



MAX-PLANCK-GESELLSCHAFT

**Max Planck Institute Magdeburg
Preprints**

Martin Stoll

Hamdullah Yücel

**Symmetric Interior Penalty Galerkin
Method For Fractional-In-Space
Allen-Cahn Equations**



MAX-PLANCK-INSTITUT
FÜR DYNAMIK KOMPLEXER
TECHNISCHER SYSTEME
MAGDEBURG

Abstract

Fractional differential equations are becoming increasingly popular as a modelling tool to describe a wide range of non-classical phenomena with spatial heterogeneities throughout the applied science and engineering. However, the non-local nature of the fractional operators causes essential difficulties and challenges for numerical approximations. We here address an efficient approach to solve fractional-in-space Allen-Cahn equations via the contour integral method (CIM) for computing the fractional power of a matrix times a vector. Time discretization is performed by the first- and second-order implicit-explicit schemes with an adaptive time-step size approach, whereas spatial discretization is performed by a symmetric interior penalty Galerkin (SIPG) method. Several numerical examples are presented to illustrate the effect of the fractional power.

Impressum:

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg

Publisher:

Max Planck Institute for
Dynamics of Complex Technical Systems

Address:

Max Planck Institute for
Dynamics of Complex Technical Systems
Sandtorstr. 1
39106 Magdeburg

www.mpi-magdeburg.mpg.de/preprints

1 Introduction

Fractional models, in which a standard time or space differential operator is replaced by a corresponding fractional operator, have gained considerable popularity and importance during the last few decades, although fractional calculus is an old topic in mathematics, see [22] for historical notes. Fractional calculus is now used to describe a broad range of non-classical phenomena in the applied sciences, engineering, and finance due to the intrinsic non-local property of fractional derivatives, for example, the filtration of solutes in porous soils [8], diffusion of water molecules in brain tissues [12], electrical charge transport in polymer networks [25], the relationship between certain option pricing and heavy-tailed stochastic process [42], anomalous diffusion process for continuous time random walk models [38].

It is well known that the derivation of the analytical solutions to the fractional differential equations is generally difficult and computation of them is very expensive due to infinite series in the analytical solutions. On the other hand, the implementation of numerical approaches to solve the fractional differential equations also has essential difficulties and challenges due to the non-local nature of the fractional operators (space fractional) and the dependence on the full history (time fractional). However, in recent years, a number of successful numerical approaches for the fractional differential equations have been considered such as finite difference methods [27, 36, 44, 9, 48], spectral methods [39, 34], finite element methods [15, 18, 49, 10], and discontinuous Galerkin methods [16, 41]. Many of these approaches have limitations in terms of computational efficiency when two and three spatial dimensions are considered. They either do not scale well or their scalability has not been shown. Recently, Yang et al. in [47] proposed a new approach using a matrix transfer technique with finite difference and finite element methods to solve the time-space fractional diffusion equation in two spatial dimensions with homogeneous Dirichlet boundary conditions. The solution was advanced in time by computing the function of a matrix times a vector by preconditioned Lanczos method. This concept is also considered in [11] using the finite element method in space and a semi-implicit Euler approximation in time. The computation of the fractional power of a matrix times a vector was done by the contour integral method, the extended Krylov subspace method, and the preassigned poles and interpolation nodes method.

We here concern ourselves with the fractional-in-space Allen-Cahn equations of the form

$$u_t + (-\Delta)^\alpha u + \frac{1}{\varepsilon} f(u) = 0 \quad \text{in } \Omega \times (0, T), \quad (1a)$$

$$u = g^D \quad \text{in } \Gamma^D \times (0, T), \quad (1b)$$

$$\frac{\partial u}{\partial n} = g^N \quad \text{in } \Gamma^N \times (0, T), \quad (1c)$$

with initial condition $u(x, 0) = u_0(x)$. The operator $(-\Delta)^\alpha$ denotes the fractional operator of order $\alpha \in (0.5, 1]$.

In problem (1), u represents the concentration of one of the species of the alloy, the parameter ε represents the diffuse interface width parameter, $\Omega \subset \mathbb{R}^d$ ($d = 1, 2, 3$) is a bounded domain with Dirichlet and Neumann boundary conditions denoted by Γ^D and Γ^N , respectively. The nonlinear term $f(u) = F'(u)$ is the derivative of a free energy functional $F(u)$. There are many not just two types of free energy functional $F(u)$ in the literature. The first one we consider is

the non-convex logarithmic free energy [6]

$$F(u) = \frac{\theta}{2} [(1+u)\ln(1+u) + (1-u)\ln(1-u)] - \frac{\theta_c}{2} u^2 \quad (2)$$

with $0 < \theta \leq \theta_c$, where θ_c is the transition temperature. When θ converges to θ_c , the logarithmic free energy (2) can be approximated by a smooth double equal well potential taking its global minimum value 0 at $u = \pm 1$

$$F(u) = \frac{1}{4}(u^2 - 1)^2. \quad (3)$$

Then, $f(u) = u(u^2 - 1)$ represents the bistable non-linearity for the double-well potential, whereas $f(u) = \frac{\theta}{2} \ln\left(\frac{1+u}{1-u}\right) - \theta_c u$ is for the logarithmic free energy (2). Further, the fractional-in-space Allen-Cahn equation can be viewed as the gradient flow of the energy

$$\mathcal{E}(u) = \int_{\Omega} \left(\frac{1}{2} |\nabla u|^{(2\alpha)} + \frac{1}{\varepsilon} F(u) \right) d\Omega. \quad (4)$$

Equation (1) with $\alpha = 1$, which is known as a scaled in time form of the Allen-Cahn equation, was originally introduced by Allen and Cahn in [1] to describe the phase separation process of a binary alloy at a fixed temperature. The Allen-Cahn equations (and the related Cahn-Hilliard equations) are essential building blocks in the phase field methodology or the diffuse interface methodology for moving interface and free boundary problems, see e.g., [5, 37]. There are several challenges to obtain numerical approximations of these problems such as the existence of a nonlinear term and the presence of the small interfacial length parameter ε . An appropriate numerical scheme requires a proper relation between physical and numerical scales, that is, the size of spatial mesh h and time step τ have to properly be related to the interaction length ε . For the spatial discretization, well known methods like spectral methods [14], finite element methods [35], and discontinuous Galerkin methods [20, 31] have been used for the classic Allen-Cahn equation. The resulting system is inherently stiff system due to the small positive parameter ε . This is then handled by appropriate temporal discretizations methods, such as the implicit-explicit (IMEX) techniques [43, 21], and the average vector field (AVF) method [31, 13].

Recently, there has been a fast increasing number of studies on front propagation of reaction diffusion systems with an anomalous diffusion as super diffusion, i.e., the fractional Allen-Cahn equation (1). Such super diffusion is related to Lévy processes and can be modeled by a fractional operator $(-\Delta)^\alpha$ with $0.5 < \alpha \leq 1$. Especially, the fractional models offer insight that traditional approaches do not offer, in case of diffusion in heterogeneous environments. However, there are some subtle issues associated with the interpretation of the fractional Laplace operator; see [46] for further discussions. Ilić et al. in [28] have shown that the fractional Laplace operator $(-\Delta)^\alpha$ has the same interpretation as $(-\Delta)$ in terms of its spectral decomposition for homogeneous boundary conditions. Further, the matrix transfer technique was introduced in [29, 30] to compute the fractional Laplacian by first computing a matrix representation of the Laplace (independent of discretization approach) and then raising it to the fractional order.

We here solve the equation of the form (1) by computing the fractional power of a matrix times a vector. To compute the fractional power, the contour integral method proposed in

[24] is applied. It is expected that the fractional reaction-diffusion models as (1) with smaller fractional order exhibit more heterogeneous environments. In addition, the sharp gradients and singularities emerge locally for the small values of the parameter ε . To handle these difficulties, we apply the symmetric interior penalty Galerkin (SIPG) method as a discontinuous Galerkin method for the spatial discretization. Then, the implicit-explicit (IMEX) methods are applied for the temporal discretization. In order to save computational cost we have addressed an adaptive-time stepping algorithm based on the difference between the first order IMEX method and the second order IMEX method.

The rest of this paper is organized as follows: in the next section, we introduce the symmetric interior penalty Galerkin (SIPG) method as a discontinuous Galerkin discretization. In section 3 we review the contour integral method, which allows us to approximate the fractional Laplacian by a fractional power of a matrix. The implicit-explicit methods are given in section 4 for the temporal discretization. Also, an adaptive-time stepping algorithm is addressed to reduce the computational cost. Finally, in the last section, several numerical examples are presented to show the effect of the fractional power.

2 Symmetric interior penalty Galerkin (SIPG) discretization

In this section, we introduce the symmetric interior penalty Galerkin (SIPG) discretization as a discontinuous Galerkin (DG) method. It is chosen due to the symmetric property of its bilinear form, i.e., $a_h(y, v) = a_h(v, y)$, see e.g., [2].

We begin with the continuous weak formulation of the classical Allen-Cahn equation defined by

$$u_t - \Delta u + \frac{1}{\varepsilon} f(u) = 0 \quad \text{in } \Omega \times (0, T). \quad (5)$$

Then, find $u(t) \in U$ such that

$$(u_t, v) + a(u, v) = \ell(v) \quad \forall v \in V, \quad t \in (0, T], \quad (6a)$$

$$(u(\cdot, 0), v) = (u_0, v) \quad \forall v \in V, \quad (6b)$$

where the space of solutions U , and the space of test functions are defined by

$$U = \{u \in H^1(\Omega) : y|_{\Gamma^D} = g^D\}, \quad V = \{v \in H^1(\Omega) : v|_{\Gamma^D} = 0\},$$

and the (bi)-linear forms are given by

$$a(u, v) = \int_{\Omega} (\nabla u \cdot \nabla v) dx, \quad \text{and} \quad \ell(v) = - \int_{\Omega} \frac{1}{\varepsilon} f(u) v dx + \int_{\Gamma^N} g^N v ds.$$

We assume that the domain Ω is polygonal such that the boundary is exactly represented by boundaries of triangles. We denote $\{\mathcal{T}_h\}_h$ as a family of shape-regular simplicial triangulations of Ω . Each mesh \mathcal{T}_h consists of closed triangles such that $\overline{\Omega} = \bigcup_{K \in \mathcal{T}_h} \overline{K}$ holds. We assume that the mesh is regular in the following sense: for different triangles $K_i, K_j \in \mathcal{T}_h$, $i \neq j$, the intersection $K_i \cap K_j$ is either empty or a vertex or an edge, i.e., hanging nodes are not allowed. The diameter of an element K and the length of an edge E are denoted by h_K and h_E , respectively.

We split the set of all edges \mathcal{E}_h into the set \mathcal{E}_h^0 of interior edges, the set \mathcal{E}_h^D of Dirichlet boundary edges and the set \mathcal{E}_h^N of Neumann boundary edges so that $\mathcal{E}_h = \mathcal{E}_h^B \cup \mathcal{E}_h^0$ with $\mathcal{E}_h^B = \mathcal{E}_h^D \cup \mathcal{E}_h^N$. Let the edge E be a common edge for two elements K and K^e . For a piecewise continuous scalar function y , there are two traces of y along E , denoted by $y|_E$ from inside K and $y^e|_E$ from inside K^e . The jump and average of y across the edge E are defined by:

$$[[y]] = y|_E \mathbf{n}_K + y^e|_E \mathbf{n}_{K^e}, \quad \{\{y\}\} = \frac{1}{2}(y|_E + y^e|_E), \quad (7)$$

where \mathbf{n}_K (resp. \mathbf{n}_{K^e}) denotes the unit outward normal to ∂K (resp. ∂K^e).

Similarly, for a piecewise continuous vector field ∇y , the jump and average across an edge E are given by

$$[[\nabla y]] = \nabla y|_E \cdot \mathbf{n}_K + \nabla y^e|_E \cdot \mathbf{n}_{K^e}, \quad \{\{\nabla y\}\} = \frac{1}{2}(\nabla y|_E + \nabla y^e|_E). \quad (8)$$

For a boundary edge $E \in K \cap \Gamma$, we set $\{\{\nabla y\}\} = \nabla y$ and $[[y]] = y\mathbf{n}$, where \mathbf{n} is the outward normal unit vector on Γ .

For continuous finite element methods (FEMs), the idea is to approximate (6) using a conforming, finite dimensional space $V_h \subset V$. On the other hand, we point out that in discontinuous Galerkin methods the space of solutions or test functions consist of piecewise discontinuous polynomials. That is, no continuity constraints are explicitly imposed on the state and test functions across the element interfaces. As a consequence, weak formulations must include jump terms across interfaces, and typically penalty terms are added to control the jump terms. Then, we define the spaces of test functions, of the solutions by

$$V_h = U_h = \{u \in L^2(\Omega) : u|_K \in \mathbb{P}^r(K) \quad \forall K \in \mathcal{T}_h\}, \quad (9)$$

where $\mathbb{P}^r(K)$ is the set of polynomials of degree at most r in K . Note that the space U_h of discrete solutions and the space of test functions V_h are identical due to the weak treatment of boundary conditions in DG methods. Note also that the space V_h is a non-conforming space such that $V_h \not\subset V$.

Now, we are ready to set up the SIPG discretization of the continuous weak formulation (5). Multiply (5) by a test function $v \in V_h$, and then integrate over each element $K \in \mathcal{T}_h$

$$\sum_{K \in \mathcal{T}_h} \int_K (u_t v - \Delta u v) dx = \sum_{K \in \mathcal{T}_h} \int_K \frac{1}{\varepsilon} f(u) v dx.$$

An application of integration by parts on each element integral gives us

$$\sum_{K \in \mathcal{T}_h} \int_K (u_t v + \nabla u \cdot \nabla v) dx - \sum_{K \in \mathcal{T}_h} \int_{\partial K} (\nabla u \cdot \mathbf{n}) v ds = \sum_{K \in \mathcal{T}_h} \int_K \frac{1}{\varepsilon} f(u) v dx.$$

Then, using the definition of the jump operator we obtain

$$\sum_{K \in \mathcal{T}_h} \int_K (u_t v + \nabla u \cdot \nabla v) dx - \sum_{E \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_E [[v \nabla u]] ds = \sum_{K \in \mathcal{T}_h} \int_K f(u) v dx + \sum_{E \in \mathcal{E}_h^N} \int_E g^N v ds.$$

The following equality

$$[[v\nabla u]] = \{\{\nabla u\}\} \cdot [[v]] + [[\nabla u]] \cdot \{\{v\}\},$$

which one can verify easily and the fact that $[[\nabla u]] = 0$ (u is assumed to be smooth) yield

$$\begin{aligned} \sum_{K \in \mathcal{T}_h} \int_K (u_t v + \nabla u \cdot \nabla v) dx - \sum_{E \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_E \{\{\nabla u\}\} \cdot [[v]] ds \\ = \sum_{K \in \mathcal{T}_h} \int_K f(u) v dx + \sum_{E \in \mathcal{E}_h^N} \int_E g^N v ds. \end{aligned}$$

To handle the coercivity of the left hand side and control the jump terms, we add the following equalities via $[[u]] = 0$ on the interior edges $E \in \mathcal{E}_h^0$

$$\begin{aligned} \sum_{E \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_E \{\{\nabla v\}\} \cdot [[u]] ds &= \sum_{E \in \mathcal{E}_h^D} \int_E g^D (\nabla v \cdot \mathbf{n}) ds, \\ \sum_{\mathcal{E}_h^0 \cup \mathcal{E}_h^D} \frac{\sigma}{h_E} \int_E [[u]] \cdot [[v]] ds &= \sum_{E \in \mathcal{E}_h^D} \frac{\sigma}{h_E} \int_E g^D v ds, \end{aligned}$$

where σ is the penalty parameter, which should be chosen a sufficiently large to ensure the stability of the SIPG scheme, see, e.g., [2]. Then, the weak formulation of the Allen-Cahn equation (5), discretized by the SIPG method reads as: find $u_h \in U_h$ such that

$$\left(\frac{\partial u_h}{\partial t}, v \right) + a_h(u_h, v) = \ell_h(v) \quad \forall v \in V_h, \quad t \in (0, T], \quad (10a)$$

$$(u_h(\cdot, 0), v) = (u_0, v) \quad \forall v \in V_h, \quad (10b)$$

where the (bi)-linear forms are given by

$$\begin{aligned} a_h(u, v) &= \sum_{K \in \mathcal{T}_h} \int_K (\nabla u \cdot \nabla v) dx - \sum_{E \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_E \left(\{\{\nabla u\}\} \cdot [[v]] + \{\{\nabla v\}\} \cdot [[u]] \right) ds \\ &+ \sum_{E \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \frac{\sigma}{h_E} \int_E [[u]] \cdot [[v]] ds, \end{aligned} \quad (11a)$$

$$\ell_h(v) = \sum_{K \in \mathcal{T}_h} \int_K \frac{1}{\varepsilon} f(u) v dx + \sum_{E \in \mathcal{E}_h^D} \int_E g^D \left(\frac{\sigma}{h_E} [[v]] - \{\{\nabla v\}\} \right) ds + \sum_{E \in \mathcal{E}_h^N} \int_E g^N v ds, \quad (11b)$$

and $u_h(\cdot, 0)$ is an orthogonal L^2 -projection of the initial condition u_0 onto U_h .

For each time step, we can expand the discrete solution as

$$u_h(t) = \sum_{i=1}^N \sum_{j=1}^{n_p} U_j^i \phi_j^i, \quad (12)$$

where U_j^i and ϕ_j^i are the unknown coefficients and the basis functions, respectively, for $j = 1, 2, \dots, n_p$ and $i = 1, 2, \dots, N$. The number N denotes the number dG elements and n_p is the local dimension of each dG element with

$$n_p = \frac{(p+1)(p+2)}{2},$$

where p is the degree of the polynomial order.

Inserting (12) into (10), we obtain

$$\frac{dU}{dt} + \mathbf{M}^{-1}\mathbf{L}U = \mathbf{B}(U), \quad (13)$$

where U is the unknown coefficient vector $U = (U_1^1, \dots, U_{n_p}^1, \dots, U_N^1, \dots, U_{n_p}^N)$, \mathbf{M} is the mass matrix, \mathbf{L} is the stiffness matrix corresponding to $a_h(u, v)$, and $\mathbf{B}(\cdot)$ is the nonlinear vector of the unknown coefficient vector U corresponding to $\ell_h(v)$.

We are now ready to employ the matrix transfer technique introduced in [29], which states that the error introduced by approximating the fractional Laplacian by a fractional power of the matrix $\mathbf{A} = \mathbf{M}^{-1}\mathbf{L}$ converges at the same rate as the underlying discretization method. In the following section, we employ the contour integral method introduced in [24] to compute the fractional power of \mathbf{A} times a vector.

3 Contour integration method (CIM)

We remark that an analytic function h of a square matrix \mathbf{A} can be represented as a contour integral in the complex plane [26, Definition 1.11]

$$h(\mathbf{A}) = \frac{1}{2\pi i} \int_{\Gamma} h(z)(z\mathbf{I} - \mathbf{A})^{-1} dz, \quad (14)$$

where $i \equiv \sqrt{-1}$, and Γ is a closed contour lying in the region of analyticity of h and enclosing the spectrum of \mathbf{A} . Then, numerical quadrature method is applied to the integration (14) to approximate $h(\mathbf{A})$.

We here compute the vector $h(\mathbf{A})\mathbf{b}$ for a given vector \mathbf{b} by using definition (14) with the technique proposed in [24]. The basic principle is based on an application of the midpoint rule over a circle contained within an annulus whose outer boundary maps to the interval $(-\infty, 0]$ and whose inner boundary maps to the interval $[\lambda_1, \lambda_N]$, which are the eigenvalues of \mathbf{A} , see Figure 1.

Then, the vector $h(\mathbf{A})\mathbf{b}$ is computed via the following quadrature formula

$$h(\mathbf{A})\mathbf{b} \approx \text{Im} \sum_{i=1}^{n_q} w_i (\eta_i \mathbf{I} - \mathbf{A})^{-1} \mathbf{b}, \quad (15)$$

where the weights and shifts are denoted by w and η , respectively, and n_q is the number of quadrature points.

The SIPG discretization of the Dirichlet problem provides a nonsingular and real-symmetric matrix $\mathbf{A} = \mathbf{M}^{-1}\mathbf{L}$. Then, by using the symmetry property of \mathbf{A} , we can integrate over only the upper half the contour. The algorithm based on the method in [24] is given in Algorithm 1. In the algorithm, we use the routines `ellipkqp` and `ellipjc`, which are described in [17] to compute complex arguments.

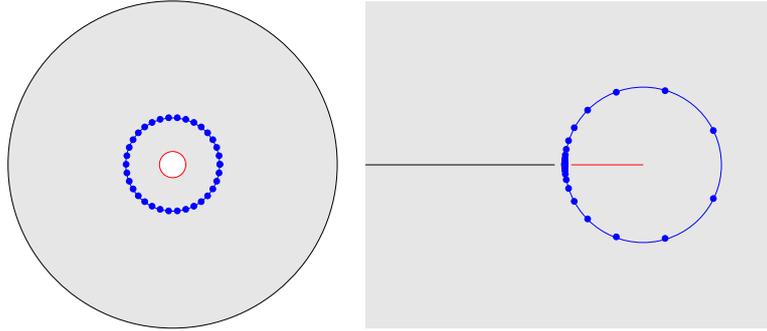


Figure 1: Conformal map from the annulus (left) to the domain $\mathcal{C} \setminus \{(-\infty, 0] \cup [\lambda_1, \lambda_N]\}$ (right). The quadrature points in the CIM denoted by the dots. See, [11] in details.

Algorithm 1 CIM for computing \mathbf{A}^α for the Dirichlet problem

```

1: l = eigs(L,M,1,'SM'); l1 = l(1);           % min. eigenvalue of A
2: l = eigs(L,M,1,'LM'); lN = l(1);          % max. eigenvalue of A
3: k = (sqrt(lN/l1)-1)/(sqrt(lN/l1)+1);      % a convenient constant
4: [K Kp] = ellipkcp(-log(k)/pi);           % elliptic integrals
5: t = .5i*Kp-K+(nq-.5:-1:0)*2*K/nq;        % midpoint rule points
6: [sn cn dn] = ellipjc(t,-log(k)/pi);      % jacobi elliptic functions
7: xi = sqrt(l1*lN)*(1/k+sn)/(1/k-sn);       % quadrature nodes
8: dxidt = cn.*dn./(1/k-sn).^2;            % derivative wrt t
9: wts = h(xi).*dxidt;                      % quadrature weights
10: v = zeros(length(b),1);                 % initialize output
11: for i = 1 : nq do
12:   y = (xi(j)*M-L) \ (M*b);
13:   v = v + wtj(j)*y;                      % update solution vector
14: end for
15: v = -4*K*sqrt(l1*lN)*imag(v)/(k*pi*nq); % scale the solution

```

However, many applications require Neumann-type boundary conditions, which make the matrix \mathbf{A} singular. A contour Γ surrounding the eigenvalues of \mathbf{A} cannot be found. Therefore, the Algorithm 1 should be modified to compute (15). Burrage et al. in [11, Sec. 4] handle this problem by adding a correction term. It is shown in Algorithm 2. The first line of the Algorithm 2 yields the first non-zero eigenvalue of the matrix \mathbf{A} .

4 Implicit-explicit schemes

After spatial discretization of the Allen-Cahn equations, the leading system is typically stiff for small values of the parameter ε . Explicit methods are not suitable for stiff systems, whereas implicit methods require the solution of nonlinear equations at each time step. Therefore, the

Algorithm 2 CIM for computing \mathbf{A}^α for the Neumann problem

```
1: l = eigs(L,M,3,'SM'); l1 = l(1);           % min. eigenvalue of A
2: l = eigs(L,M,1,'LM'); lN = l(1);          % max. eigenvalue of A
3:                                     :
4: v = -4*K*sqrt(l1*lN)*imag(v)/(k*pi*nq); % scale the solution
5: e = ones(length(b),1);
6: v = v + (e'*(M*(b-v)))/(e'*M*e)*e;      % corrector term
```

implicit-explicit (IMEX) method can play an important role for such problems, see, [3, 4]. In such a procedure, the Laplacian term is discretized implicitly in time and the nonlinear terms are discretized explicitly. This can also be recognized and analyzed as a splitting technique. In addition, it typically allows a larger time step than explicit methods while avoiding the use of nonlinear solvers.

We first divide the time interval $[0, T]$ as follows

$$0 = t_0 < t_1 < \dots < t_{N_T} = T$$

with the time step size $\tau_n = t_n - t_{n-1}$, $n = 1, 2, \dots, N_T$. Then, we can consider the first- and second-order IMEX approximations of the following system of ordinary differential equations (ODEs)

$$\frac{dU}{dt} + \mathbf{A}^\alpha U = \mathbf{B}(U). \quad (16)$$

4.1 First-order implicit-explicit schemes

The first-order implicit-explicit schemes for ODEs (13) can be written as

$$\frac{U^{n+1} - U^n}{\tau_n} + \mathbf{A}^\alpha \left(\theta U^{n+1} + (1 - \theta) U^n \right) = \mathbf{B}(U^n), \quad (17)$$

where θ is a free parameter and α is the fractional order. We point out that choosing $\theta = 1$ results in the backward Euler scheme.

4.2 Second-order implicit-explicit schemes

The second-order implicit-explicit schemes for ODEs (13) can be written as

$$\begin{aligned} & \frac{U^{n+1} + \theta_1 U^n - (1 - \theta_1) U^{n-1}}{\tau_n} \\ & + \mathbf{A}^\alpha \left(\left(\theta_2 - \frac{\theta_1}{2} \right) U^{n+1} + \left(2 + \frac{3}{2} \theta_1 - 2\theta_2 \right) U^n + \theta_2 U^{n-1} \right) \\ & = \left(2 + \frac{\theta_1}{2} \right) \mathbf{B}(U^n) + \frac{\theta_1}{2} \mathbf{B}(U^{n-1}), \end{aligned} \quad (18)$$

where θ_1 and θ_2 are two free parameters. In our numerical examples, we use the following modified Crank-Nicolson/Adams-Bashforth scheme by choosing $(\theta_1, \theta_2) = (-1, \frac{1}{16})$

$$\frac{U^{n+1} - U^n}{\tau_n} + \mathbf{A}^\alpha \left(\frac{9}{16} U^{n+1} + \frac{3}{8} U^n + \frac{1}{8} U^{n-1} \right) = \frac{3}{2} \mathbf{B}(U^n) - \frac{1}{2} \mathbf{B}(U^{n-1}). \quad (19)$$

Remark 4.1 In our numerical simulations, we use two different matrix functions $h(z)$ to compute the fractional matrix \mathbf{A}^α , which is formed on both left-and right-hand side of the IMEX scheme in (17) or (18). When we first apply the Laplace transform and then Laplace inversion to the ODE system (13), the matrix function $h(z)$ is defined in terms of exponential function, see [40]. Then, we have $h(z)$ as

$$h(z) = \frac{1}{\exp(\tau z^\alpha)}$$

for the left-hand side. Note that the fraction is due to inversion in (17) or (18). On the other hand, for the right-hand side, we choose as $h(z) = z^\alpha$ as was done in [24].

4.3 Time-step size adaptivity

For small values of the parameter ε , the transition layer moves slowly and then an inordinate number of time steps is required to resolve the dynamics response of the fractional-in-space Allen-Cahn equation (1). To reduce the amount of work, adaptivity in time should be used. Our time-step size adaptivity is based on the ideas presented in [7, 45]. To update the time-step size, we use the difference between two solutions, which are a predictor and a corrector. The first-order IMEX schemes are chosen as a predictor, whereas the second-order IMEX schemes are chosen as a corrector. The time-step adaptive algorithm is presented in Algorithm 3. To update the time-step size, we use the following controller

$$\tau_{n+1}^* = \rho \left(\frac{Tol}{e} \right)^{1/2} \tau, \quad (20)$$

where ρ is a safety coefficient, which is introduced to reduce the probability of rejecting τ_{n+1}^* . In numerical examples, we take $\rho = 0.9$ as suggested in [32]. The parameter Tol determines the required accuracy of the numerical solution. The impact of Tol on the number of time steps will be studied in Section 5. Finally, to avoid a strong increase or decrease of subsequent time steps, we use the following formula as proposed in the deterministic framework [23]

$$\Lambda(e_n, \tau_n) = \min\{s_{max} \tau_n, \max\{s_{min} \tau_n, \tau_{n+1}^*\}\}, \quad (21)$$

In numerical simulations, $s_{min} = 0.1$ and $s_{max} = 2$ are used. A step size is accepted if $e_n \leq Tol$, otherwise it is rejected.

The time-step size adaptivity allows us to reduce the computation time by factors of hundreds compared to the uniform step size.

5 Numerical results

In this section, we investigate the performance of our spatial and temporal discretization strategies for the fractional-in-space Allen-Cahn equations. To achieve the required accuracy for all

Algorithm 3 Time-step adaptive algorithm

```
1: Given  $U_0, \tau_0, Tol$ 
2: for  $n = 1, 2, \dots, N_T$  do
3:   Compute  $P_n$  using a first-order implicit-explicit scheme.
4:   Compute  $C_n$  using a second-order implicit-explicit scheme.
5:   Calculate  $e_n = \frac{\|C_n - P_n\|}{\|C_n\|}$ .
6:   Set  $reject = 0$ .
7:   if  $e_n > Tol$  then
8:     Recalculate time-step size  $\tau_n \leftarrow \Lambda(e_n, \tau_n)$ .
9:     Update  $reject = reject + 1$ .
10:    goto step 3.
11:  else
12:    Update time-step size  $\tau_{n+1} = \Lambda(e_n, \tau_n)$ .
13:  continue
14:  end if
15: end for
```

examples, 50 quadrature points are used in the contour integral method described in Section 3. We use piecewise linear polynomials to form the SIPG discretization in space in all numerical experiments. The penalty parameter σ in the SIPG discretization is chosen as $\sigma = 6$ on the interior edges \mathcal{E}^0 and $\sigma = 12$ on the boundary edges \mathcal{E}^{∂} . All examples are implemented on a mesh, constructed by first dividing Ω into 32×32 uniform squares and then dividing each square into two triangles.

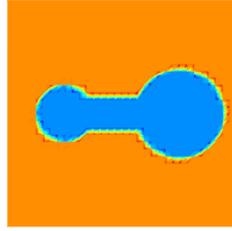


Figure 2: Initial condition of Example 5.1.

5.1 Dumbbell example with double-well potential

We first consider a dumbbell example, taken from [19], with the double-well potential (3). The data of problem are

$$\Omega = [-1, 1]^2, \quad \partial\Omega = \Gamma^N, \quad g^N = 0, \quad \varepsilon = 0.0025, \quad \tau_0 = 5 \times 10^{-5},$$

with the following initial condition

$$u_0(x,y) = \begin{cases} \tanh\left(\frac{3}{\epsilon}\left((x-0.5)^2 + y^2 - (0.39)^2\right)\right), & \text{if } x > 0.14, \\ \tanh\left(\frac{3}{\epsilon}\left(y^2 - (0.15)^2\right)\right), & \text{if } -0.3 \leq x \leq 0.14, \\ \tanh\left(\frac{3}{\epsilon}\left((x+0.5)^2 + y^2 - (0.25)^2\right)\right), & \text{if } x < -0.3. \end{cases}$$

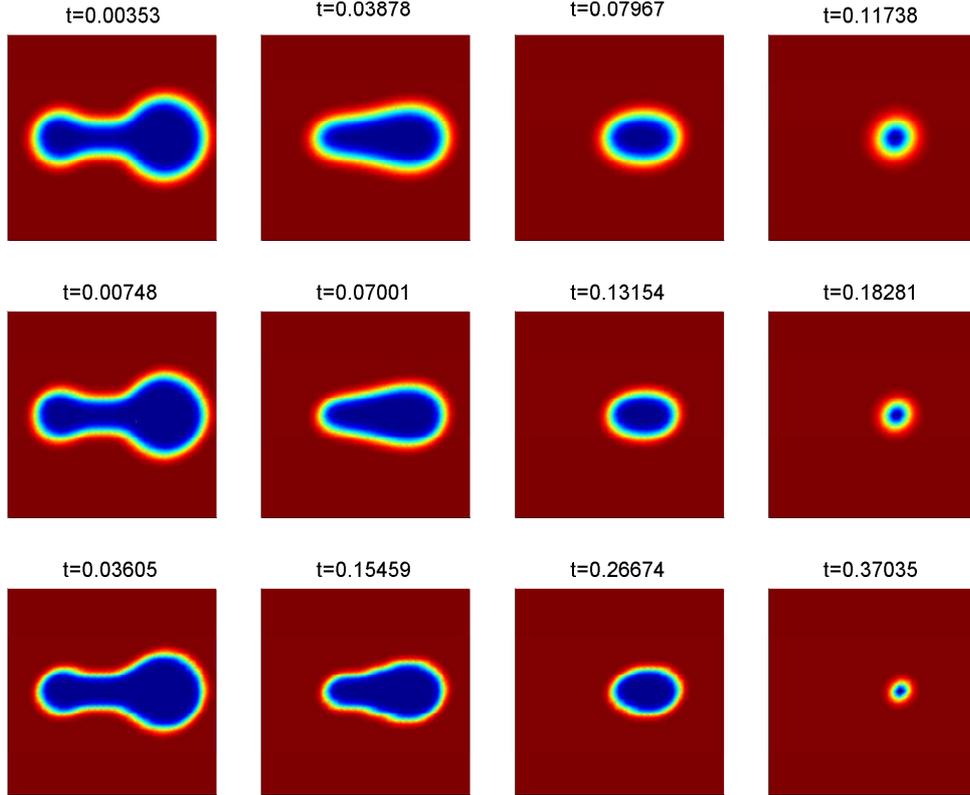


Figure 3: Example 5.1: Diffusion power $\alpha = 1, 0.9, 0.8$ (from left to right) with $Tol = 10^{-3}$.

Figure 2 displays of the initial function u_0 , which is a dumbbell shape with unequal bells. The snapshots of the solution of the fractional-in-space Allen-Cahn equation in time are displayed for various fractional powers ($\alpha = 1, 0.9, 0.8$) in Figure 3. With standard diffusion, i.e., $\alpha = 1$, we see that the curvature drives toward to a circle (constant curvature) in time. The motion of smaller fractional powers is similar, although the rate is slower.

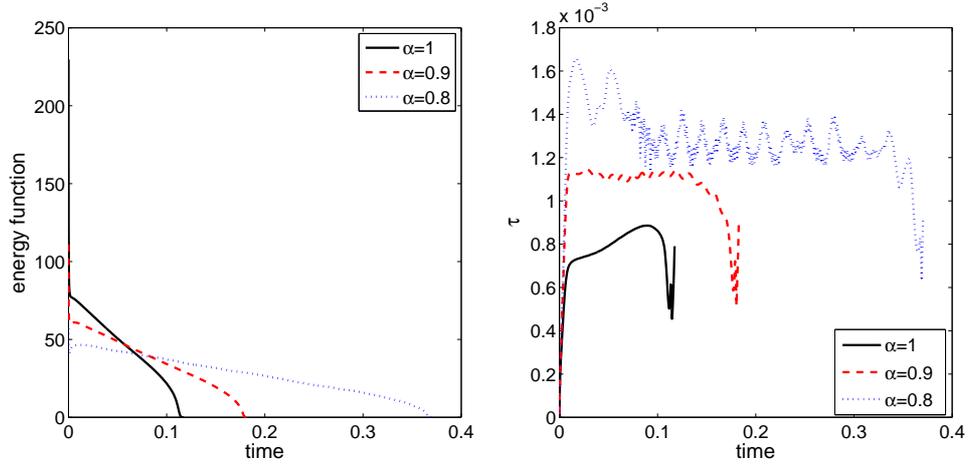


Figure 4: Example 5.1: Energy function versus time (left) and time-step size versus time (right) with $Tol = 10^{-3}$.

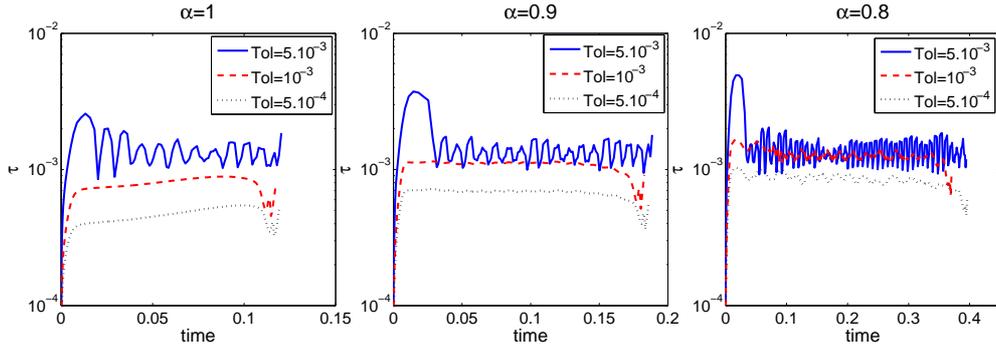


Figure 5: Example 5.1: Evolution of the length of the time step with various Tol parameters.

The behaviour of the numerical energy function (4) and the adaptive time-step size is displayed in Figure 4 with the tolerance parameter $Tol = 10^{-3}$. The energy function (4) decreases in time for all cases. Reducing the fractional power increases the required time to reach the metastable state. For parameters $Tol \in \{5 \cdot 10^{-3}, 10^{-3}, 5 \cdot 10^{-4}\}$, the evolution of the length of the time step is shown in Figure 5. The time-step size oscillates for smaller fractional powers, when the tolerance parameter is large. In addition, the number of time steps increases with decreasing the parameter Tol and the fractional power α .

The number of time-steps are given in Table 1 with various tolerance parameters. It can be seen that the number of rejected time-steps is increasing for small fractional powers with large tolerance parameter. Table 2 also shows the performance of the adaptive time-step size

Tol	$\alpha = 1$		$\alpha = 0.9$		$\alpha = 0.8$	
	Total	Rej.	Total	Rej.	Total	Rej.
5.10^{-3}	99	16	143	22	288	45
10^{-3}	174	2	194	2	312	5
5.10^{-4}	290	2	307	2	500	2

Table 1: Example 5.1: Number of time steps (total, rejected) for $\alpha = 1, 0.9, 0.8$ with various tolerance parameters.

α	# Adaptive time-steps	# Uniform time-steps
1	174	2348
0.9	194	3657
0.8	312	7408

Table 2: Example 5.1: Number of time steps for adaptive and uniform time-step size approaches with $Tol = 10^{-3}$.

with respect to the uniform time-step size, i.e., $\tau = 5 \times 10^{-5}$. As expected, the time-step size adaptivity allows us to reduce the computing time compared to the uniform time-step size.

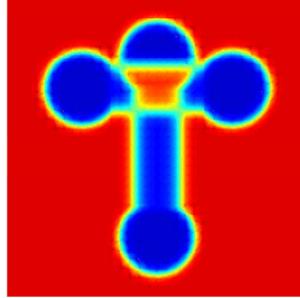


Figure 6: Initial condition of Example 5.2.

5.2 Intersection of two dumbbells with double-well potential

We now investigate an intersection of two dumbbells on the Laplacian. The double-well potential (3) function is taken. The rest of problem data are

$$\Omega = [-1, 1]^2, \quad \partial\Omega = \Gamma^N, \quad g^N = 0, \quad \varepsilon = 0.01, \quad \tau_0 = 5 \times 10^{-5},$$

with the following initial condition

$$u_0(x, y) = u_0^1(x, y)u_0^2(x, y),$$

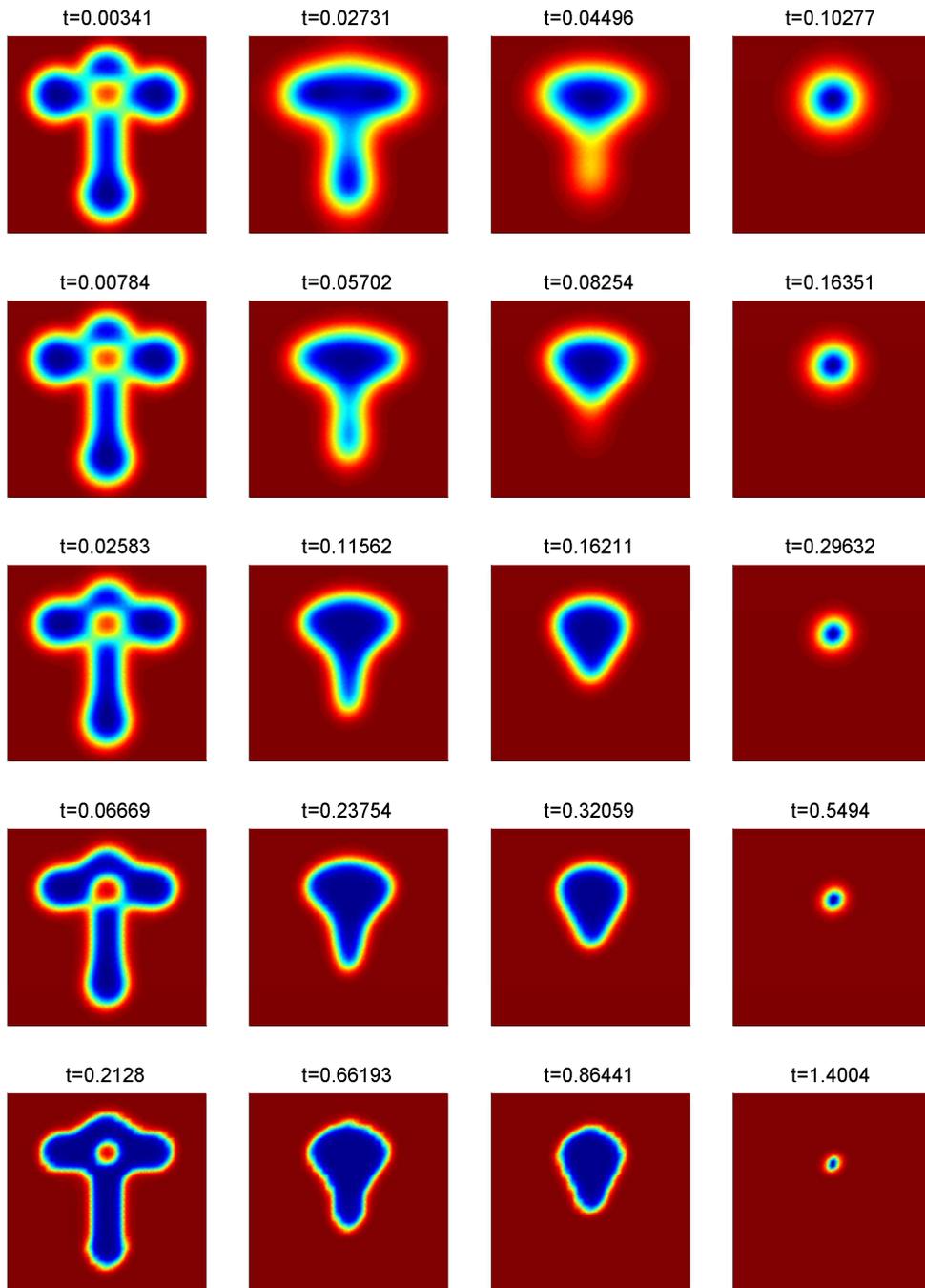


Figure 7: Example 5.2: Diffusion power $\alpha = 1, 0.9, 0.8, 0.7, 0.6$ (from top to bottom).

where

$$u_0^1(x,y) = \begin{cases} \tanh\left(\frac{3}{\sqrt{\varepsilon}}((x-0.5)^2 + (y-0.4)^2 - (0.25)^2)\right), & \text{if } x > 0.3, \\ \tanh\left(\frac{3}{\sqrt{\varepsilon}}((y-0.4)^2 - (0.15)^2)\right), & \text{if } -0.3 \leq x \leq 0.3, \\ \tanh\left(\frac{3}{\sqrt{\varepsilon}}((x+0.5)^2 + (y-0.4)^2 - (0.25)^2)\right), & \text{if } x < -0.3, \end{cases}$$

and

$$u_0^2(x,y) = \begin{cases} \tanh\left(\frac{3}{\sqrt{\varepsilon}}(x^2 + (y-0.6)^2 - (0.25)^2)\right), & \text{if } y > 0.4, \\ \tanh\left(\frac{3}{\sqrt{\varepsilon}}(x^2 - (0.15)^2)\right), & \text{if } -0.4 \leq y \leq 0.4, \\ \tanh\left(\frac{3}{\sqrt{\varepsilon}}(x^2 + (y-0.6)^2 - (0.25)^2)\right), & \text{if } y < -0.4. \end{cases}$$

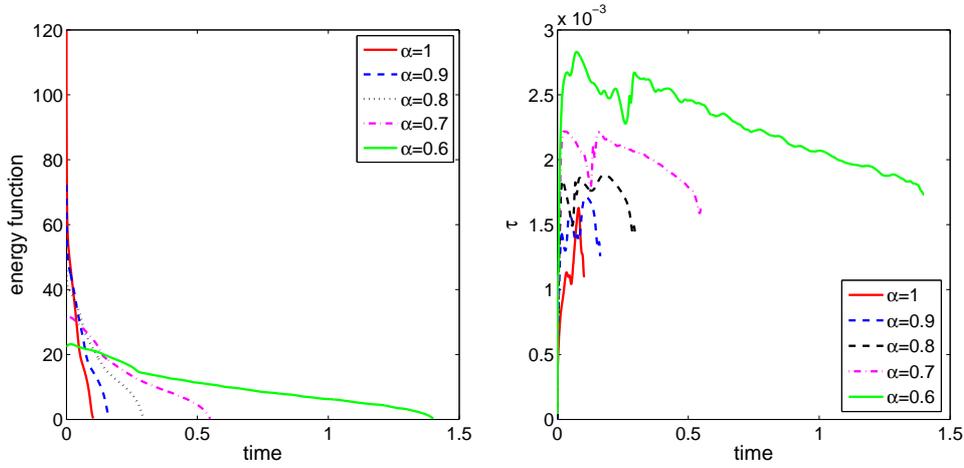


Figure 8: Example 5.2: Energy function versus time (left) and time-step size versus time (right) with $Tol = 10^{-3}$.

The initial function u_0 , which is an intersection of two dumbbells, is shown in Figure 6. For various fractional powers ($\alpha = 1, 0.9, 0.8, 0.7, 0.6$), the snapshots of the solutions in time are displayed in Figure 7. As the previous example, reducing of fractional power decreases the rate of motion of initial curvature. Figure 8 illustrates the behaviour of the numerical energy function (4) and the adaptive time-step size. The evolution of the length of the time-step for $\alpha = 0.8, 0.7, 0.6$ is shown in Figure 9 for parameters $Tol \in \{5 \cdot 10^{-3}, 10^{-3}, 5 \cdot 10^{-4}\}$. It is observed that decreasing the tolerance parameter makes the motion of time-step size smoother.

Table 3 shows the number of time-steps for $\alpha = 1, 0.9, 0.8, 0.7, 0.6$ with various tolerance parameters. The number of time steps increases with decreasing the parameter Tol and the fractional power α . Further, the number of the adaptive and uniform time-steps are displayed in Figure 4. It can be seen that the time-step size adaptivity allows us to reduce the computation time by factors of hundreds compared to the uniform time-step size.

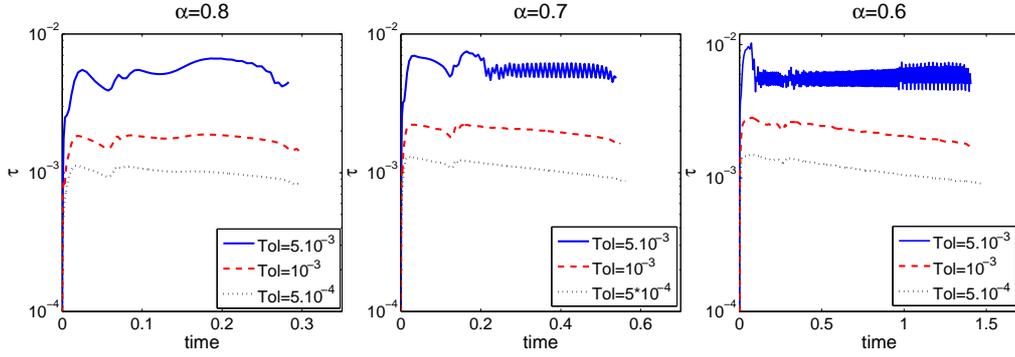


Figure 9: Example 5.2: Evolution of the length of the time step with various Tol parameters for $\alpha = 0.8, 0.7, 0.6$.

Tol	$\alpha = 1$		$\alpha = 0.9$		$\alpha = 0.8$		$\alpha = 0.7$		$\alpha = 0.6$	
	Total	Rej.	Total	Rej.	Total	Rej.	Total	Rej.	Total	Rej.
5.10^{-3}	40	4	47	2	62	1	102	29	250	115
10^{-3}	107	1	123	1	181	1	285	1	637	0
5.10^{-4}	192	1	203	1	313	1	530	1	1287	1

Table 3: Example 5.2: Number of time steps (total, rejected) for $\alpha = 1, 0.9, 0.8, 0.7, 0.6$ with various tolerance parameters.

α	# Adaptive time-steps	# Uniform time-steps
1	107	2056
0.9	123	3271
0.8	181	5927
0.7	285	10988
0.6	637	28090

Table 4: Example 5.2: Number of time steps for adaptive and uniform time-step size approaches with $Tol = 10^{-3}$.

5.3 Star-shaped interface with double-well potential

This example is a star-shaped interface in a curvature-driven flow, taken from [33]. The rest of problem data are

$$\Omega = [0, 1]^2, \quad \partial\Omega = \Gamma^N, \quad g^N = 0, \quad \varepsilon = 0.003, \quad \tau_0 = 5 \times 10^{-5},$$

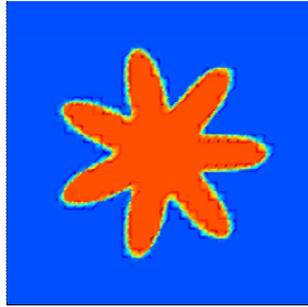


Figure 10: Initial condition of Example 5.3.

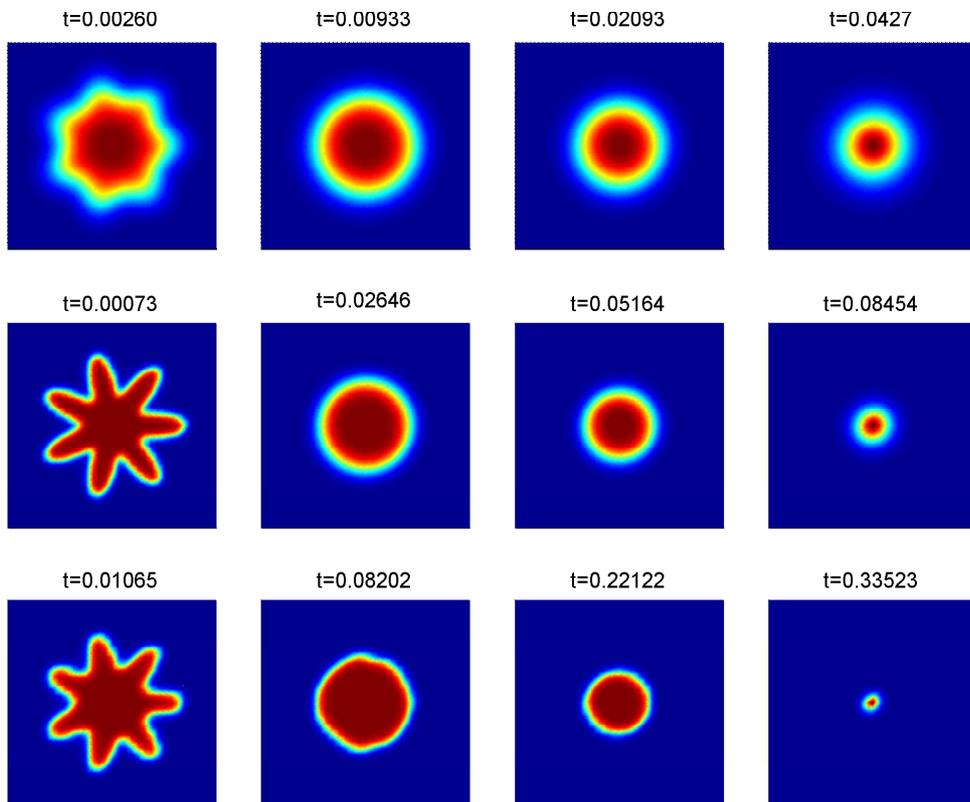


Figure 11: Example 5.3: Diffusion power $\alpha = 1, 0.82, 0.65$ (from left to right) with $Tol = 10^{-3}$.

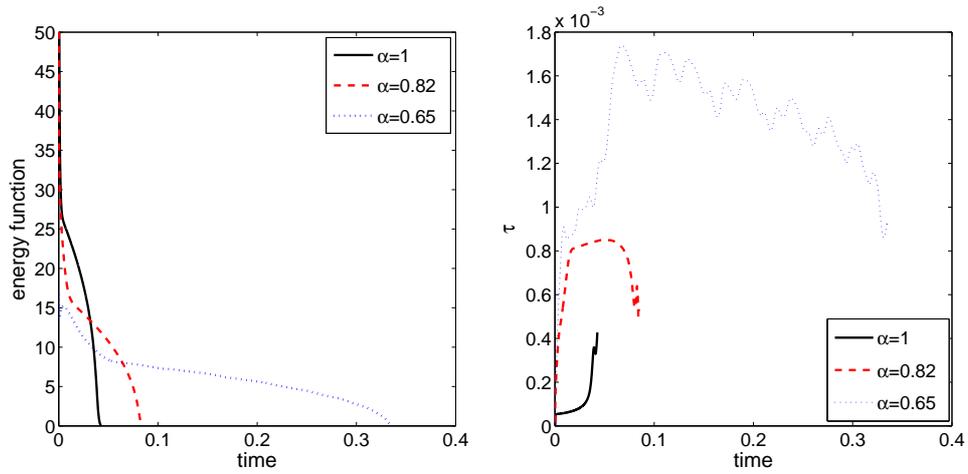


Figure 12: Example 5.3: Energy function versus time (left) and time-step size versus time (right) with $Tol = 10^{-3}$.

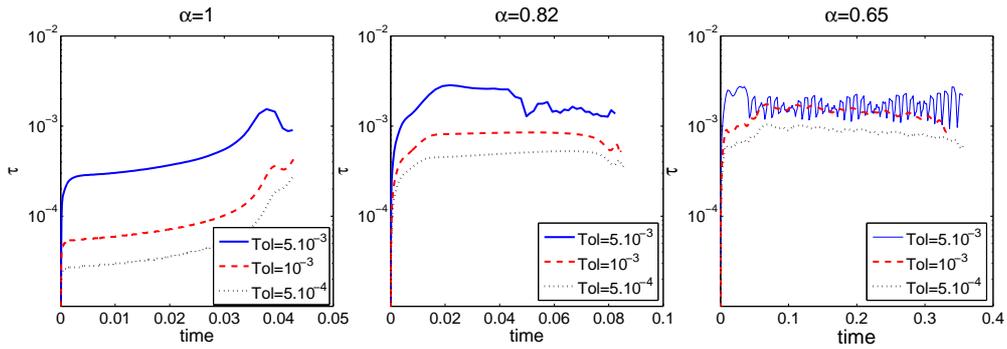


Figure 13: Example 5.3: Evolution of the length of the time step with various Tol parameters for $\alpha = 1, 0.82, 0.65$.

with the following initial condition

$$u_0(x, y) = \tanh \frac{0.25 + 0.1 \cos(7\theta) - \sqrt{(x-0.5)^2 + (y-0.5)^2}}{\sqrt{2}\varepsilon},$$

where

$$\theta = \begin{cases} \tan^{-1} \left(\frac{y-0.5}{x-0.5} \right), & \text{if } x > 0.5, \\ \pi + \tan^{-1} \left(\frac{y-0.5}{x-0.5} \right), & \text{otherwise.} \end{cases}$$

The initial function u_0 , which is a star-shaped interface in a curvature-driven flow, is shown in Figure 10. The snapshots of the computed solutions are displayed in Figure 11. The tips of the star move inward, while the gaps between the tips move outward. The curvature deforms to a circular shape and the radius of the circle shrinks in time. As expected, the rate of the motion is slower for the small number of the fractional powers. Figure 12 shows the behaviour of the energy function and the adaptive time-step size in time, while Figure 13 shows the evolution of the time-step size for various tolerance parameters.

5.4 Spinodal decomposition with logarithmic free energy

We now consider a test example with the logarithmic free energy (2). The initial condition is a random state by randomly distributing numbers from -0.01 to 0.01 . The rest of problem data are

$$\Omega = [0, 2\pi]^2, \quad \partial\Omega = \Gamma^D, \quad g^D = 0, \quad \tau_0 = 5 \times 10^{-5}, \quad \theta = 0.1, \quad \theta_c = 0.2.$$

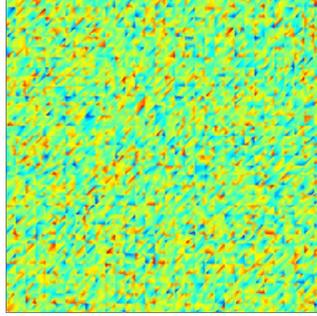


Figure 14: Initial condition of Example 5.4.

In this example, we investigate the effect of fractional power when a spinodal decomposition is considered. The initial state is well mixed, see Figure 14. The snapshots of phase evolution for various values of fractional power ($\alpha = 1, 0.8, 0.6$) are illustrated in Figure 15 with $\varepsilon = 10^{-3}$. Early stages of phase transition yields a rapid movement to bulk regions for $\alpha = 1$. However, smaller fractional powers leads much more heterogeneous phase structures with smaller bulk regions. Figure 16 also shows the snapshots of phase evaluation at $t = 0.016$ with $\varepsilon = 10^{-4}$.

The behaviour of the numerical energy function (4) and the adaptive time-step size versus time is displayed in Figure 17 for $\varepsilon = 10^{-4}$. The numerical energy decrease is observed for all the cases. Lastly, Figure 18 shows the evolution of the time-step size for various tolerance parameters.

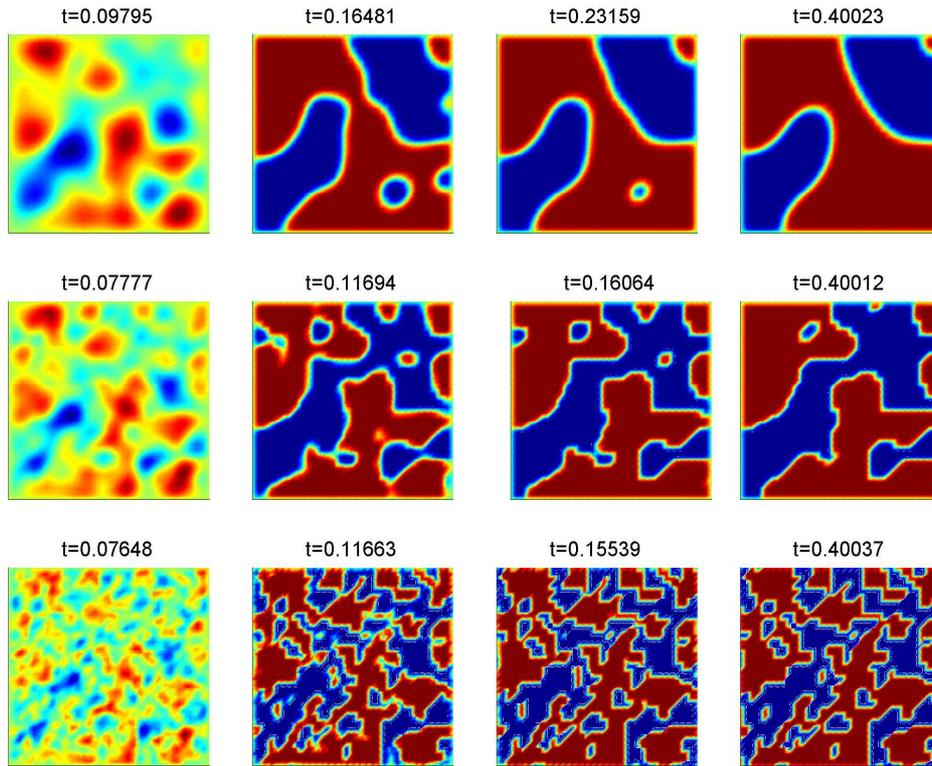


Figure 15: Example 5.4: Diffusion power $\alpha = 1, 0.8, 0.6$ (from top to bottom) with $\varepsilon = 10^{-3}$ and $Tol = 10^{-4}$.

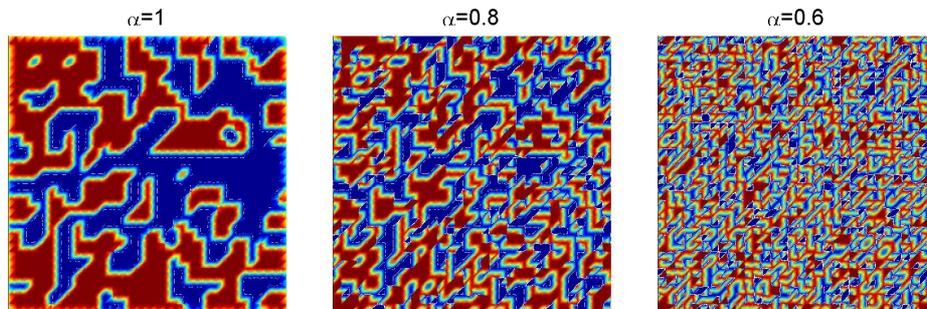


Figure 16: Example 5.4: Diffusion power $\alpha = 1, 0.8, 0.6$ (from left to right) with $\varepsilon = 10^{-4}$ and $Tol = 10^{-4}$ at $t = 0.016$.

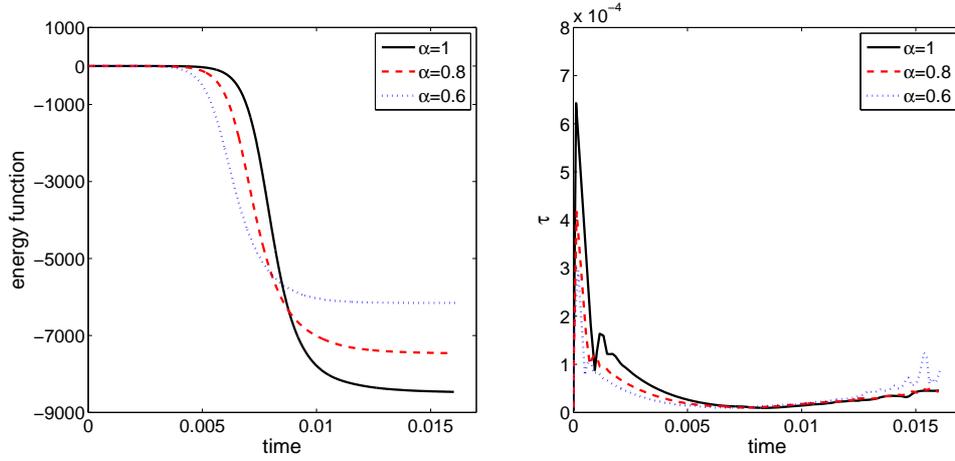


Figure 17: Example 5.3: Energy function versus time (left) and time-step size versus time (right) with $\varepsilon = 10^{-4}$ and $Tol = 10^{-4}$.

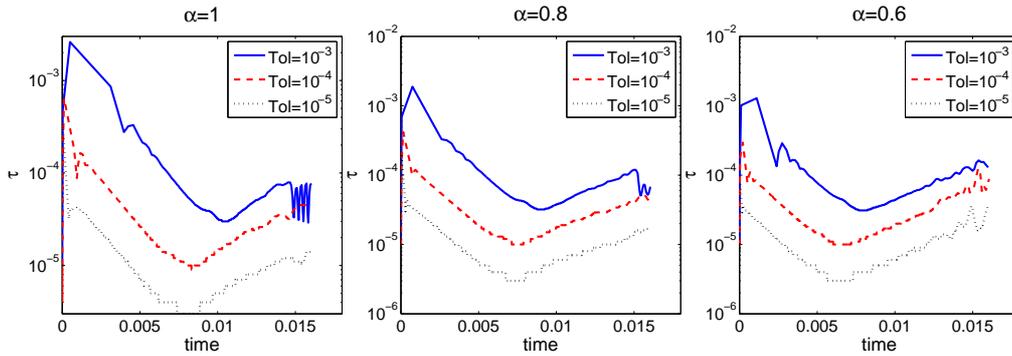


Figure 18: Example 5.4: Evolution of the length of the time step with various Tol parameters with $\varepsilon = 10^{-4}$ for $\alpha = 1, 0.8, 0.6$.

6 Conclusions

In this paper we have investigated the numerical solutions of the fraction-in-space Allen-Cahn equations, discretized the symmetric interior penalty Galerkin (SIPG) method in space and an implicit-explicit (IMEX) method in time. The contour integral method (CIM) has been used to compute the fractional power of a matrix times a vector. To reduce computation time, an adaptive time-step size method is proposed. The numerical results of the fractional-in-space Allen-Cahn equations show that such a kind of modelling can be an aid to understanding the effects of spatial heterogeneity. Although, the ideas expressed in this paper have applicability

in this setting, the numerical approximations of the Riesz derivatives, discretized by discontinuous Galerkin methods, should also be considered in the more general framework.

Acknowledgements

This work was supported by the Forschungszentrum Dynamische Systeme (CDS): Biosystemtechnik, Otto-von-Guericke-Universität Magdeburg.

References

- [1] S. M. Allen and J. W. Cahn. A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. *Acta Metall*, 27:1159–1180, 1979.
- [2] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779, 2002.
- [3] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit runge-kutta method for time dependent partial differential equations. *Appl. Numer. Math.*, 25:151–167, 1997.
- [4] U. M. Ascher, S. J. Ruuth, and T. R. Wetton. Implicit-explicit method for time dependent partial differential equations. *SIAM J. Numer. Anal.*, 32:797–823, 1995.
- [5] G. Barles, H. M. Soner, and P. E. Souganidis. Front propagation and phase field theory. *SIAM J. Control. Optim.*, 31:439–469, 1993.
- [6] S. Bartels and R. Müller. Error control for the approximation of AllenCahn and Cahn-Hilliard equations with a logarithmic potential. *Numer. Math.*, 119(3):409–435, 2011.
- [7] G. Benderskaya, M. Clemens, H. De Gerssem, and T. Weiland. Embedded Runge-Kutta methods for field-circuit coupled problems with switching elements. *IEEE Trans. Magn.*, 41(5):1612–1615, 2005.
- [8] D. A. Benson, S. W. Wheatcraft, and M. M. Meerschaert. Application of a fractional advection-dispersion equation. *Water Resour. Res.*, 36:1403–1412, 2000.
- [9] T. Breiten, V. Simoncini, and M. Stoll. Low-rank solvers for fractional differential equations, 2014. MPI Magdeburg Preprint,MPIMD/14-02.
- [10] W. Bu, Y. Tang, and J. Yang. Galerkin finite element method for two dimensional Riesz space fractional diffusion equations. *J. Comput. Phys.*, 276:26–38, 2014.
- [11] K. Burrage, N. Hale, and D. Kay. An efficient implicit fem scheme for fractional-in-space reaction-diffusion equations. *SIAM J. Sci. Comput.*, 34:A2145–A2172, 2012.
- [12] S. Capuani, M. Palombo, A. Orlandi, B. Maraviglia, and F. S. Pastore. Spatio-temporal anomalous diffusion imaging: results in controlled phantoms and in excised human meningiomas. *Magn. Reson. Imaging.*, 31:359–365, 2013.

- [13] E. Celledoni, V. Grimm, R. I. McLachlan, D. I. McLaren, D. O’Neale, B. Owren, and G. R. W. Quispel. Preserving energy resp. dissipation in numerical PDEs using the ”Average Vector Field” method. *J. Comput. Phys.*, 231:6770–6789, 2012.
- [14] A. Christlieb, J. Jones, B. Wetton, K. Promislow, and M. Willoughby. High accuracy solutions to energy gradient flows from material science models. *J. Comput. Phys.*, 257:193–215, 2014.
- [15] W. H. Deng. Finite element method for the space and time fractional Fokker-Planck equation. *SIAM J. Numer. Anal.*, 47:204–226, 2008.
- [16] W. H. Dong and J. S. Hesthaven. Local discontinuous Galerkin methods for fractional diffusion equations. *ESAIM: Math. Model. Numer. Anal.*, 47:1845–1864, 2013.
- [17] T. A. Driscoll. Improvement to the Schwarz-Christoffel toolbox for MATLAB. *ACM Trans. Math. Software*, 31:239–251, 2005.
- [18] V. J. Ervin and J. P. Roop. Variational formulation for the stationary fractional advection dispersion equation. *Numer. Methods. Partial Differ. Eqs.*, 22:558–576, 2005.
- [19] X. Feng and Y. Li. A posteriori error estimates and an adaptive finite element method for the Allen-Cahn equation and the mean curvature flow. *J. Sci. Comput.*, 24:121–146, 2005.
- [20] X. Feng and Y. Li. Analysis of symmetric interior penalty discontinuous galerkin methods for the Allen-Cahn equation and the mean curvature flow. *IMA J. Numer. Anal.*, pages 193–215, 2014.
- [21] X. Feng, H. Song, T. Tang, and J. Yang. Nonlinear stability of the implicit-explicit methods for the Allen-Cahn equation. *Inverse Probl. Imag.*, 7:679–695, 2013.
- [22] R. Gorenflo and F. Mainardi. Fractional calculus: integral and differential equations of fractional order, 2008. arXiv:0805.3823.
- [23] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential Algebraic Problems*. Springer Series in Computational Mathematics, Vol. 14. Springer Verlag, Berlin, Heidelberg, New York, 1991.
- [24] N. Hale, N. J. Higham, and L. N. Trefethen. Computing A^α , $\log(A)$, and related matrix functions by contour integrals. *SIAM J. Numer. Anal.*, 46:2505–2323, 2008.
- [25] G. R. Hernández-Labrodo, R. E. Constreas-Donayre, J. E. Collazos-Castro, and J. L. Polo. Subdiffusion behaviour in poly(3,4-ethylenedioxythiophene):polystyrene sulfonate (PEDOT:PSS) evidenced by electrochemical impedance spectroscopy. *J. Electroanal. Chem.*, 659:201–204, 2011.
- [26] N.J. Higham. *Functions of Matrices: Theory and Computation*. SIAM, Philadelphia, 2008.

- [27] J. Huang, Y. Tang, and L. Vázquez. Converge analysis of a Block-by-Block method for fractional differential equations. *Numer. Math., Theory Methods Appl.*, 339:229–241, 2012.
- [28] M. Ilić, F. Liu, I. Turner, and V. Anh. Numerical approximation of a fractional-in-space diffusion equation, I. *Frac. Calc. and App. Anal.*, 8:323–341, 2005.
- [29] M. Ilić, F. Liu, I. Turner, and V. Anh. Numerical approximation of a fractional-in-space diffusion equation (II) with nonhomogeneous boundary conditions. *Frac. Calc. and App. Anal.*, 9:333–349, 2006.
- [30] M. Ilić, I. Turner, F. Liu, and V. Anh. Analytical and numerical solutions of a one-dimensional fractional-in-space diffusion equation in a composite medium. *Appl. Math. Comput.*, 216:2248–2262, 2010.
- [31] B. Karasözen, A. S. Filibelioğlu, and M. Uzunca. Energy stable discontinuous Galerkin finite element method for the Allen-Cahn equation, 2015. arXiv:1409.3997.
- [32] J. Lang. Two-dimensional fully adaptive solutions of reaction-diffusion equations. *Appl. Numer. Math.*, 18:223–240, 1995.
- [33] Y. Li, H. G. Lee, D. Jeong, and J. Kim. An unconditionally stable hybrid numerical method for solving the Allen-Cahn equation. *Comput. Math. Appl.*, 60:1591–1606, 2010.
- [34] Y. Lin and C. Xue. Finite difference/spectral approximations for the time-fractional diffusion equation. *J. Comput. Phys.*, 225:1533–1552, 2007.
- [35] F. Liu and J. Shen. Stabilized semi-implicit spectral deferred correction methods for Allen-Cahn and Cahn-Hilliard equations. *Math. Methods Appl. Sci.*, 2013. In press.
- [36] F. Liu, P. Zuang, V. Anh, I. Turner, and K. Burrage. Stability and convergence of the difference methods for the space-time fractional advection-diffusion equation. *Appl. Math. Comput.*, 191:12–20, 2007.
- [37] G. B. Mcfadden. Phase field models of solidification. *Contemp. Math.*, 295:107–145, 2007.
- [38] R. Metzler and J. Klafter. The random walk’s guide to anomalous diffusion: a fractional dynamics approach. *Phys. Rep.*, 339:1–77, 2000.
- [39] N. Nie, J. Huang, W. Wang, and Y. Tang. Solving spatial-fractional partial differential diffusion equations by spectral method. *J. Stat. Comput. Simul.*, 84:1173–1189, 2014.
- [40] H. K. Pang and H. W. Sun. Fast numerical contour integral method for fractional diffusion equations. *J. Sci. Comput.*, pages 1–26, 2015.
- [41] L. Qiu, W. Deng, and J. S. Hesthaven. Nodal discontinuous Galerkin methods for fractional diffusion equations on 2D domain with triangular meshes, 2014. arXiv:1410.0796.

- [42] E. Scales, R. Gorenflo, and F. Mainardi. Fractional calculus and continuous time-finance. *Phys. A.*, 284:376–384, 2000.
- [43] J. Shen and X. Yang. Numerical approximations of Allen-Cahn and CahnHilliard equations. *Discret. Contin. Dyn.-A*, 28:1669–1691, 2010.
- [44] C. Tadjeran and M. M. Meerschaert. A second-order accurate numerical method for the two-dimensional fractional diffusion equation. *J. Comput. Phys.*, 220:813–823, 2007.
- [45] P. J. van der Houwen, B. P. Sommeijer, and W Couzy. Embedded diagonally implicit Runge-Kutta algorithms on parallel computers. *Math. Comput.*, 58:135–159, 1992.
- [46] Q. Yang. *Novel analytical and numerical methods for solving fractional dynamical systems*. PhD thesis, Queensland University of Technology, Brisbane, 2005.
- [47] Q. Yang, I. Turner, F. Liu, and M. Ilić. Novel numerical methods for solving the time-space fractional diffusion equation in two dimensions. *SIAM J. Sci. Comput.*, 33:1159–1180, 2011.
- [48] X. Zhao and Z. Z. Sun. Compact Crank-Nicolson schemes for a class of fractional Cattaneo equation inhomogeneous medium. *J. Sci. Comput.*, 62:747–771, 2015.
- [49] Z. Zhao and C. Li. Fractional difference finite element approximations for the time-space fractional telegraph equation. *Appl. Math. Comput.*, 219:2975–2988, 2012.

