred from particle j to particle i (or vice-versa because W, in this case, is symmetric with respect to $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$) and ρ_{ij} is a normalizing constant that ensures conservation of mass. We specify here that this normalization would not be required in the limiting, infinite-particle case, but for any finite number of particles, N samples from the weighting function W (which is necessarily a density) will not sum to unity and thus not conserve mass. As such, we normalize our discretized density according to (6).

In the case of isotropic diffusion, we have

$$W(\boldsymbol{X}_i, \boldsymbol{X}_j; (D_i + D_j)\Delta t) = \left(2\pi(D_i + D_j)\Delta t\right)^{-d/2} \exp\left[-\frac{\|\boldsymbol{X}_j - \boldsymbol{X}_i\|^2}{2(D_i + D_j)\Delta t}\right], \tag{7}$$

where $d = 1, 2, 3$, is the number of spatial dimensions, and $D_k := D(\boldsymbol{X}_k)$. The matrix-vector form of (5) is written as

$$\boldsymbol{m}(t+\Delta t) = \boldsymbol{T}\boldsymbol{m}(t), \tag{8}$$

in which

$$\boldsymbol{T} := \boldsymbol{I} + \boldsymbol{\mathcal{W}} - \mathrm{diag}(\boldsymbol{\mathcal{W}}\boldsymbol{1}), \tag{9}$$

where $\boldsymbol{I}$ is the $N \times N$ identity matrix, $\boldsymbol{1}$ is an $N \times 1$ vector of ones, and $\mathrm{diag}(\boldsymbol{x})$ is a square matrix with the entries of vector $\boldsymbol{x}$ on its main diagonal. A popular choice for ρ_{ij} that results in symmetric $\boldsymbol{\mathcal{W}}$ (and thus conservation of mass by the operator $\boldsymbol{T}$) is

$$\rho_{ij} := \frac{[\boldsymbol{\mathcal{W}}\boldsymbol{1}]_i + [\boldsymbol{1}^T\boldsymbol{\mathcal{W}}]_j}{2}, \tag{10}$$

or, in words, ρ_{ij} is the arithmetic mean of the sums of row i and column j.

We see in the formulation outlined above that (7) is the analytical solution to the diffusion equation over the interval $[0, \Delta t]$ as a function of position $\boldsymbol{X}_j$, given a unit point source located at $\boldsymbol{X}_i$ (or vice-versa, with respect to $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$). If we rewrite (7) as

$$\begin{aligned} W(\boldsymbol{X}_i, \boldsymbol{X}_j; (D_i + D_j)\Delta t) &= \left(2\pi(D_i + D_j)\Delta t\right)^{-d/2} \exp\left[-\frac{\|\boldsymbol{X}_j - \boldsymbol{X}_i\|^2}{2(D_i + D_j)\Delta t}\right] \\ &= \left(4\pi\left(\frac{D_i + D_j}{2}\right)\Delta t\right)^{-d/2} \exp\left[-\frac{\|\boldsymbol{X}_j - \boldsymbol{X}_i\|^2}{4\left(\frac{D_i+D_j}{2}\right)\Delta t}\right] \\ &= \left(4\pi\widehat{D}\Delta t\right)^{-d/2} \exp\left[-\frac{\|\boldsymbol{X}_j - \boldsymbol{X}_i\|^2}{4\widehat{D}\Delta t}\right], \end{aligned} \tag{11}$$

we see that his formulation computes an effective diffusion coefficient, $\widehat{D}$, as the arithmetic mean of the diffusion coefficients at particle locations $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ (or, equivalently, linearly interpolates the diffusion coefficients between these two points and chooses the value located at the midpoint). However, as discussed in LaBolle et al. (1996), this linear interpolation fails in the case of discontinuous diffusion coefficients without the inclusion of some sort of reflection scheme to account for the infinite divergence in D at the interface. Put simply, this method only yields favorable results when $D(\boldsymbol{x})$ can be reasonably approximated with a linear fit over distances on the order of $\ell := \sqrt{(D_i + D_j)\Delta t}$, and clearly a linear approximation of an infinitely steep gradient will not suffice. We note here that if we employ a harmonic mean to compute $\widehat{D}$ in (11) we can obtain low-error results in 1D and in certain 2D cases, but this approximation is not reliable in general. As such, it would seem that we need a more flexible functional form for our weighting function W, and the best possible choice would be the analytical solution to the diffusion equation that accounts for discontinuities in $D(\boldsymbol{x})$.

3.2.1. Analytical solution for mass-transfer weight function

Carslaw and Jaeger (1959) present a relatively simple solution in 1D for the problem of two subdomains. We generalize that solution here for an instantaneous pulse of unit concentration at location $x = x_0 \in (-\infty, \infty)$ and time $t = 0$ (i.e., $C(t = 0, x) = \delta(x - x_0)$). More specifically, for a chosen $\gamma \in (-\infty, \infty)$ we define the subdomains to be $\Omega_1 = (-\infty, \gamma]$ and $\Omega_2 = (\gamma, \infty)$, each with constant diffusion coefficients D_1 and D_2, respectively. See Fig. 1(a) for a conceptual depiction of this system. If $x_0 \geq \gamma$, we have

$$C_A(t, x) = C_1(t, x; D_1, D_2)I_{\Omega_1}(x) + C_2(t, x; D_1, D_2)I_{\Omega_2}(x), \tag{12}$$

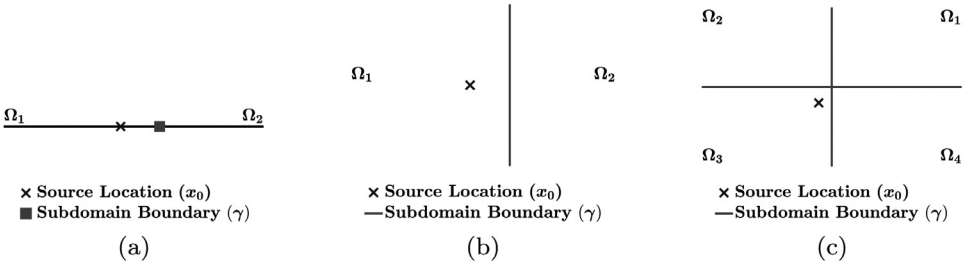

Fig. 1. Conceptual figures for the discontinuous diffusion coefficient problems we consider. (a) 1D problem with 2 subdomains, Ω_1 and Ω_2, with respective diffusion coefficients D_1 and D_2. The subdomains are split by the point $x = \gamma$, and the point-source initial condition is located at the point $x = \gamma$. (b) 2D problem with 2 subdomains, Ω_1 and Ω_2, with respective diffusion coefficients D_1 and D_2. The subdomains are split by the line $x = \gamma$, and the point-source initial condition is located at the point $\boldsymbol{x} = (x_0, y_0)$. (c) 2D problem with 4 subdomains, Ω_1, Ω_2, Ω_3, Ω_4, with respective diffusion coefficients D_1, D_2, D_3, D_4. The subdomains are split by the lines $x = \gamma_x$ and $y = \gamma_y$ and the point-source initial condition is located at the point $\boldsymbol{x} = (x_0, y_0)$.

where $I_A(z)$ is the indicator function on the set A, such that

$$I_A(z) := \begin{cases} 1, & z \in A \\ 0, & z \notin A \end{cases}, \tag{13}$$

and

$$C_1(t,x;D_1,D_2) = \frac{D_2 D_1 (\pi D_1 D_2 t)^{-1/2}}{(D_2\sqrt{D_1} + D_1\sqrt{D_2})} \exp \frac{-\left|x-\gamma-(x_0-\gamma)\sqrt{D_1/D_2}\right|^2}{4D_1 t}, \tag{14}$$

$$C_2(t,x;D_1,D_2) = \frac{1}{2\sqrt{\pi D_2 t}} \exp \frac{-|x-x_0|^2}{4D_2 t} + \frac{D_2\sqrt{D_1} - D_1\sqrt{D_2}}{2(D_2\sqrt{D_1} + D_1\sqrt{D_2})\sqrt{\pi D_2 t}} \exp \frac{-|x+x_0-2\gamma|^2}{4D_2 t}, \tag{15}$$

and if $x_0 < \gamma$, the complementary solution is

$$\hat{C}_A(t,x) = C_2(t,x;D_2,D_1) I_{\Omega_1}(x) + C_1(t,x;D_2,D_1) I_{\Omega_2}(x). \tag{16}$$

For the sake of compact notation, we may combine (12) and (16) into

$$W_A(t,x) := \hat{C}_A(t,x;x_0) I_{\Omega_1}(x_0) + C_A(t,x;x_0) I_{\Omega_2}(x_0). \tag{17}$$

We note that the solution given in (17) is not symmetric with respect to x and x_0 (this is seen most clearly in the numerator of the exponential term in (14)); however, in application and due to the sharp decay in the exponential, (17) is typically symmetric to the order of machine precision. As our objective is to eliminate errors, including those from a lack of symmetry, we alter the mass-transfer algorithm given in (5) such that

$$m_i(t+\Delta t) = m_i(t) + \sum_{j=1}^{N} \mathcal{W}_{ij} m_j(t) - \sum_{j=1}^{N} \mathcal{W}_{ji} m_i(t), \tag{18}$$

in which the mass of particle i at time $t+\Delta t$ is its mass at time t plus the sum of all the incoming mass-transfers, minus the sum of all outgoing mass-transfers. Also, because $\mathcal{W}_{ij} \neq \mathcal{W}_{ji}$, we now strictly define $\mathcal{W}_{ij}$ to be the normalized weight for the mass transfer from particle j to particle i (the converse is no longer true). Eq. (18) may be rewritten in an analogous form to (8), with

$$\boldsymbol{T} := \boldsymbol{I} + \boldsymbol{\mathcal{W}} - \operatorname{diag}\left(\mathbf{1}^T \boldsymbol{\mathcal{W}}\right). \tag{19}$$

If we use (17) as our weighting function in (19), again employing the symmetric normalization given in (10) to form $\boldsymbol{\mathcal{W}}$ (because $\boldsymbol{W}$ is almost certainly symmetric to machine precision), then we obtain a mass-transfer method that generates very little error in simulating this system. The algorithm for conducting a single mass transfer (within a timestep of length Δt) according to this method is given in Algorithm 1 , in which `WtFunction()` is defined to be (17).

A major drawback of this method is that we must possess an analytic solution to the system of interest. Granted, for small Δt, this solution is still relatively flexible; for example, we can still use this solution in the case of a 1D domain with three subdomains (considered in Section 4), provided that the time step is sufficiently small or the magnitude of diffusion in the center domain is low enough that mass-transfers do not "see" two subdomain boundaries at the same time. Calculating an analytical solution is a non-trivial enterprise in spatial dimensions greater than one, particularly if we have a more complicated interface (for instance a 2×2 tiling of 4 subdomains in 2D, which we consider in Section 4). In fact, even for the relatively "simple" 2D problem of two half-planes, split by the line $x = \gamma$ (as considered in Section 4.2), the analytical solution is quite complex and likely infeasible as a mass-transfer kernel (see Shendeleva, 2001). As such, we seek a semi-analytical solution to the discontinuous $D(\boldsymbol{x})$ problem, valid for small Δt, that will be flexible enough that it may be applied, by extension, to higher-dimensional problems. We discuss this approach in the following section.

Algorithm 1: Mass-transfer Algorithm for Non-symmetric Weighting Function

Input: Particle positions, $X = \boldsymbol{X}(t)$, and particle masses $m = \boldsymbol{m}(t)$.
Output: Updated particle masses, $m = \boldsymbol{m}(t+\Delta t)$.

```
                                          ▷ Build weight matrix
for i = 1 to N do
    for j = 1 to N do
        W(i,j) = WtFunction(x_0 = X(j), x = X(i), γ, D_1, D_2, Δt)
    end
end
                                          ▷ Normalize weight matrix
for i = 1 to N do
    for j = 1 to N do
        W(i,j) = W(i,j) / (Sum(W(i,:)) + Sum(W(:,j)) / 2)
    end
end
                                          ▷ Build transfer matrix
for i = 1 to N do
    for j = 1 to N do
        T(i,j) = 1 + W(i,j) − Sum(W(:,i))        ▷ i^th column sum
    end
end
m = matMul(T, m)                          ▷ Conduct mass transfers
```

3.2.2. Semi-analytical solution for mass-transfer weight function

In order to formulate our semi-analytical solution to the problem of a discontinuous diffusion coefficient, we take a predictor-corrector approach, much like that described in Section 3.1 (LaBolle et al., 2000). We consider the same 1D problem setup outlined in Section 3.2.1; however, for $x_0 \geq \gamma$, our semi-analytical solution shall have the form

$$C_S(t,x) = C_r(x;D_1,\Delta t) I_{(-\infty,x_c]}(x) + C_k(x;D_2,\Delta t) I_{\Omega_2}(x), \tag{20}$$

where the subscript k stands for "keep" because this represents the amount of solute that is kept in the domain where it started (and is distributed according to a diffusion coefficient of D_2), and the subscript r stands for "redistribute" because this represents the mass that is redistributed according to a diffusion coefficient of D_1, and x_c is some "corrected" x-value that alters the support of the C_r solution so that (20) conserves mass. Also, we make the distinction that C_k and C_r are parameterized by the necessarily small time step, Δt, rather than being functions of t, because this solution is only valid for short time. In (20), we define

$$C_k(x;D_2,\Delta t) := \frac{1}{\sqrt{4\pi D_2 \Delta t}} \exp\left[-\frac{|x-x_0|^2}{4D_2\Delta t}\right], \tag{21}$$

$$C_r(x;D_1,\Delta t) := \frac{1}{\sqrt{4\pi D_1 \Delta t}} \exp\left[-\frac{|x-x_0|^2}{4D_1\Delta t}\right]. \tag{22}$$

Integrating each of these expressions over their respective support, in order to compute the total mass in each branch of the total solution, gives

$$m_k = \int_{\gamma}^{\infty} C_k(x)dx = \frac{1}{2}\left[1 - \operatorname{erf}\left(\frac{\gamma - x_0}{\sqrt{4D_2\Delta t}}\right)\right], \tag{23}$$

$$m_r = \int_{-\infty}^{x_c} C_r(x)dx = \frac{1}{2}\left[1 - \operatorname{erf}\left(\frac{x_0 - x_c}{\sqrt{4D_1\Delta t}}\right)\right], \tag{24}$$

where $\operatorname{erf}(\cdot)$ is the error function. Setting $m_{\text{total}} = 1 = m_k + m_r$ and solving for x_c yields

$$x_c = x_0 - (x_0 - \gamma)\sqrt{\frac{D_1}{D_2}}, \tag{25}$$

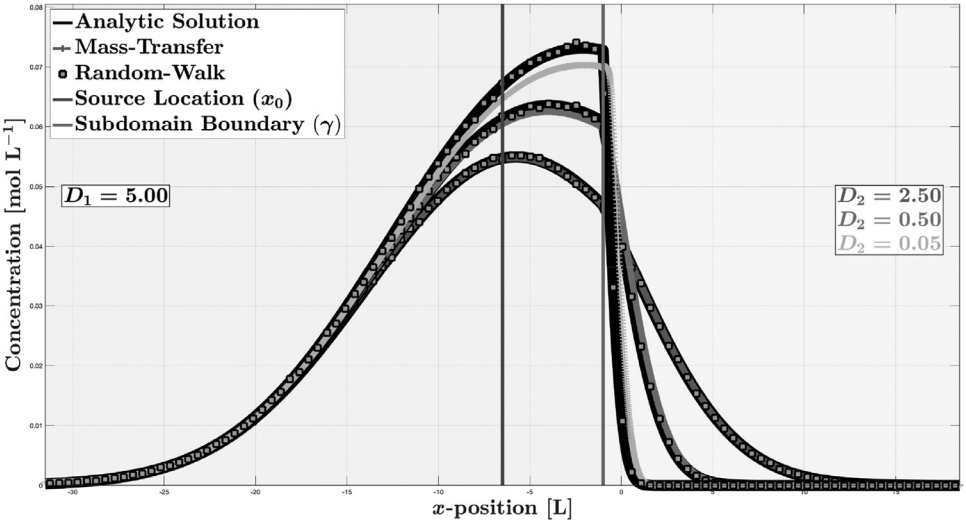

Fig. 2. 1D purely-diffusive simulation for two subdomains with diffusion coefficients D_1 and D_2 (shown for 3 different values of D_2). The MTPT method employs the semi-analytical solution given in (28) using Algorithm 1, and is compared to the predictor-corrector RWPT method of LaBolle et al. (2000) and the analytical solution given in Section 3.2.1. Results are shown for a simulation with 5000 MT particles, 10^6 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 6$. All dimensioned quantities are unitless. Note the oscillation that occurs near the subdomain boundary ($x = \gamma$) for the $D_2 = 0.05$ case (yellow plot). This is attributable to applying Algorithm (1) (symmetric normalization) rather than Algorithm 2 (Sinkhorn-Knopp normalization) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

and we may repeat the calculations above for $x_0 < \gamma$, with the solution

$$\widehat{C}_S(t,x) = C_k(x;D_1,\Delta t)I_{\Omega_1}(x) + C_r(x;D_2,\Delta t)I_{(x_c,\infty)}(x), \tag{26}$$

to find

$$x_c = x_0 - (x_0-\gamma)\sqrt{\frac{D_2}{D_1}}. \tag{27}$$

As in Section 3.2.1, we may combine (20) and (26) into one general solution, namely

$$W_S(t,x) := \widehat{C}_S(\Delta t, x; x_0)I_{\Omega_1}(x_0) + C_S(\Delta t, x; x_0)I_{\Omega_2}(x_0). \tag{28}$$

Unfortunately, if we use W_S as the `WtFunction()` in Algorithm 1, we obtain solutions that display a troubling amount of oscillation near the subdomain boundary (see Fig. 2).

This is because we no longer have a symmetric weight matrix (even numerically), due to W_S lacking symmetry with respect to x and x_0, and, as a result, it also no longer makes sense to apply the symmetric normalization given in (10). In order for the mass-transfer method to both conserve mass and generate solutions with low error, we must make the following changes:

1. We normalize the weight matrix and form $\widehat{\mathcal{W}}$ by employing the Sinkhorn-Knopp (SK) algorithm (Knight, 2008), a computationally-efficient iterative method for obtaining a doubly-stochastic matrix that is mathematically equivalent to alternately normalizing the rows and the columns of a matrix to sum to unity. In order to conserve mass, the columns must be normalized last and must sum to unity with high precision. We find that for all of the cases we considered, 1000 iterations produced satisfactory results.
2. We employ a weight matrix that is the transpose of that used in Algorithm 1; i.e.,

   $$\widehat{W}_{ij} := W_S(\Delta t, x = X_j; x_0 = X_i).$$

   To contrast, note that if we use (28) in Algorithm 1, we have

   $$W_{ij} := W_S(\Delta t, x = X_i; x_0 = X_j).$$

   This is done purely for numerical convenience, as applying the SK algorithm to $\widehat{W}$ converges more reliably to the desired stochastic matrix $\widehat{\mathcal{W}}$ than applying SK to $\boldsymbol{W}$. In fact, starting with $\boldsymbol{W}$ leads to solutions that display a "kink" near the boundary, and much greater resolution in both time and space is required to generate acceptable solutions.

Written in the sum form of (5) and (18), after normalizing $\widehat{W}$ via SK to form $\widehat{\mathcal{W}}$, the above amounts to

$$\begin{aligned} m_i(t+\Delta t) &= m_i(t) + \sum_{j=1}^{N}\widehat{\mathcal{W}}_{ij}m_j(t) - \sum_{j=1}^{N}\widehat{\mathcal{W}}_{ji}m_i(t) \\ &= m_i(t) + \sum_{j=1}^{N}\widehat{\mathcal{W}}_{ij}m_j(t) - m_i(t)\sum_{j=1}^{N}\widehat{\mathcal{W}}_{ji}\,\overset{1}{\nearrow} \\ &= \sum_{j=1}^{N}\widehat{\mathcal{W}}_{ij}m_j(t), \end{aligned} \tag{29}$$

or in matrix-vector form we have

$$\boldsymbol{m}(t+\Delta t) = \widehat{\mathcal{W}}\boldsymbol{m}(t). \tag{30}$$

The algorithm for conducting mass-transfers (within a timestep of length Δt), according to this modified method is presented in pseudocode in Algorithm 2 , in which `WtFunction()` is defined to be (28).

Algorithm 2: Modified Mass-transfer Algorithm for Semi-analytical Weighting Function

Input: Particle positions, $X = \boldsymbol{X}(t)$, and particle masses $m = \boldsymbol{m}(t)$.
Output: Updated particle masses, $m = \boldsymbol{m}(t+\Delta t)$.

```
                                          ▷ Build weight matrix
1  for i = 1 to N do
2  |  for j = 1 to N do
3  |  |  W(i,j) = WtFunction(x_0 = X(i), x = X(j), γ, D_1, D_2, Δt)
4  |  end
5  end
                                          ▷ Normalize weight matrix
6  for i = 1 to normCount do
7  |  W = rowNormalize(W)          ▷ Normalize the rows of W
8  |  W = colNormalize(W)          ▷ Normalize the columns of W
9  end
10 m = matMul(W, m)                       ▷ Conduct mass transfers
```

We note that the normalization, conducted at lines 6–9 in Algorithm 2, is not strictly the SK algorithm, but is instead meant to be demonstrative, rather than computationally efficient.

Extension to 2D

A major advantage of our semi-analytical solution is that it is straightforward to extend to 2D by applying the same strategy as used

in 1D. Let us first consider the case of 2 subdomains that are split by the line $x = \gamma$, $\Omega_1 = \{(-\infty, \gamma] \times \mathbb{R}\}$ and $\Omega_2 = \{(\gamma, \infty) \times \mathbb{R}\}$ with respective constant diffusion coefficients D_1, D_2. The initial condition is again the instantaneous point source $C(t = 0, \boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{x}_0)$, and $\boldsymbol{x}_0 = (x_0, y_0)$. See Fig. 1(b) for a conceptual depiction of this system. In this case, our general solution is nearly identical to the 1D case, namely

$$W_S(t, \boldsymbol{x}) := \widehat{C}_S(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_1}(\boldsymbol{x}_0) + C_S(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_2}(\boldsymbol{x}_0), \tag{31}$$

where C_S and $\widehat{C}_S$ are the same as in (20) and (26), and the form of C_k and C_r are merely altered to contain 2D Gaussian functions; i.e.,

$$C_i^{2D}(\boldsymbol{x}; \mathcal{D}, \Delta t) := \frac{1}{4\pi \mathcal{D} \Delta t} \exp\left[-\frac{\|\boldsymbol{x} - \boldsymbol{x}_0\|^2}{4\mathcal{D}\Delta t}\right], \quad i = k, r, \quad \mathcal{D} = D_1, D_2. \tag{32}$$

The extension to a more complicated subdomain interface is also straightforward. In this case, we consider a 2×2 tiling of 4 subdomains in 2D, and this condition captures the challenges presented by a highly heterogeneous diffusion (velocity) field that is discretized on a grid, perhaps generated by a finite-difference method. Specifically, the challenge is that mass originating in a given quadrant can end up in any or all of the three neighboring quadrants, with the most complicated path being the diagonal one across the origin. For this problem the full domain Ω is split along the lines $x = \gamma_x$ and $y = \gamma_y$. Thus, we have $\Omega_1 = \{(\gamma_x, \infty] \times (\gamma_y, \infty)\}$, $\Omega_2 = \{(-\infty, \gamma_x] \times (\gamma_y, \infty)\}$, $\Omega_3 = \{(-\infty, \gamma_x] \times (-\infty, \gamma_y]\}$, $\Omega_4 = \{(\gamma_x, \infty) \times (-\infty, \gamma_y]\}$ with respective constant diffusion coefficients D_1, D_2, D_3, D_4. Once again, the initial condition has the form $C(t = 0, \boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{x}_0)$, with $\boldsymbol{x}_0 = (x_0, y_0)$. See Fig. 1(c) for a conceptual depiction of this system. The general solution may be written

$$\begin{aligned} W_S(t, \boldsymbol{x}) :&= C_S^1(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_1}(\boldsymbol{x}_0) + C_S^2(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_2}(\boldsymbol{x}_0) \\ &\quad + C_S^3(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_3}(\boldsymbol{x}_0) + C_S^4(\Delta t, \boldsymbol{x}; \boldsymbol{x}_0) I_{\Omega_4}(\boldsymbol{x}_0). \end{aligned} \tag{33}$$

Above, the portion of the solution corresponding to $\boldsymbol{x}_0 \in \Omega_1$ is composed of the sum of four local solutions with the form of (32), namely

$$\begin{aligned} C_S^1(t, \boldsymbol{x}) :&= C_k^{2D}(\boldsymbol{x}; D_1, \Delta t) I_{\Omega_1}(\boldsymbol{x}) \\ &\quad + C_r^{2D}(\boldsymbol{x}; D_2, \Delta t) I_{(-\infty, x_c^{12}] \times (\gamma_y, \infty)}(\boldsymbol{x}) \\ &\quad + C_r^{2D}(\boldsymbol{x}; D_3, \Delta t) I_{(-\infty, x_c^{13}) \times (-\infty, y_c^{13})}(\boldsymbol{x}) \\ &\quad + C_r^{2D}(\boldsymbol{x}; D_4, \Delta t) I_{(\gamma_x, \infty) \times (-\infty, y_c^{14}]}(\boldsymbol{x}), \end{aligned} \tag{34}$$

where x_c^{ij} and y_c^{ij} are the x and/or y corrections for the mass-transfers from subdomain Ω_i to Ω_j and are calculated so as to ensure conservation of mass. Similar to the 1D problem, we have

$$\begin{aligned} x_c^{12} &= x_0 - (x_0 - \gamma_x)\sqrt{\tfrac{D_2}{D_1}}, \\ x_c^{13} &= x_0 - (x_0 - \gamma_x)\sqrt{\tfrac{D_3}{D_1}}, \\ y_c^{13} &= y_0 - (y_0 - \gamma_y)\sqrt{\tfrac{D_3}{D_1}}, \\ y_c^{14} &= y_0 - (y_0 - \gamma_y)\sqrt{\tfrac{D_4}{D_1}}, \end{aligned} \tag{35}$$

and the calculations are analogous for the portions of (33) corresponding to the other subdomains.

As in the 1D case, the solutions given in (31) and (33) do not conserve mass if they are used as the `WtFunction()` in Algorithm 1, but they do conserve mass and generate minimal error if they are used in Algorithm 2.

4. Results

In this section, we consider the results of applying the MTPT algorithm described in Section 3.2 to solve a series of increasingly-complicated test problems involving discontinuous $D(\boldsymbol{x})$. To do this we constrain our tests to only the mass-transfer (MT) portion of the MTPT algorithm (i.e., stationary particles that do not random-walk), and we compare the results of our simulation to known solutions. In the simple 1D case of 2 subdomains, we compare our MTPT results to an analytical solution, and in all other cases, we use the established RWPT predictor-corrector method of LaBolle et al. (2000) as our baseline for comparison.

We note that the idealized case of stationary, evenly-spaced particles we consider does not appear to bear much resemblance to an actual Lagrangian, or particle-tracking, method in which particles are stochastically positioned due to random walk diffusion. However, even when particles random-walk, the MT algorithm is fully deterministic within each timestep, and, in fact, is conceptually a finite difference scheme with a stochastic stencil. In previous work, the evenly-spaced, stationary condition is shown to bear more similarity to the random-walking particle case than it does to a randomly-spaced, stationary condition (Sole-Mari et al., 2019b). The reason for this is that when particles random-walk, they are "on average" equally-spaced at any given time; whereas, randomly-spaced, stationary particles inevitably contain persistent gaps between particles that degrade solution accuracy. As a result, in order to isolate the performance of the MT algorithm and analyze its accuracy, we choose to simulate the algorithm on evenly-spaced, stationary particles.

From an algorithmic standpoint, we generate the MTPT results according to Algorithm 2, and we use the appropriate semi-analytical solution as `WtFunction()`. For the MTPT case, we model the initial condition by assigning the mass corresponding to unit mass to the particle located at $\boldsymbol{x}_0$, and in the RWPT case, we place all particles at location $\boldsymbol{x}_0$, each with mass $1/N$. We then simulate a purely diffusive system with discontinuous $D(\boldsymbol{x})$ up to final time T. For MTPT, constructing the numerical solution at final time is as simple as plotting the concentration on each particle versus its position; however, in the case of RWPT, particles must first be binned to construct concentrations (equal length in 1D and equal-area squares in 2D), and the number of bins was chosen in each case so as to balance between low resolution and noisiness. Lastly, for simplicity, all dimensioned quantities are unitless.

All numerical simulations were conducted in MATLAB, using a MacBook Pro with a 2.9 GHz Intel Core i5 processor and 8 GB of RAM. The code used to generate the results in this section is available at http://doi.org/10.5281/zenodo.3706926 (Schmidt, 2020).

4.1. 1D Results

We begin with the simplest case of a 1D domain with two subdomains, as described in Sections 3.2.1 and 3.2.2, and we hold $D_1 = 5.0$ constant while we test 3 values of D_2 ranging from half the magnitude of D_1 to two orders of magnitude smaller. In the simulations, we employ 5000 particles for the MTPT simulations and 1 million particles in the RWPT simulations (grouped into 100 bins for plotting). We choose a timestep of length $\Delta t = 10^{-2}$ with a total simulation time $T = 6$.

We first examine what occurs when we apply the original MTPT method that our proposed algorithm is based upon (i.e., using (7) as the weighting function in (9)) (Benson and Bolster, 2016; Schmidt et al., 2018; Benson et al., 2020). These results are shown in Fig. 3, and we see that the original MT algorithm holds up for a small magnitude difference in the diffusion coefficient, as when $(D_1, D_2) = (5, 2.5)$. However, when the disparity becomes larger, the accuracy deteriorates, and the MTPT solution is quite poor for $(D_1, D_2) = (5, 0.05)$, as compared to the analytical solution and the RWPT results. The results of applying our new MT algorithm (i.e., the semi-analytical solution given in (28) used within Algorithm (2)) are depicted in Fig. 4. Comparing MTPT results both to the analytical solution, given in (17), and the RWPT results, we see very close agreement between all solutions, indicating that our proposed approach is successful here and that we may move on to more complicated cases.

The next experiment we conduct focuses on a 1D problem with three subdomains, Ω_1, Ω_2, and Ω_3, with their own respective diffusion coef-

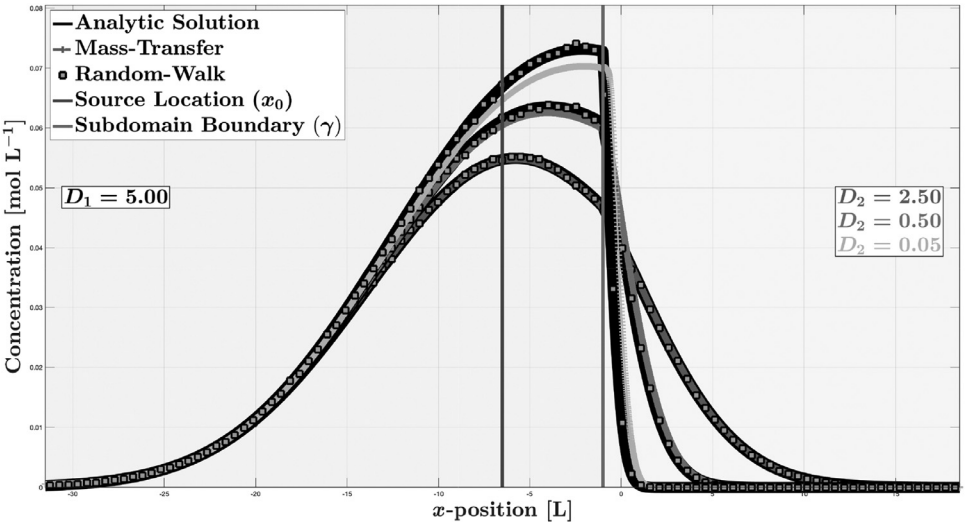

Fig. 3. Results for a 1D purely-diffusive simulation for two subdomains with diffusion coefficients D_1 and D_2 (shown for 3 different values of D_2). The MTPT method employs the original MTPT algorithm on which we base our work (Benson and Bolster, 2016; Schmidt et al., 2018; Benson et al., 2020), as compared to the predictor-corrector RWPT method of LaBolle et al. (2000) and the analytical solution given in Section 3.2.1. RW particles are grouped into 100 bins for plotting. Results are shown for a simulation with 5000 MT particles, 10^6 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 6$. All dimensioned quantities are unitless. Note that the original MTPT algorithm performs quite poorly when there is a large disparity between D_1 and D_2.

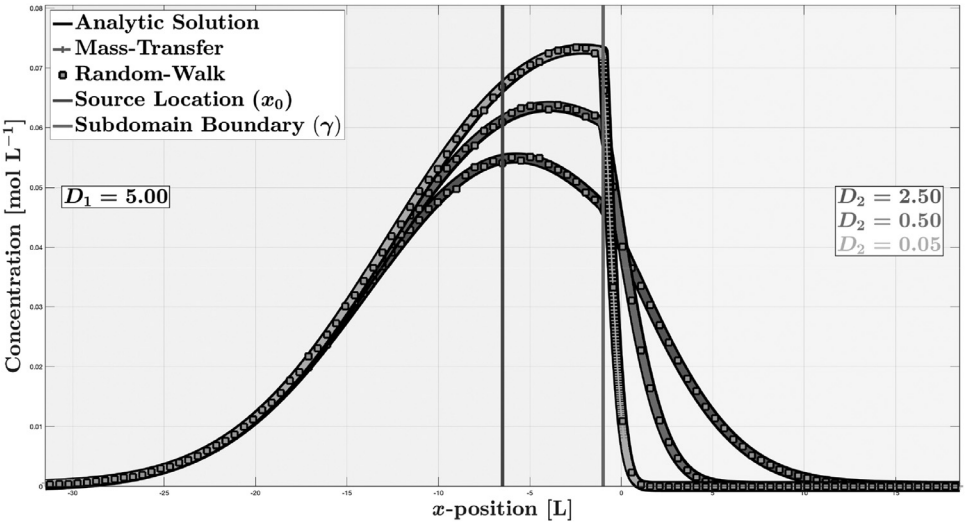

Fig. 4. Results for a 1D purely-diffusive simulation for two subdomains with diffusion coefficients D_1 and D_2 (shown for 3 different values of D_2). The MTPT method employs the semi-analytical solution given in (28) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000) and the analytical solution given in Section 3.2.1. RW particles are grouped into 100 bins for plotting. Results are shown for a simulation with 5000 MT particles, 10^6 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 6$. All dimensioned quantities are unitless.

ficients, representing diffusion in, for example, a layered system. We hold $D_1 = 5.0$ and $D_3 = 0.05$, so as to span two orders of magnitude, and we test three values of $D_2 \in \{2.5, 1.0, 0.5\}$ in the central subdomain. In the simulations, we employ 5000 particles for the MTPT simulations and 1 million particles in the RWPT simulations (grouped into 100 bins for plotting), and we choose a timestep of length $\Delta t = 10^{-2}$ with a total simulation time of $T = 6$. The results of this experiment are displayed in Fig. 5. Because this problem has no simple analytical solution, we take the RWPT results as our baseline case and find very close agreement of the MTPT results with the baseline.

4.2. 2D results

Moving to 2D, we first consider the case of 2 subdomains split along the line $x = \gamma$, corresponding to the semi-analytical solution given in (31). For these simulations, we hold $D_1 = 5.0$ and test $D_2 \in \{2.5, 1.0, 0.5\}$. In the simulations, we employ 10201 particles for the MTPT simulations (101×101 equally-spaced particles, with the number chosen so as to capture the integer-valued source location) and 10 million particles in the RWPT simulations (grouped into 6400 bins for plotting) and choose a timestep of length $\Delta t = 10^{-1}$ with a total simulation time of $T = 6$. The results of this experiment are shown in Figs. 6 and 7. In Fig. 6, we see good visual agreement of the MTPT solutions to the RWPT baseline, and this is verified by plotting the constant-concentration contours on the same axes in Fig. 7 where the match is seen to be nearly exact, aside from the slight noise induced by the randomness in the RWPT simulation.

The next problem we consider is the 2D example of 4 subdomains split along the lines $x = \gamma_x$ and $y = \gamma_y$, corresponding to the semi-analytical solution given in (33). For these simulations, the four cases we consider, in terms of choices for D_i, $i = 1, \ldots, 4$, are: (1) 4 different values for D_i, spanning an order of magnitude; and 3 equal values for D_i and one value that is an order of magnitude smaller, with (2) source location in a subdomain laterally adjacent to the small value of D_i, (3) source location in the subdomain containing the small value value of D_i, and (4) source location in a subdomain diagonally adjacent to the small value of D_i. Of these four cases, case (4) is the least interesting, as the majority of solute remains in the three subdomains with large D_i, so we do not depict results of this simulation, though they were always favorable. In the simulations, we employ $40,401$ particles for the MTPT simulations (201×201 equally-spaced particles) and 10 million particles in the RWPT simulations (grouped into 6400 bins for plotting), and we choose a timestep of length $\Delta t = 10^{-1}$ for the MTPT simulations and $\Delta t = 10^{-2}$ for the RWPT simulations (this was required to generate smooth enough results for comparison), with a total simulation time of $T = 3$.

The results of this experiment are shown in Figs. 8 and 9. In Fig. 8, we see favorable visual agreement of the MTPT solutions to the RWPT baseline, and this is confirmed by the overlaid constant-concentration

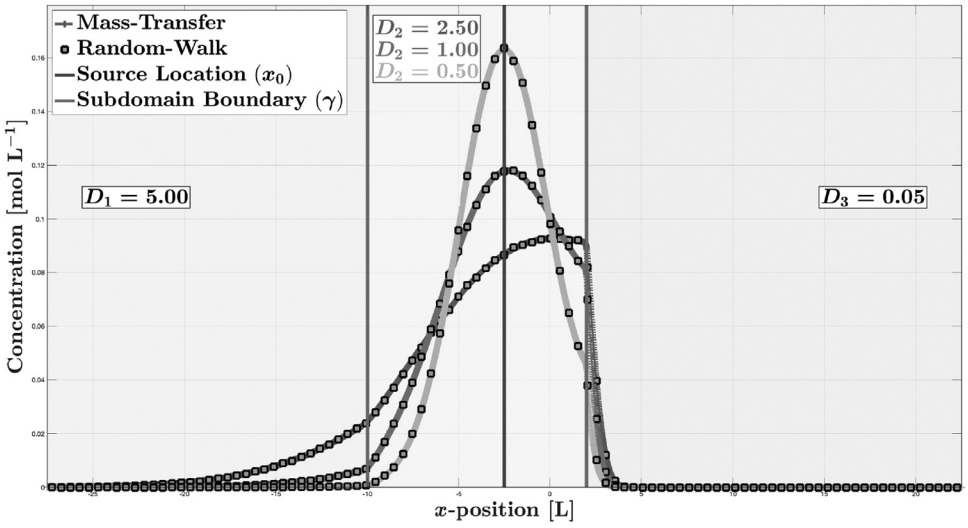

Fig. 5. Results for a 1D purely-diffusive simulation for three subdomains with diffusion coefficients D_1, D_2, and D_3 (shown for 3 different values of D_2). The MTPT method employs the semi-analytical solution given in (28) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000). RW particles are grouped into 100 bins for plotting. Results are shown for a simulation with 5000 MT particles, 10^6 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 6$. All dimensioned quantities are unitless.

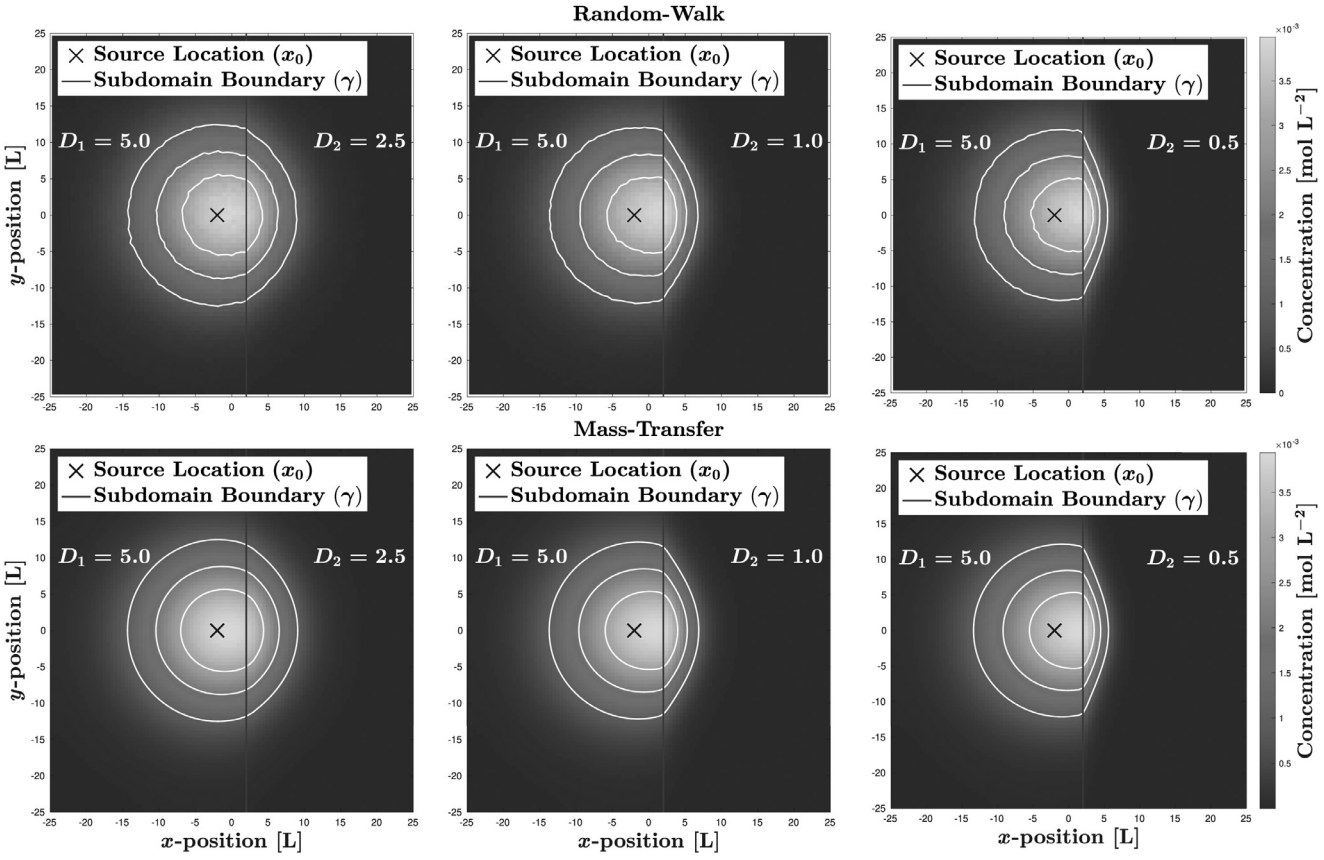

Fig. 6. Concentration heatmap (magnitude given by the color bar on the righthand side) with constant-concentration contours (white curves) depicting results of a 2D simulation for two subdomains with diffusion coefficients D_1 and D_2 (shown for 3 different values of D_2). The MTPT method employs the semi-analytical solution given in (31) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000). RW particles are grouped into 6400 bins for plotting. Results are shown for a simulation with 10201 MT particles, 10^7 RW particles, $\Delta t = 10^{-1}$, and total simulation time $T = 6$. All dimensioned quantities are unitless.

contour plots depicted in Fig. 9. We note that in the 2D experiments, we only consider a single order or magnitude difference between diffusion coefficients. This was in favor of fast run times, as the required number of particles for a MTPT simulation is dictated by the inter-particle spacing, which must be on the order of $\ell := \sqrt{2\widetilde{D}\Delta t}$, where $\widetilde{D}$ is the smallest diffusion coefficient in the system. However, there are no theoretical barriers to considering larger disparities in $D(\boldsymbol{x})$.

4.3. Speed and accuracy

Here, we address two measures of algorithmic performance for our proposed MTPT method for discontinuous $D(\boldsymbol{x})$. First, as to speed, run times for the MTPT method are consistently lower than those for corresponding RWPT solutions. For example, to generate the 1D, 2 subdomain results discussed in Section 4.1 and depicted in Fig. 4, the MTPT simulations run approximately 4.5 times faster than the RWPT simu-

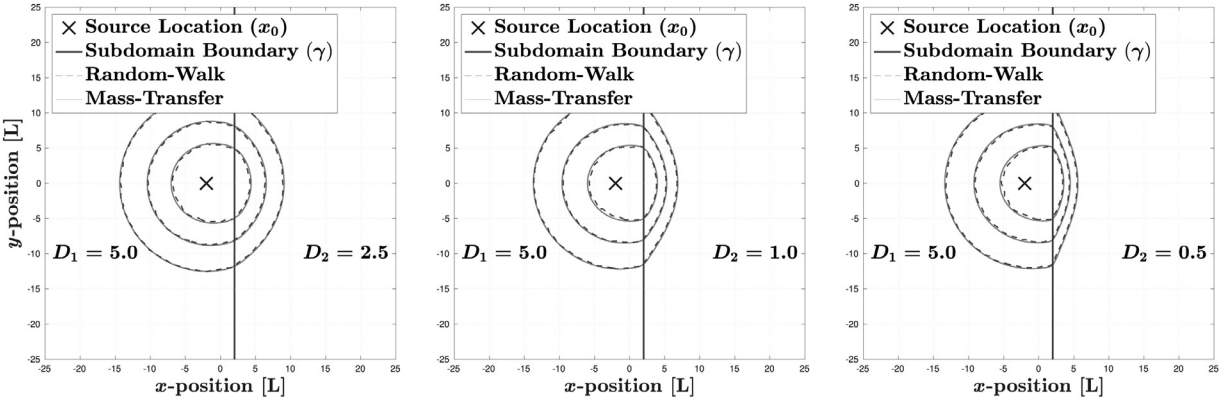

Fig. 7. Constant-concentration contours comparing results depicting results of a 2D simulation for two subdomains with diffusion coefficients D_1 and D_2 (shown for 3 different values of D_2). The MTPT method employs the semi-analytical solution given in (31) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000). RW particles are grouped into 6400 bins for plotting. Results are shown for a simulation with 10201 MT particles, 10^7 RW particles, $\Delta t = 10^{-1}$, and total simulation time $T = 6$. All dimensioned quantities are unitless.

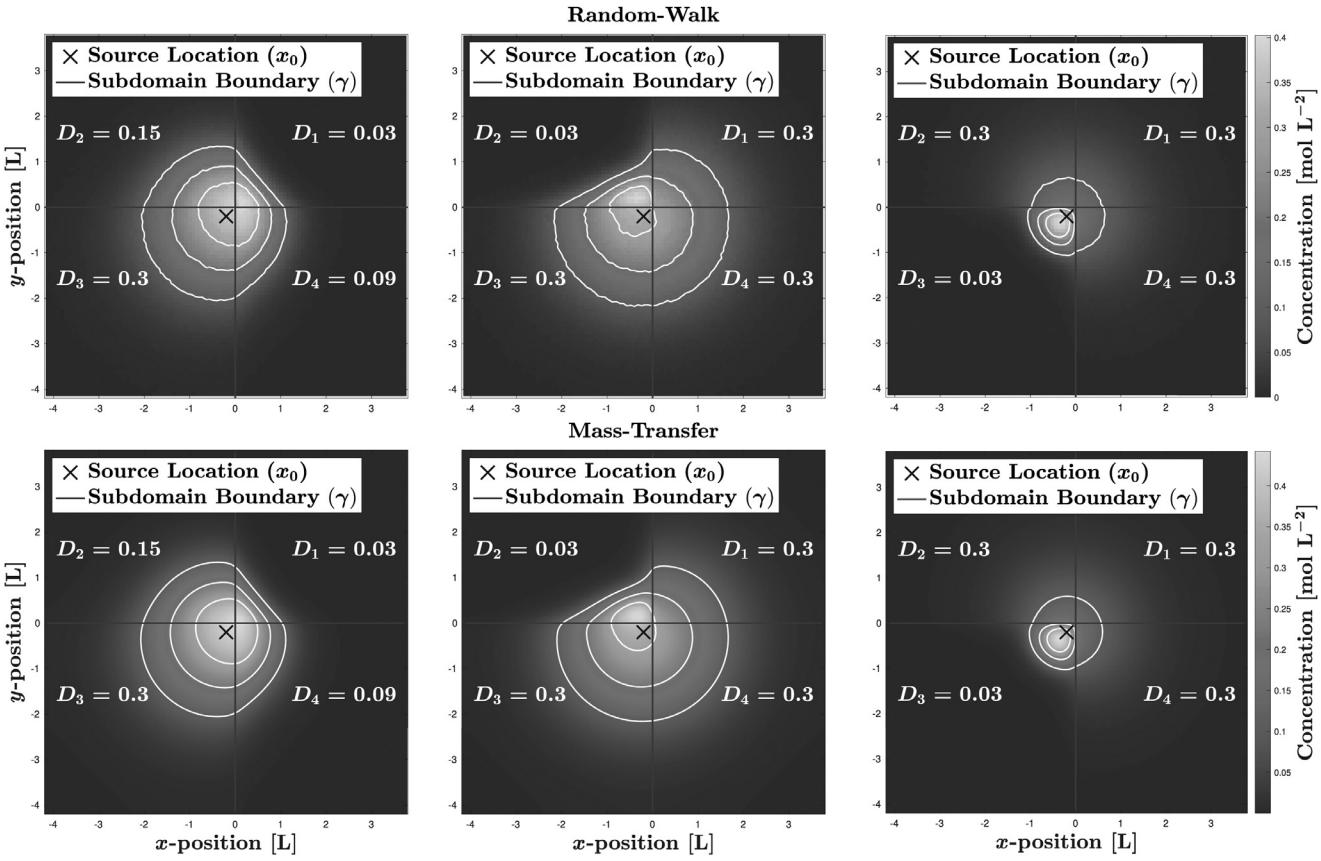

Fig. 8. Concentration heatmap (magnitude given by the color bar on the righthand side) with constant-concentration contours (white curves) depicting results of a 2D simulation for four subdomains with diffusion coefficients D_1, D_2, D_3, and D_4. The MTPT method employs the semi-analytical solution given in (33) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000). RW particles are grouped into 10201 bins for plotting. Results are shown for a simulation with 40401 MT particles, 10^7 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 3$. All dimensioned quantities are unitless.

lations to which the solutions are compared. For the 2D, 4 subdomain case, discussed in Section 4.2 and depicted in Figs. 6 and 9, the MTPT simulations run approximately 1.5 times faster than the RWPT simulations. This speedup for MTPT can primarily be attributed to the fact that mass-transfer interactions only occur among nearest-neighbors, and this allows for speedup via sparse linear algebra. Note, however, that these run time comparisons are for reference only, as both algorithms can be optimized in various ways, and that was not the goal of this work.

As to accuracy, we perform a convergence analysis for the proposed MTPT algorithm to see how error is affected by the level of discretization; i.e., refinements in time step length, Δt, or increase in particle number, N. This convergence analysis considers the 1D, 2 subdomain case, and we compute error in comparison to the analytical solution given in Section 3.2.1. For each convergence analysis we employ all of the same parameters as were used to generate the results in Section 4.1 and depicted in Fig. 4, varying only Δt or N in successive refinements. The results for a convergence analysis in terms of Δt are depicted in Fig. 10,

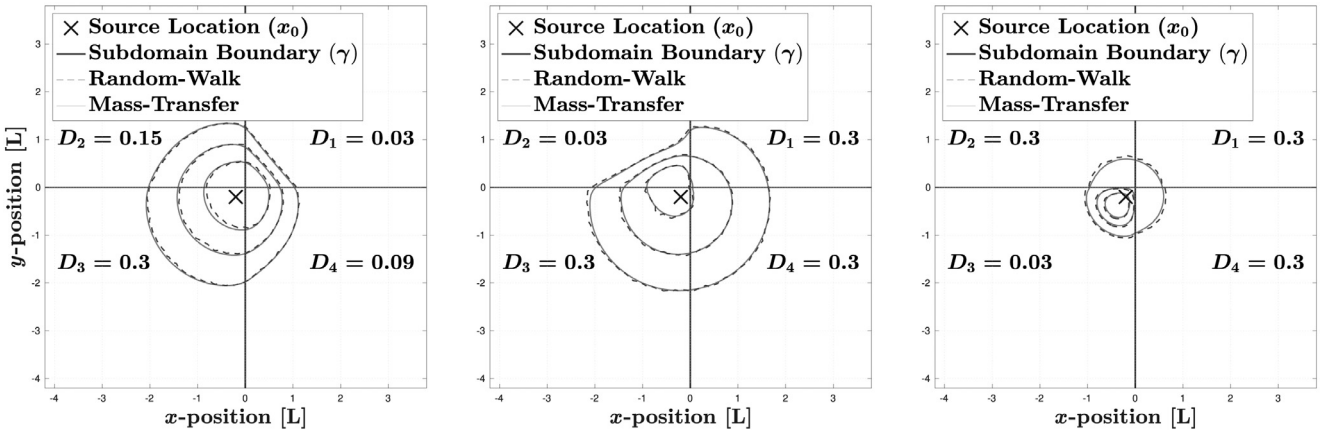

Fig. 9. Constant-concentration contours comparing results depicting results of a 2D simulation for four subdomains with diffusion coefficients D_1, D_2, D_3, and D_4. The MTPT method employs the semi-analytical solution given in (33) using Algorithm 2, as compared to the predictor-corrector RWPT method of LaBolle et al. (2000). RW particles are grouped into 10201 bins for plotting. Results are shown for a simulation with 40401 MT particles, 10^7 RW particles, $\Delta t = 10^{-2}$, and total simulation time $T = 3$. All dimensioned quantities are unitless.

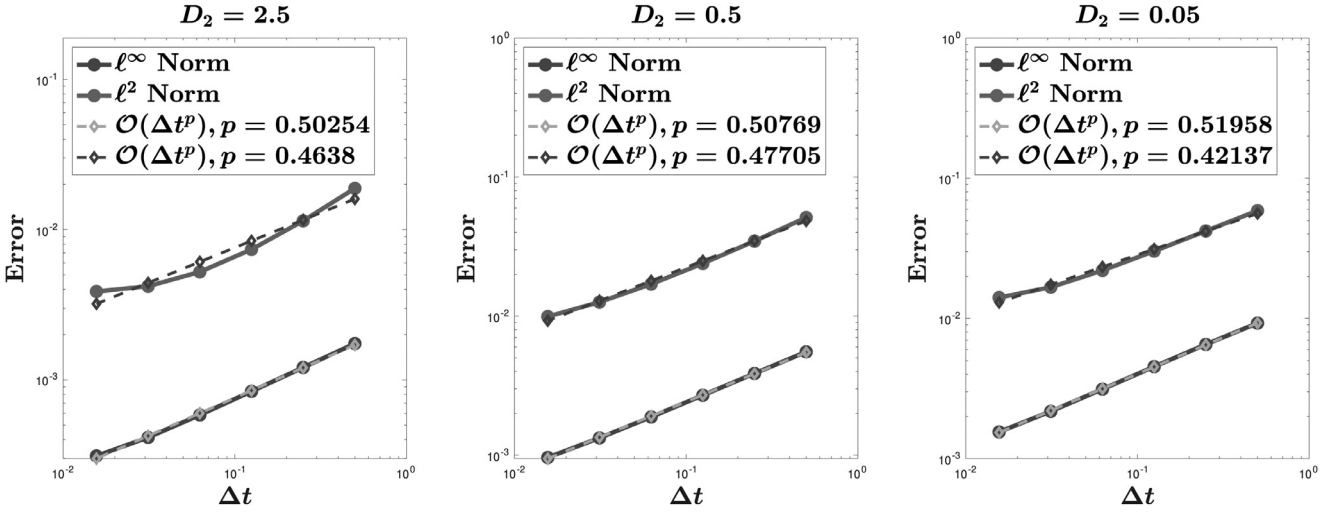

Fig. 10. Convergence analysis in terms of time step length, Δt, for the MTPT Algorithm 2 employing the semi-analytical solution given in (33). These results are for the 1D, 2 subdomain problem for which we have an analytical solution (see Section 4.1 and Fig. 4). Each plot corresponds to a single value for D_2. Error is computed in terms of the ℓ^2 and ℓ^∞ norms and best-fit reference lines are shown to demonstrate the experimental order of convergence. $\mathcal{O}(\Delta t^{1/2})$ convergence appears most clearly in the ℓ^∞ norm.

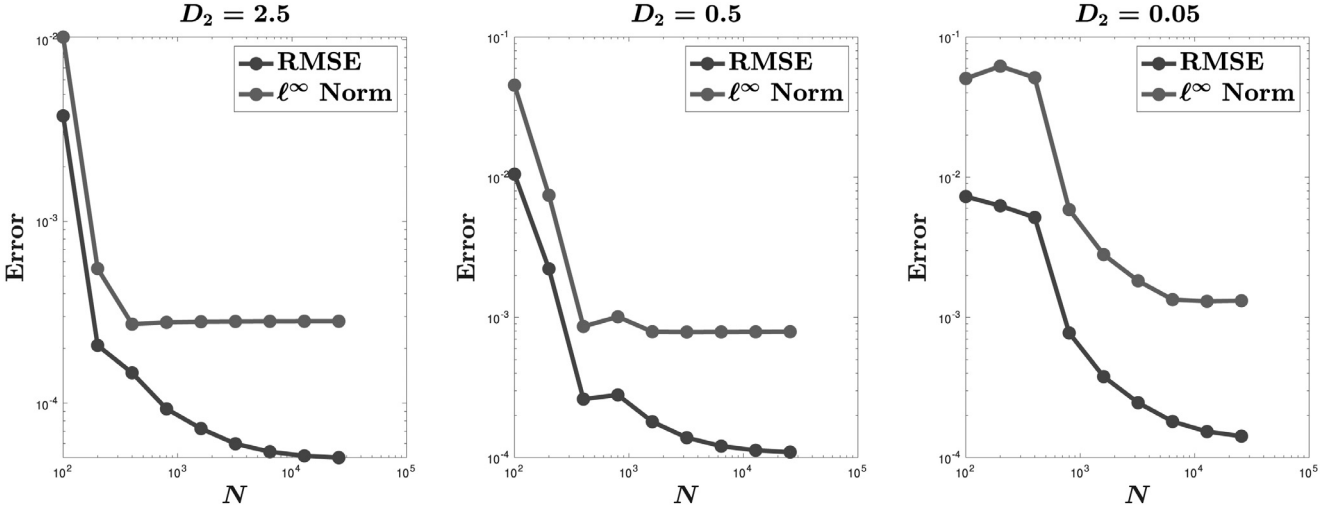

Fig. 11. Convergence analysis in terms of particle number, N, for the MTPT Algorithm 2 employing the semi-analytical solution given in (33). These results are for the 1D, 2 subdomain problem for which we have an analytical solution (see Section 4.1 and Fig. 4). Each plot corresponds to a single value for D_2. Error is computed in terms of the root-mean-squared error (RMSE) and ℓ^∞ norms. Depending on the value of D_2 being considered, rapid convergence is seen with increasing N, until leveling off at a minimal level.

and therein we plot error, in terms of the ℓ^∞ and ℓ^2 norms, as a function of Δt for each of the three values of D_2 we consider in Section 4.1 (Fig. 4). For each of the error curves we also plot a reference $\mathcal{O}(\Delta t^p)$ line of best fit to obtain the order of convergence, p, and we see the general trend of what appears to be $p = 1/2$ order of convergence, and this is demonstrated most clearly in the ℓ^∞ norm. The results for a convergence analysis in terms of N are depicted in Fig. 11, and therein we plot error, in terms of the ℓ^∞ norm and root-mean-squared error (RMSE), as a function of N for each of the three values of D_2 we consider in Section 4.1 (Fig. 4). We note that we employ RMSE here, as opposed to the ℓ^2 norm, to normalize for varying vector-length. In this case, we see a period of rapid convergence with increasing N, before error levels off to a minimal level that is controlled by the time discretization, and this is the expected behavior that is commonly seen in MTPT methods (Schmidt et al., 2018; 2019a).

5. Conclusions

Discontinuous diffusion coefficients arise naturally within simulations of transport through heterogeneous porous media, but accurately modeling diffusion across these interfaces has remained an outstanding problem for MTPT algorithms. Here, we have generalized MTPT algorithms to addresses this deficiency, including for multi-dimensional systems. This is a significant advance both from a numerical perspective and in terms of improving the realism of such simulations. Additionally, these results serve to eliminate one of the few remaining barriers that limit the capabilities of Lagrangian methods in comparison to their Eulerian counterparts.

In particular, within the current work, we have:

1. generalized the MT algorithm to incorporate non-symmetric mass-transfer kernels;
2. presented an MT algorithm that employs a relatively simple 1D analytic solution to the discontinuous $D(x)$ problem;
3. derived a semi-analytical solution to the discontinuous $D(\boldsymbol{x})$ problem that is straightforward to generalize to higher dimensions and complicated subdomain interfaces;
4. presented an MT algorithm that incorporates this semi-analytical solution;
5. applied this updated MTPT algorithm to a variety of test cases, including a 2D problem that corresponds to a standard velocity grid with order-of-magnitude differences in $D(\boldsymbol{x})$;
6. attained favorable results of this application of the new MTPT algorithm.

Additionally, while not considered in this work, it would be a simple matter to handle moving subdomain interfaces with this algorithm. This is because particle interactions occur pairwise, and to make the relevant mass-transfer, the only required information is each particle's mass, position, and local diffusion coefficient, which are easy enough to establish within a timestep, no matter the current orientation of a subdomain boundary.

Open questions remain in this direction, however. For instance, what would be the effect of running a hybrid version of MTPT including diffusive random walks in the algorithm, and how would it affect the accuracy of solutions? Or, how might the solution be generalized to subdomains that possess more complicated geometry; for example, boundaries that are not right angles, such as on a triangulated grid, or boundaries that are not straight lines at all (e.g. Sole-Mari et al., 2019a). Additionally, we have only considered the scalar, or isotropic, $D(\boldsymbol{x})$ case because it is common in the MTPT literature to simulate large-scale, anisotropic spreading by random walks and the micro-scale, isotropic mixing process by mass transfers (Schmidt et al., 2018; Benson et al., 2019a).

In summary, we have extended the capabilities of MTPT methods to solve the problem of discontinuous diffusion coefficients, thus adding flexibility to a tool that already is able to: model arbitrarily complex reactions, including fluid-solid interactions; separately simulate macro-scale spreading and micro-scale mixing; capture arbitrarily fine resolution in mixing and concentration gradients; and achieve nearly linear speedup when parallelized.

Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Michael J. Schmidt: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Nicholas B. Engdahl:** Conceptualization, Methodology, Software, Writing - original draft, Writing - review & editing. **Stephen D. Pankavich:** Formal analysis, Writing - original draft, Writing - review & editing. **Diogo Bolster:** Methodology, Formal analysis, Writing - original draft, Writing - review & editing, Supervision, Funding acquisition.

Acknowledgments

We thank the editor and reviewers for their insightful and helpful comments. The first author would like to thank Paul Martin for his assistance in exploring analytical solutions to the 2D problems considered in this work. This work was supported by the US Army Research Office under contract/grant number W911NF-18-1-0338; the National Science Foundation under awards EAR-1417145 and DMS-1614586; and the DOE Office of Science under award DE-SC0019123.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.advwatres.2020.103577.

References

Appuhamillage, T.A., Bokil, V.A., Thomann, E., Waymire, E., Wood, B.D., 2010. Solute transport across an interface: A Fickian theory for skewness in breakthrough curves. Water Resources Res. 46 (7). https://doi.org/10.1029/2009WR008258.

Bagtzoglou, A.C., Tompson, A.F.B., Dougherty, D.E., 1992. Projection functions for particle-grid methods. Numer. Methods Partial Differ. Equ. 8 (4), 325–340. https://doi.org/10.1002/num.1690080403.

Bechtold, M., Vanderborght, J., Ippisch, O., Vereecken, H., 2011. Efficient random walk particle tracking algorithm for advective-dispersive transport in media with discontinuous dispersion coefficients and water contents. Water Resour. Res. 47 (10). https://doi.org/10.1029/2010WR010267.

Benson, D.A., Aquino, T., Bolster, D., Engdahl, N., Henri, C.V., Fernàndez-Garcia, D., 2017. A comparison of Eulerian and Lagrangian transport and non-linear reaction algorithms. Adv. Water Resour. 99, 15–37. https://doi.org/10.1016/j.advwatres.2016.11.003.

Benson, D.A., Bolster, D., 2016. Arbitrarily complex chemical reactions on particles. Water Resour. Res. 52 (11), 9190–9200. https://doi.org/10.1002/2016WR019368.

Benson, D.A., Meerschaert, M.M., 2008. Simulation of chemical reaction via particle tracking: Diffusion-limited versus thermodynamic rate-limited regimes. Water Resour. Res. 44, W12201. https://doi.org/10.1029/2008WR007111.

Benson, D.A., Pankavich, S., Bolster, D., 2019. On the separate treatment of mixing and spreading by the reactive-particle-tracking algorithm: An example of accurate upscaling of reactive poiseuille flow. Adv. Water Resour. 123, 40–53. https://doi.org/10.1016/j.advwatres.2018.11.001. URL: http://www.sciencedirect.com/science/article/pii/S0309170818304354

Benson, D.A., Pankavich, S., Schmidt, M.J., Sole-Mari, G., 2020. Entropy: 1) the former trouble with particle-tracking simulation, and 2) a measure of computational information penalty. Adv. Water Resour. 103509. https://doi.org/10.1016/j.advwatres.2020.103509. URL: http://www.sciencedirect.com/science/article/pii/S0309170819303458

Benson, D.A., Schmidt, M.J., Bolster, D., Harmon, C., Engdahl, N.B., 2019. Aging and mixing as pseudo-chemical-reactions between, and on, particles: Perspectives on particle interaction and multi-modal ages in hillslopes and streams. Adv. Water Resour. 103386. https://doi.org/10.1016/j.advwatres.2019.103386. URL: http://www.sciencedirect.com/science/article/pii/S0309170819303951

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

Carslaw, H.S., Jaeger, J.C., 1959. Conduction of Heat in Solids, 2nd ed. Clarendon, Oxford.

Engdahl, N.B., Benson, D.A., Bolster, D., 2017. Lagrangian simulation of mixing and reactions in complex geochemical systems. Water Resour. Res. 53 (4), 3513–3522. https://doi.org/10.1002/2017WR020362.

Engdahl, N.B., Schmidt, M.J., Benson, D.A., 2019. Accelerating and parallelizing Lagrangian simulations of mixing-limited reactive transport. Water Resour. Res. 55.

Gingold, R.A., Monaghan, J.J., 1977. Smoothed particle hydrodynamics: theory and application to non-spherical stars. Monthly Notices R. Astronom. Soc. 181 (3), 375–389. https://doi.org/10.1093/mnras/181.3.375.

Herrera, P.A., Beckie, R.D., 2013. An assessment of particle methods for approximating anisotropic dispersion. Int. J. Numer. Methods Fluids 71 (5), 634–651. https://doi.org/10.1002/fld.3676.

Herrera, P.A., Massabó, M., Beckie, R.D., 2009. A meshless method to simulate solute transport in heterogeneous porous media. Adv. Water Resour. 32 (3), 413–429. https://doi.org/10.1016/j.advwatres.2008.12.005. URL: http://www.sciencedirect.com/science/article/pii/S0309170808002273

Hoteit, H., Mose, R., Younes, A., Lehmann, F., Ackerer, P., 2002. Three-dimensional modeling of mass transfer in porous media using the mixed hybrid finite elements and the random-walk methods. Math. Geol. 34 (4), 435–456. https://doi.org/10.1023/A:1015083111971.

Knight, P., 2008. The Sinkhorn–Knopp algorithm: Convergence and applications. SIAM J. Matrix Anal. Appl. 30 (1), 261–275. https://doi.org/10.1137/060659624.

LaBolle, E.M., Fogg, G.E., Tompson, A.F.B., 1996. Random-walk simulation of transport in heterogeneous porous media: Local mass-conservation problem and implementation methods. Water Resour. Res. 32 (3), 583–593. https://doi.org/10.1029/95WR03528.

LaBolle, E.M., Quastel, J., Fogg, G.E., 1998. Diffusion theory for transport in porous media: Transition-probability densities of diffusion processes corresponding to advection-dispersion equations. Water Resour. Res. 34 (7), 1685–1693. https://doi.org/10.1029/98WR00319.

LaBolle, E.M., Quastel, J., Fogg, G.E., Gravner, J., 2000. Diffusion processes in composite porous media and their numerical integration by random walks: Generalized stochastic differential equations with discontinuous coefficients. Water Resour. Res. 36 (3), 651–662.

Lejay, A., Lenôtre, L., Pichot, G., 2019. An exponential timestepping algorithm for diffusion with discontinuous coefficients. J. Comput. Phys. 396, 888–904. https://doi.org/10.1016/j.jcp.2019.07.013. URL: http://www.sciencedirect.com/science/article/pii/S002199911930498X

Lejay, A., Pichot, G., 2016. Simulating diffusion processes in discontinuous media: Benchmark tests. J. Comput. Phys. 314, 384–413. https://doi.org/10.1016/j.jcp.2016.03.003. URL: http://www.sciencedirect.com/science/article/pii/S0021999116001534

Monaghan, J., 2012. Smoothed particle hydrodynamics and its diverse applications. Ann. Rev. Fluid Mech. 44 (1), 323–346. https://doi.org/10.1146/annurev-fluid-120710-101220.

Oukili, H., Ababou, R., Debenest, G., Noetinger, B., 2019. Random walks with negative particles for discontinuous diffusion and porosity. J. Comput. Phys. 396, 687–701. https://doi.org/10.1016/j.jcp.2019.07.006. URL: http://www.sciencedirect.com/science/article/pii/S0021999119304917

Paster, A., Bolster, D., Benson, D.A., 2014. Connecting the dots: Semi-analytical and random walk numerical solutions of the diffusion–reaction equation with stochastic initial conditions. J. Comput. Phys. 263, 91–112. https://doi.org/10.1016/j.jcp.2014.01.020. URL: http://www.sciencedirect.com/science/article/pii/S0021999114000473

Salamon, P., Fernàndez-Garcia, D., Gómez-Hernández, J.J., 2006. A review and numerical assessment of the random walk particle tracking method. J. Contam. Hydrol. 87 (3–4), 277–305. https://doi.org/10.1016/j.jconhyd.2006.05.005. URL: http://www.sciencedirect.com/science/article/pii/S0169772206000957

Schmidt, M. J., 2020. Mschmidt271/MTPT_discontD v1.0. 10.5281/zenodo.3706926

Schmidt, M.J., Pankavich, S., Benson, D.A., 2017. A kernel-based Lagrangian method for imperfectly-mixed chemical reactions. J. Comput. Phys. 336, 288–307. https://doi.org/10.1016/j.jcp.2017.02.012. URL: http://www.sciencedirect.com/science/article/pii/S0021999117301055

Schmidt, M.J., Pankavich, S.D., Benson, D.A., 2018. On the accuracy of simulating mixing by random-walk particle-based mass-transfer algorithms. Adv. Water Resour. 117, 115–119. https://doi.org/10.1016/j.advwatres.2018.05.003. URL: http://www.sciencedirect.com/science/article/pii/S0309170818301830

Schmidt, M.J., Pankavich, S.D., Navarre-Sitchler, A., Benson, D.A., 2019. A Lagrangian method for reactive transport with solid/aqueous chemical phase interaction. J. Comput. Phys. X 100021. https://doi.org/10.1016/j.jcpx.2019.100021. URL: http://www.sciencedirect.com/science/article/pii/S259005521930037X

Schmidt, M.J., Pankavich, S.D., Navarre-Sitchler, A., Engdahl, N.B., Bolster, D., Benson, D.A., 2019. Reactive particle-tracking solutions to a benchmark problem on heavy metal cycling in lake sediments. Submitted. 1908.09818

Semra, K., Ackerer, P., Mosé, R., 1970. Three dimensional groundwater quality modelling in heterogeneous media. WIT Trans. Ecol. Environ. 2.

Shendeleva, M.L., 2001. Reflection and refraction of a transient temperature field at a plane interface using Cagniard–de Hoop approach. Phys. Rev. E 64, 036612. https://doi.org/10.1103/PhysRevE.64.036612.

Sole-Mari, G., Bolster, D., Fernàndez-Garcia, D., Sanchez-Vila, X., 2019. Particle density estimation with grid-projected and boundary-corrected adaptive kernels. Adv. Water Resour. 131, 103382. https://doi.org/10.1016/j.advwatres.2019.103382. URL: http://www.sciencedirect.com/science/article/pii/S0309170819304014

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 Resour. Res. 54, 8309–8331. https://doi.org/10.1029/2018WR023033.

Sole-Mari, G., Fernàndez-Garcia, D., Rodríguez-Escales, P., Sanchez-Vila, X., 2017. A KDE-based random walk method for modeling reactive transport with complex kinetics in porous media. Water Resour. Res. 53 (11), 9019–9039.

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. Adv. Water Resour. https://doi.org/10.1016/j.advwatres.2019.02.009. URL: http://www.sciencedirect.com/science/article/pii/S0309170818310820

Thomson, D.J., 1987. Criteria for the selection of stochastic models of particle trajectories in turbulent flows. J. Fluid Mech. 180, 529–556. https://doi.org/10.1017/S0022112087001940. URL: http://journals.cambridge.org/article_S0022112087001940

Uffink, G., 1983. A random walk method for the simulation of macrodispersion in a stratified aquifer. Relat. Groundw. Quant. Qual. 103–114.