

A Novel Discretization and Numerical Solver for Non-Fourier Diffusion

TAO XUE*, Rutgers University

HAOZHE SU*, Rutgers University

CHENGGUIZI HAN, Rutgers University

CHENFANFU JIANG, University of Pennsylvania

MRIDUL AANJANEYA, Rutgers University



Fig. 1. Our generalized diffusion solver allows for some of the most characteristic visual aspects of diffusion-driven physics. Large-scale particle-based snowfall simulation, where the zoom-ins show the different snowflake patterns generated by our diffusion solver. These patterns include those that can be captured using traditional Fourier-type diffusion as well as new ones that cannot be described with the traditional approach, but which are observed in the real world.

We introduce the *C-F diffusion model* [Anderson and Tamma 2006; Xue et al. 2018] to computer graphics for diffusion-driven problems that has several attractive properties: (a) it fundamentally explains diffusion from the perspective of the non-equilibrium statistical mechanical Boltzmann Transport Equation, (b) it allows for a finite propagation speed for diffusion, in contrast to the widely employed Fick's/Fourier's law, and (c) it can capture some of the most characteristic visual aspects of diffusion-driven physics, such as hydrogel swelling, limited diffusive domain for smoke flow, snowflake

and dendrite formation, that span from Fourier-type to non-Fourier-type diffusive phenomena. We propose a unified convection-diffusion formulation using this model that treats both the diffusive quantity *and* its associated flux as the primary unknowns, and that recovers the traditional Fourier-type diffusion as a limiting case. We design a novel semi-implicit discretization for this formulation on staggered MAC grids and a geometric Multigrid-preconditioned Conjugate Gradients solver for efficient numerical solution. To highlight the efficacy of our method, we demonstrate end-to-end examples of elastic porous media simulated with the Material Point Method (MPM), and diffusion-driven Eulerian incompressible fluids.

*joint first authors

Authors' addresses: Tao Xue, Rutgers University; Haozhe Su, Rutgers University; Chengguizi Han, Rutgers University; Chenfanfu Jiang, University of Pennsylvania; Mridul Aanjaneya, Rutgers University.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.

0730-0301/2020/12-ART178 \$15.00

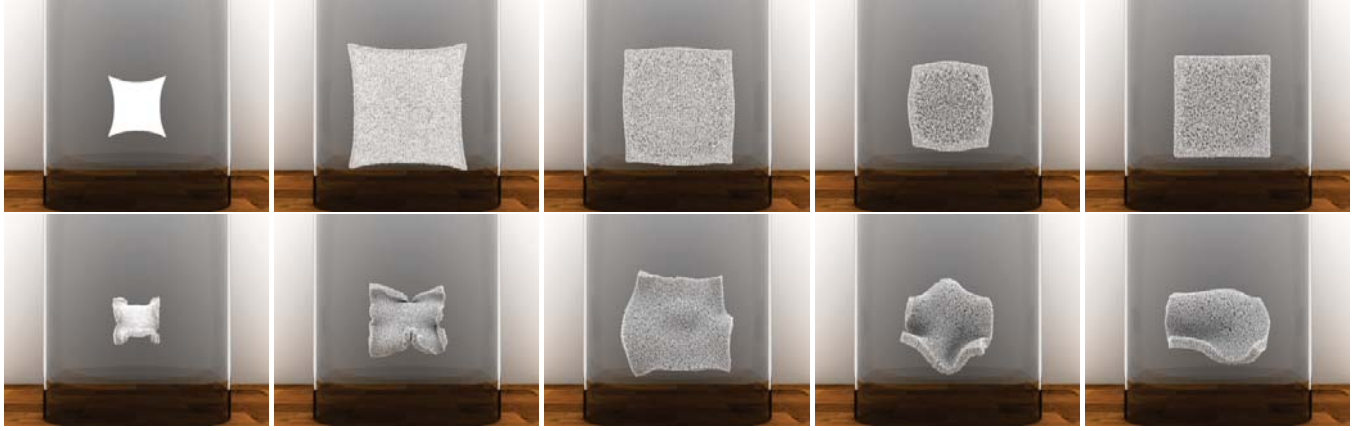
<https://doi.org/10.1145/3414685.3417863>

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

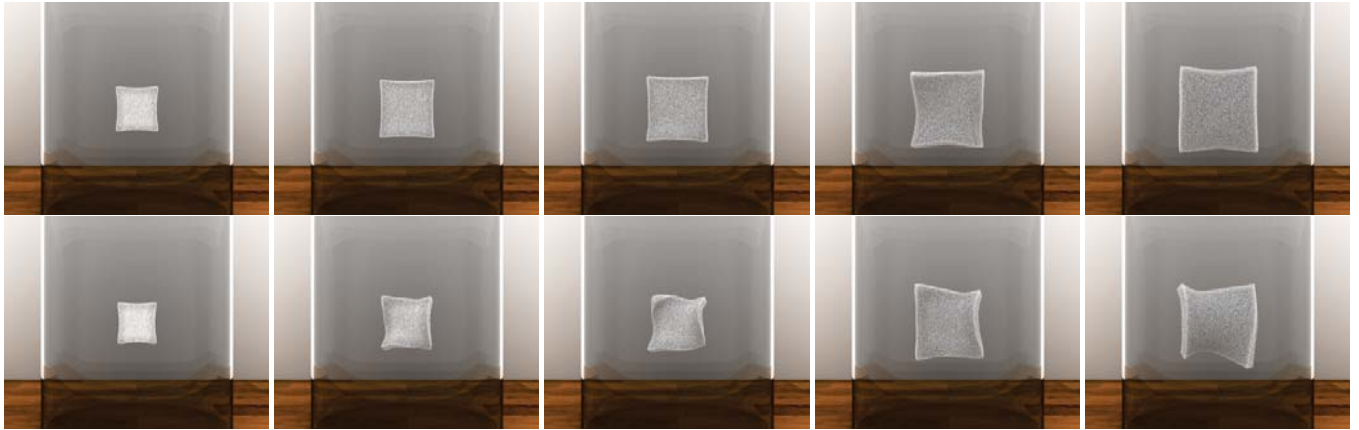
Additional Key Words and Phrases: non-Fourier, C-F diffusion, Fick's law, convection-diffusion, MPM, incompressible flow, Multigrid solver

ACM Reference Format:

Tao Xue, Haozhe Su, Chengguizi Han, Chenfanfu Jiang, and Mridul Aanjaneya. 2020. A Novel Discretization and Numerical Solver for Non-Fourier Diffusion. *ACM Trans. Graph.* 39, 6, Article 178 (December 2020), 14 pages. <https://doi.org/10.1145/3414685.3417863>



Hydrogel with compliant material (Young's modulus 40) using the Fick's law (top) and the generalized C-F diffusion model (bottom).



Hydrogel with stiff material (Young's modulus 200) using the Fick's law (3rd row) and the generalized C-F diffusion model (4th row).

Fig. 2. Poroelastic material dynamics of a hydrogel driven by fluid absorption, simulated using our proposed method in combination with MPM (512^3 grid resolution). The hydrogel exhibits rich anisotropic deformations with the generalized C-F diffusion model, which qualitatively match real-world observations.

1 INTRODUCTION

Diffusion-driven phenomena such as smoke spreading in air, sponge swelling in water, ink dispersing on paper, heat-induced material expansion/shrinkage, and many more are integral parts of our daily lives [Ding et al. 2019; Lenaerts et al. 2008; Yang et al. 2017] as well as nature. In contrast to other physical processes, diffusion typically affects material constitution, and can drive interesting dynamics as a consequence, encompassing complex physical phenomena such as heat and mass transfer, dynamics of porous mixtures, thermo-fluid-structural interaction, etc. Thus, to precisely describe such processes and the like, the fundamental model of diffusion, specifically the diffusive constitutive law, plays a vital role. The most popular model for diffusion is based on Fick's /Fourier's law [Fick 1855; Fourier 1878]. This model has been acknowledged to be able to describe most problems of interest. Not surprisingly, Fourier-type models have been dominant in computer graphics for decades. However, it can be shown via rigorous mathematical derivations that the speed of diffusion in Fourier-based models is infinite [Baumeister and Hamill 1969; Cattaneo 1948], and thus, non-physical.

In this paper, we introduce the *C-F diffusion model* [Anderson and Tamma 2006; Xue et al. 2018] to computer graphics that (a) fundamentally explains diffusion from the perspective of the nonequilibrium statistical mechanical Boltzmann Transport Equation (BTE), (b) addresses the issue of infinite propagation speed of diffusion, and (c) captures some of the most characteristic visual aspects of diffusion-driven physics, such as hydrogel swelling, limited diffusive domain for smoke flow, ink dispersing in water, dendrite and snowflake growth, etc. To simulate the effects of this model on staggered MAC grids, we propose a novel semi-implicit discretization that leads to a better conditioned linear system than the Fourier system. This allows iterative Krylov solvers, such as preconditioned Conjugate Gradients (PCG) to converge in relatively fewer iterations compared to traditional Fourier-type diffusion. For even faster convergence, we design a tailored geometric Multigrid preconditioner for Conjugate Gradients following [McAdams et al. 2010].

Our method uses a novel two-field convection-diffusion formulation to simulate diffusion-driven effects using the generalized constitutive model. In contrast to the traditional Fourier-type equation for diffusion, which uses a single-field formulation where the primary

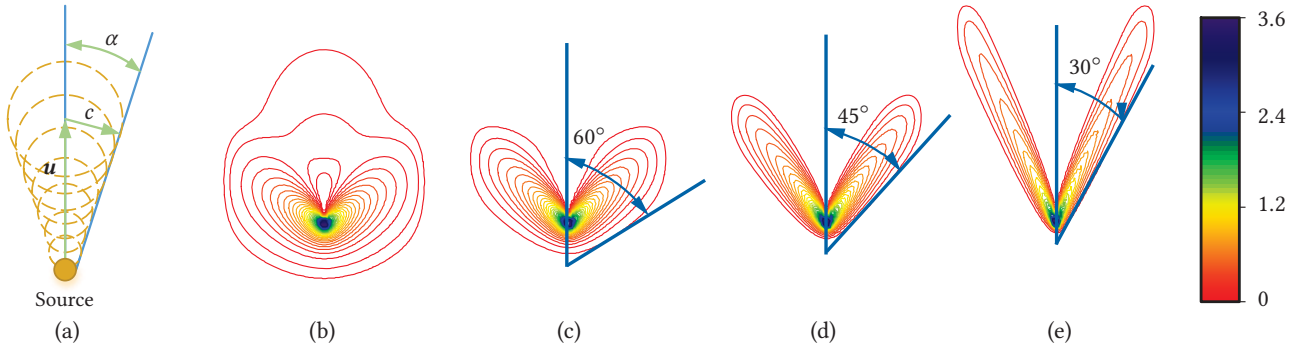


Fig. 3. (a) A source emits diffusive matter in a medium (e.g. smoke flow) with uniform background velocity $\mathbf{u}$. The propagation speed of diffusion is c , and the angle $\alpha = \sin^{-1}(1/H_c)$. (b-d) Density contours for diffusive matter in the critical regime ($H_c = 0$) (b), and super-critical regime with $H_c = 2/\sqrt{3}$ and angle 60° (c), $H_c = \sqrt{2}$ and angle 45° (d), and $H_c = 2$ and angle 30° (e). As shown, in the super-critical regime, the diffusive matter only propagates in a limited diffusive domain that assumes a “conical” shape.

unknown is only the diffusive quantity, both the diffusive quantity and its associated flux are treated as primary unknowns in our framework. This allows us to analyze the interaction/competition between the diffusion process and background material dynamics. Due to the finite propagation speed of diffusion, we define a specific taxonomy of diffusion processes and provide a mathematical prediction of the active and null regime of diffusive quantities. These particular phenomena are illustrated via simulations of an ideal example where a source emits diffusive matter in a medium with uniform background velocity. The proposed diffusion taxonomy also shows that our model can recover the traditional Fourier-type diffusion. More realistic examples, including ink dispersion in water and smoke flow, are presented to further demonstrate non-Fourier effect on real-world diffusion processes.

We demonstrate the versatility of our method in simulations of poroelastic material dynamics induced by change in fluid saturation. We use the Material Point Method (MPM) [Jiang et al. 2016] to simulate the porous material. The solid continuum contains “voids” into which the fluid diffuses, changing the local saturation and driving the motion. As the saturation changes in a wave-like propagation, the resulting motion is an interaction between a diffusive wave and a stress wave. The resulting motion is more dynamic compared to the one obtained using the traditional Fourier-type model. While our poromechanical kinematic assumptions are similar to [Ding et al. 2019; Lenaerts et al. 2008], these prior works only considered Fourier-type diffusion models. We use the example of a hydrogel swelling in water to highlight the benefits of our proposed method.

We also utilize our method to investigate various patterns of dendrite and snowflake growth in nature, which is formulated as a phase change (Stefan) problem [Kim et al. 2006; Ren et al. 2018; Stefan 1891; Voller and Swaminathan 1991]. We use phase-field modeling to establish the governing equations for this problem. The classical Fourier-based phase-field modeling is revised by introducing the generalized C-F diffusion model; specifically, the phase equation and the heat transfer equation are both reformulated in the generalized form, which gives a strongly coupled *non-Fourier* system. Similar studies regarding the phase-field modeling or Stefan problem can be

found in [Kim et al. 2006; Kim and Lin 2003; Yang et al. 2017]. However, these mathematical models are governed by the traditional Fourier-type law and result in purely parabolic partial differential equations. In contrast, our non-Fourier model involves hyperbolic characteristics. We use this generalized model to simulate the formation of dendrites (e.g., coral reef) and snowflakes. Moreover, our proposed Multigrid solver for non-Fourier diffusion can be directly utilized in both the phase parameter equation and the heat transfer equation. To summarize, our main contributions are the following:

- (1) introduction of *Cattaneo-Fourier (C-F) diffusion model* [Anderson and Tamma 2006; Xue et al. 2018] to computer graphics,
- (2) a semi-implicit discretization for diffusion with the generalized constitutive law using a convection-diffusion approach,
- (3) a geometric Multigrid-preconditioned Conjugate Gradients solver to allow fast simulations of non-Fourier diffusion,
- (4) a diffusion taxonomy using the C-F diffusion model that predicts a limited diffusion regime for convection-diffusion effects in a medium with uniform background flow, and
- (5) high resolution simulations of poroelastic material dynamics, generalized convection-diffusion problems, and Stefan problems using the proposed non-Fourier diffusion solver.

2 RELATED WORK

Diffusion is ubiquitous in continuum physical systems such as advection-diffusion and reaction-diffusion systems [Kim and Lin 2007]. While there are innumerable papers with the standard Fourier diffusion equation, or slightly modified versions of Fourier-based diffusion integrated, we only list some representative ones to define the scope of applications and methods that we study in this work.

2.1 Diffusion of liquids in solids

Diffusion of liquids in solid structures is common in everyday solid-fluid mixtures, and has stimulated a lot of computational research in the graphics community, such as watercolor painting on paper [Chu and Tai 2005; Curtis et al. 1997], wet shells [Chen et al. 2018; Um et al. 2013], wet cloth [Fei et al. 2018], wet hair [Lin 2014, 2015; Rungtjiratananon et al. 2012], wet sand [Rungtjiratananon et al. 2008],

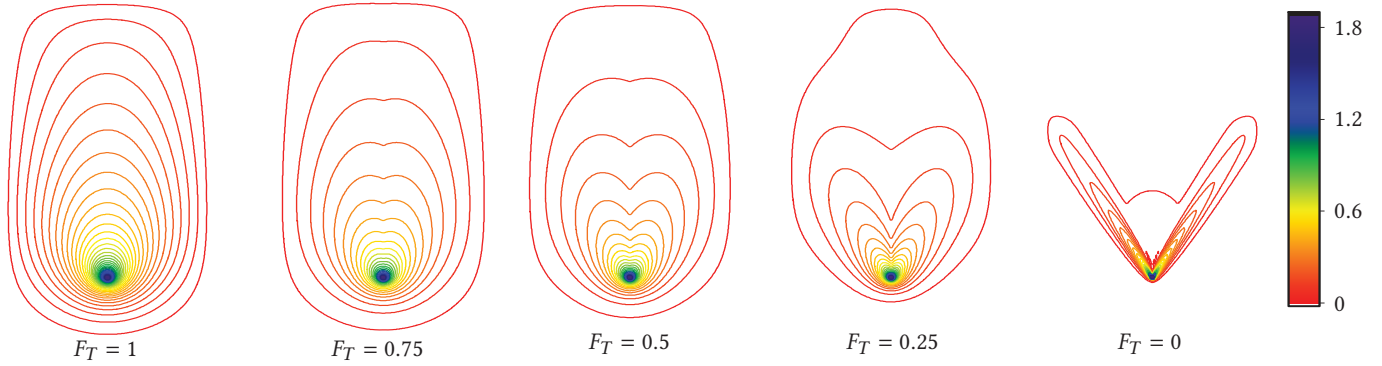


Fig. 4. Our proposed convection-diffusion formulation spans patterns from pure Fick's diffusion ($F_T = 1$) (left) to pure Cattaneo's diffusion ($F_T = 0$) (right).

and sponges [Lenaerts et al. 2008; Patkar and Chaudhuri 2013]. When ignoring inertial effects, these models reduce to the Fick's law (Fourier-based diffusion) with constant or saturation and porosity-dependent diffusivity [Fei et al. 2018; Huber et al. 2011].

2.2 Diffusion in multi-phase fluids

Miscible fluids, whose characteristic behavior is largely driven by convection-diffusion effects, is a key research area in multi-phase fluids and contributes to many interesting visual effects. Ren et al. [2014] adopted concepts from mixture theory and simulated the diffusion of multiple miscible fluids into each other due to concentration difference. This work, based on Smoothed Particle Hydrodynamics, was further extended to multi-phase solid-fluid mixtures by Yan et al. [2016]. An Eulerian framework based on diffusing volume fractions was proposed by Kang et al. [2010]. Recently, Canabal et al. [2020] proposed a reaction-diffusion type dendritic painting growth model. With a specifically designed nonlinear diffusion function by adding extra exponents and noise functions to the density field, their model enables rich sub-branching effects and thickness control. For modeling liquid sprays, Nielsen and Østerby [2013] introduced a speed dependent diffusion coefficient, evolving the Eulerian volume fraction and velocity of fluid species to capture the mass flux of small droplets.

2.3 Heat solvers

Heat transport is also a popular application of the diffusion equation and has its unique attractiveness in computer graphics since the early days [Carlson et al. 2002; Stora et al. 1999] for generating vivid melting and solidification effects. Impressive simulations have been done for thermoelastic solid-liquid transitions on locally remeshing Lagrangian meshes [Clausen et al. 2013], particles [Iwasaki et al. 2010], and hybrid Lagrangian-Eulerian structures [Ding et al. 2019; Gao et al. 2018b; Stomakhin et al. 2014]. All of these models are based on the Fourier's law, having the heat flow linearly dependent on the temperature gradient.

2.4 Phase-field modeling

We use phase-field theory to model snowflake and dendrite formation as a phase change (Stefan) problem. Phase-field modeling is advantageous in capturing material interfaces. In computer graphics, it has been applied to ice crystal growth from water vapor [Kim et al. 2004] (aka diffusion-limited growth), two-phase diffusive flow

[He et al. 2015], multiple-phase fluid mixtures through the Navier-Stokes-Cahn-Hilliard equations [Yang et al. 2015] or the Allen-Cahn equations [Yang et al. 2017], damage evolution in solid materials [Wolper et al. 2019], and controllable dendritic crystal simulations by novel orientation field model [Ren et al. 2018]. Compared to sharp interface approaches such as the level-set method [Osher et al. 2004], the phase-field representation uses a regularized scalar field to greatly simplify the interface treatment, while still converging to the sharp result under grid refinement.

2.5 Material Point Method

MPM's hybrid Lagrangian-Eulerian nature provides a natural framework for discretizing the coupled Newtonian and non-Fourier diffusion dynamics efficiently on a grid with the Lagrangian particle motion to handle the otherwise challenging conservative convection. Even though originally introduced into graphics for snow animation [Stomakhin et al. 2013], MPM has proven quite effective in systems where multi-physics with multiple governing physical mechanisms take place. In addition to conservations of mass and momentum, MPM was used for enabling coupled solve with heat propagation [Stomakhin et al. 2014], porous media [Tampubolon et al. 2017], material crack [Wretborn et al. 2017], continuum-discrete mixture [Gao et al. 2018a; Yue et al. 2018], chemical reaction [Yan et al. 2018], and phase-field damage evolution [Wolper et al. 2019]. We draw inspirations from these methods and propose a novel discretization scheme where the non-Fourier diffusion mechanism is coupled with the velocity update in a semi-implicit manner.

3 GOVERNING EQUATIONS

We first review the governing equations of diffusion, and subsequently describe the C-F diffusion model, as well as our two-field convection-diffusion formulation using this constitutive law.

3.1 Classical Diffusion Equation

The classical equation for diffusion is given as follows:

$$\rho c_p \frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{q} = W, \quad (1)$$

where ϕ is the quantity being diffused, $\mathbf{q}$ is the associated flux, W is the diffusive source/sink, ρ is the material density, and c_p is the diffusive capacity. In this paper, ϕ can be the temperature, saturation,

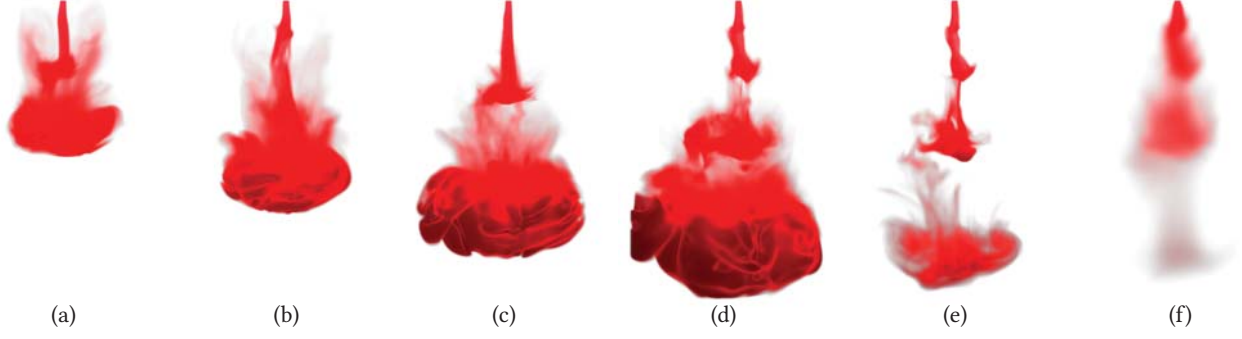


Fig. 5. (a-d) Simulation of ink diffusing in water using our generalized diffusion solver with $F_T = 0$ and $\tau = 4$ ($256^2 \times 512$ grid). Interestingly, the C-F diffusion model produces a more vivid behavior, in contrast incompressible flow with no diffusion (e) and Fourier's diffusion (f) that produces an overly diffuse profile.

and/or phase-field value. For simplicity of exposition, we set $\rho c_p = 1$ in the remaining paper.

For most engineering applications, the flux $\mathbf{q}$ has been typically described by the Fick's/Fourier's law [Fick 1855; Fourier 1878]:

$$\mathbf{q} = -k \nabla \phi, \quad (2)$$

where k is the coefficient of diffusion. It can be rigorously proved that the propagation speed of diffusion using this model is infinite [Baumeister and Hamill 1969] and thus, non-physical. To address this issue, the *Cattaneo-Vernotte (CV) diffusion model* [Cattaneo 1948] was developed that introduced a time-relaxation term to the flux $\mathbf{q}$ to ensure that the wave speed remained finite. Specifically, the CV model is given as:

$$\tau \dot{\mathbf{q}} + \mathbf{q} = -k \nabla \phi, \quad (3)$$

where τ represents the relaxation time with respect to the flux. Note that when $\tau = 0$, the CV model recovers the classical Fourier's law.

Substituting Fick's law (2) into the classical diffusion equation (1) yields the following parabolic partial differential equation:

$$\frac{\partial \phi}{\partial t} = k \nabla \cdot \nabla \phi + W, \quad (4)$$

while substituting the CV model (3) into the diffusion equation (1) yields the following hyperbolic partial differential equation:

$$\tau \frac{\partial^2 \phi}{\partial t^2} + \frac{\partial \phi}{\partial t} = k \nabla \cdot \nabla \phi + W. \quad (5)$$

Neither the Fick's law [Fick 1855; Fourier 1878] nor the CV model [Cattaneo 1948] can thoroughly describe diffusion phenomena such as ballistic and ballistic-diffusive processes [Chen 2001]. Being observed from experiments, they are empirical in nature, without a proper theoretical justification following from first principles (see [Anderson and Tamma 2006; Tzou 2014] for more details).

3.2 Generalized C-F Diffusion Model

To describe diffusion and diffusion-driven processes from the perspective of the fundamental Boltzmann Transport Equation, the generalized constitutive model, also known as the *C-F diffusion model*, was proposed in [Anderson and Tamma 2006] in the context of heat conduction processes. This model assumes that both Cattaneo-type slow processes (or *C-processes*) and Fourier-type fast

processes (or *F-processes*) coexist individually and concurrently during the diffusion process. The total flux $\mathbf{q}$ is given as follows:

$$\begin{aligned} \mathbf{q} &= \int_0^{\omega_D} \mathbf{v} f(\mathbf{x}, \phi, \omega, \cdot) \hbar \omega D(\omega) d\omega \\ &= \underbrace{\int_0^{\omega_T} \mathbf{v} f(\mathbf{x}, \phi, \omega, \cdot) \hbar \omega D(\omega) d\omega}_{\mathbf{q}_C} + \underbrace{\int_{\omega_T}^{\omega_D} \mathbf{v} f(\mathbf{x}, \phi, \omega, \cdot) \hbar \omega D(\omega) d\omega}_{\mathbf{q}_F} \end{aligned} \quad (6)$$

where f represents *concentration* in the diffusion process, $\hbar$ is the reduced Planck's constant, so $\hbar \omega$ is the energy, $\mathbf{v}$ is the particle velocity, and $D(\omega)$ is a density function. The assumption here is that the first integral from zero up to a threshold frequency ω_T associates with the slow process ($\mathbf{q}_C$) of diffusion, and the second integral from ω_T to the Debye frequency ω_D , associates with the fast process ($\mathbf{q}_F$) of diffusion. The resulting constitutive model is:

$$\begin{aligned} \mathbf{q} &= \mathbf{q}_C + \mathbf{q}_F, \\ \mathbf{q}_C + \tau \dot{\mathbf{q}}_C &= -(1 - F_T) k \nabla \phi, \\ \mathbf{q}_F &= -F_T k \nabla \phi, \end{aligned} \quad (7)$$

and we refer the reader to [Anderson and Tamma 2006; Xue et al. 2018] for further details.

The non-dimensional parameter F_T is defined as $F_T = k_F/k$, where k_F is the diffusivity of fast propagation, k_C is the diffusivity of slow propagation, and $k = k_F + k_C$ is the total diffusivity. It parametrizes diffusion and yields the following effects:

- (1) Setting $F_T = 0$ gives Cattaneo-type diffusion, which has a finite speed of propagation for diffusion.
- (2) Setting $F_T \in (0, 1)$ gives *Jeffreys-type diffusion* [Joseph and Preziosi 1989], where discontinuities that arise due to the C-process are smoothed by the F-process.
- (3) Setting $F_T = 1$ gives the familiar Fourier-type diffusion, which has infinite speed of propagation for diffusion (see Figure 4).

3.3 Two-field Convection-Diffusion Formulation

Based on the C-F diffusion model [Anderson and Tamma 2006; Xue et al. 2018], we derive a unified two-field convection-diffusion formulation for diffusion in a medium (such as a fluid) moving with background velocity $\mathbf{u}$. From equation (1), the time rate of change of the diffusive quantity ($\partial \phi / \partial t$) in a differential control volume is



Fig. 6. Simulation of diffusion in a medium with horizontal background flow with uniform velocity. (Left) The traditional Fourier-type diffusion solver produces an overly diffuse profile, since diffusion propagates both upstream and downstream. (Middle and Right) Our generalized diffusion solver produces realistic spray patterns with a limited “conical” diffusive region, as predicted by our diffusion taxonomy for different values of H_C in the super-critical diffusive regime.

determined by the flow (convection/advection) and diffusion into and out of the system, along with any generation (source) or consumption (sink) of ϕ inside the control volume. It follows that:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{q}_{\text{diff}} + \mathbf{q}_{\text{con}}) = W, \quad (8)$$

where the flux $\mathbf{q}$ has been decomposed into a convection part $\mathbf{q}_{\text{con}} = \mathbf{u}\phi$ and a diffusion part $\mathbf{q}_{\text{diff}}$. Substituting the C-F diffusion model [Anderson and Tamma 2006; Xue et al. 2018] now gives:

$$\begin{aligned} \frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) + \nabla \cdot \mathbf{q}_{\text{diff}} &= W, \\ \mathbf{q}_{\text{diff}} &= \mathbf{q}_F + \mathbf{q}_C, \\ \mathbf{q}_F + F_T k \nabla \phi &= 0, \\ \tau \frac{\partial \mathbf{q}_C}{\partial t} + \tau \nabla \cdot (\mathbf{u} \otimes \mathbf{q}_C) + \mathbf{q}_C + (1 - F_T) k \nabla \phi &= 0. \end{aligned} \quad (9)$$

Note that equation (9) includes the influence of convection on the time rate of change of $\mathbf{q}_C$. In this formulation, both ϕ and $\mathbf{q}$ are treated as primary unknowns, and the nonlinear coupling between them prevents further reduction into a single-field formulation with respect to ϕ , as is typical for Fourier-based diffusion. When the background material is an incompressible fluid, equation (9) can be further simplified by integrating the differential identities $\nabla \cdot (\mathbf{u}\phi) = \mathbf{u} \cdot \nabla \phi + (\nabla \cdot \mathbf{u})\phi$ and $\nabla \cdot (\mathbf{u} \otimes \mathbf{q}_C) = \mathbf{u} \cdot \nabla \mathbf{q}_C + (\nabla \cdot \mathbf{u})\mathbf{q}_C$ along with the incompressibility condition $\nabla \cdot \mathbf{u} = 0$ to yield the relation:

$$\begin{aligned} \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi + \nabla \cdot \mathbf{q} &= W, \\ \mathbf{q} &= \mathbf{q}_F + \mathbf{q}_C, \\ \mathbf{q}_F + F_T k \nabla \phi &= 0, \\ \tau \left(\frac{\partial \mathbf{q}_C}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{q}_C \right) + \mathbf{q}_C + (1 - F_T) k \nabla \phi &= 0. \end{aligned} \quad (10)$$

The above convection-diffusion formulation for incompressible media recovers the one proposed in [Xue et al. 2018], and is utilized in Section 4 for designing a semi-implicit discretization scheme. To further justify the novelty of our formulation and the necessity of treating both ϕ and $\mathbf{q}$ as primary unknowns, we highlight in Appendix A the ‘pitfall’ of deriving a convection-diffusion formulation using a single-field C-F diffusion model. We prove that this

formulation cannot maintain a consistent propagation speed for diffusion, which is non-physical. In contrast, our proposed formulation produces a material-invariant diffusive speed (see Section 5).

4 DISCRETIZATION

We discretize our two-field convection-diffusion formulation in equation (10) on a Cartesian MAC grid [Harlow and Welch 1965] using the splitting method [Stam 1999]. Specifically, we first explicitly update the convection terms

$$\frac{\phi^* - \phi^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \phi^n = W^n, \quad (11)$$

$$\frac{\mathbf{q}_C^* - \mathbf{q}_C^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{q}_C^n = 0, \quad (12)$$

followed by an implicit projection solve:

$$\frac{\phi^{n+1} - \phi^*}{\Delta t} + \nabla \cdot (\mathbf{q}_C^{n+1} + \mathbf{q}_F^{n+1}) = 0, \quad (13)$$

$$\frac{\mathbf{q}_C^{n+1} - \mathbf{q}_C^*}{\Delta t} + \frac{\mathbf{q}_C^{n+1}}{\tau} + (1 - F_T) \frac{k}{\tau} \nabla \phi^{n+1} = 0. \quad (14)$$

Taking the divergence of equation (14) and substituting the expression $\mathbf{q}_F = -F_T k \nabla \phi$ from equation (7) gives the following relation:

$$\left(1 + \frac{\Delta t}{\tau} \right) \nabla \cdot \mathbf{q}_C^{n+1} = \nabla \cdot \mathbf{q}_C^* - (1 - F_T) \frac{\Delta t}{\tau} k \nabla \cdot \nabla \phi^{n+1}. \quad (15)$$

Similarly, substituting the expression for $\mathbf{q}_F$ from equation (7) in equation (13) gives the following relation for $\nabla \cdot \mathbf{q}_C$ at time t^{n+1} :

$$\nabla \cdot \mathbf{q}_C^{n+1} = -\frac{\phi^{n+1} - \phi^*}{\Delta t} + F_T k \nabla \cdot \nabla \phi^{n+1}. \quad (16)$$

Finally, substituting equation (16) in equation (15) yields the following implicit Poisson-style system for ϕ :

$$\left[(\tau + \Delta t) \mathbf{I} - \left(\Delta t^2 + F_T \Delta t \tau \right) k \nabla \cdot \nabla \right] \phi^{n+1} = (\tau + \Delta t) \phi^* - \tau \Delta t \nabla \cdot \mathbf{q}_C^* \quad (17)$$

which can be efficiently solved with PCG. After solving for ϕ , we compute $\mathbf{q}_C$ at time t^{n+1} as follows:

$$\mathbf{q}_C^{n+1} = \frac{\tau}{\tau + \Delta t} \mathbf{q}_C^* - \frac{\Delta t}{\tau + \Delta t} (1 - F_T) k \nabla \phi^{n+1}. \quad (18)$$

Note that our solver reduces to a Fourier solver when $F_T = \tau = 0$.

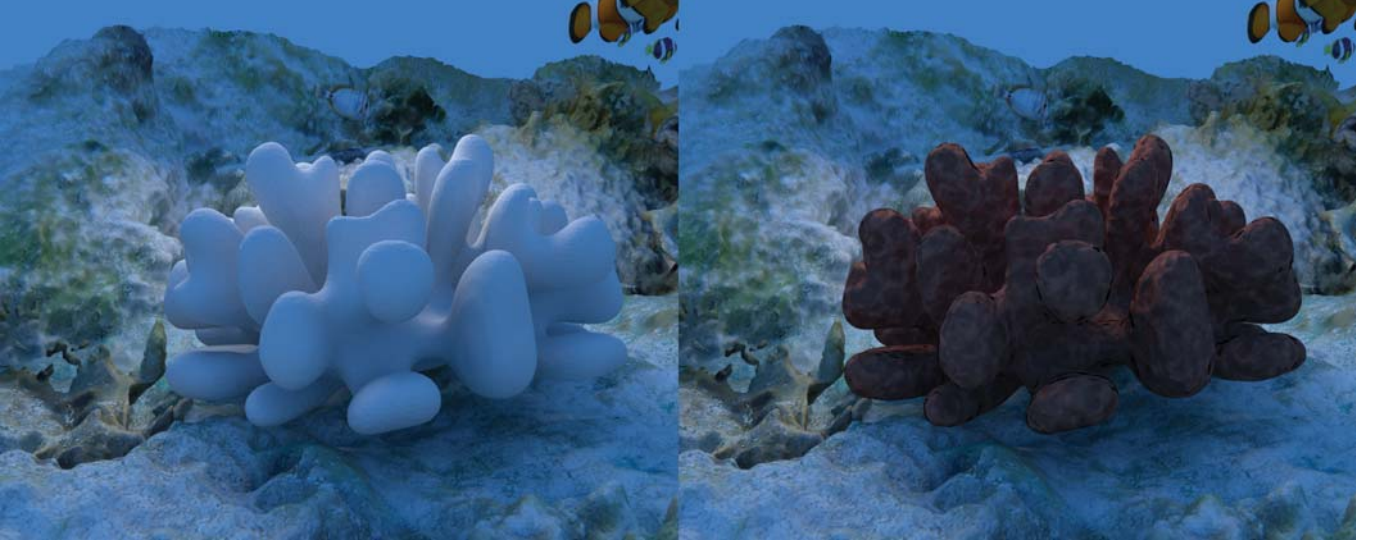


Fig. 7. A coral reef simulated using our generalized phase-field modeling approach from Section 7 (average computation time is 0.338s per time step). A bump map has been used on the right to mimic a life-like reef structure. The fishes are kinematically scripted to give the appearance of an underwater scenery.

4.1 Multigrid Solver

The implicit system in equation (17) is still *elliptic* [Trottenberg et al. 2001], although better-conditioned than a traditional Poisson system because of the presence of a scaled identity term on the diagonal, which would allow iterative Krylov solvers such as PCG to converge in relatively fewer iterations. For even faster convergence rates, we design a geometric Multigrid preconditioner, closely following [McAdams et al. 2010]. Our restriction and prolongation operators are exactly the same as those described in [McAdams et al. 2010]. For implementing the smoother, we pass down the constants τ , Δt , F_T and k to all levels in the Multigrid hierarchy to multiply with the system matrix in equation (17). As shown in Section 9, our non-Fourier solver achieves much better convergence on average, in comparison to a Multigrid solver for Fourier-based diffusion.

5 DIFFUSION TAXONOMY

Consider our convection-diffusion formulation (9) using the C-F constitutive model re-written in conservative flux form as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{G} = \mathbf{W}, \quad (19)$$

where $\mathbf{U}$ denotes the state vector, matrix $\mathbf{G}$ denotes the flux, and $\mathbf{W}$ is the vector of source terms (see Appendix B for a derivation in 3D). Defining G_i , as the i th column of $\mathbf{G}$, and using the chain rule gives:

$$\nabla \cdot \mathbf{G} = \frac{\partial G_i}{\partial U} \frac{\partial U}{\partial X_i} = A_1 \frac{\partial U}{\partial X_1} + A_2 \frac{\partial U}{\partial X_2} + A_3 \frac{\partial U}{\partial X_3}, \quad (20)$$

where $\mathbf{A} = [A_1, A_2, A_3]^T$ is the Jacobian of $\mathbf{G}$ with respect to $\mathbf{U}$ (see Appendix B for a derivation in 3D). Let $\boldsymbol{\vartheta} = [\vartheta_1, \vartheta_2, \vartheta_3]^T \in \mathbf{R}^3$ be an arbitrary vector, with $\|\boldsymbol{\vartheta}\| = 1$. Solving the eigenvalue problem:

$$\det(\lambda_k \mathbf{I} - \mathbf{A} \cdot \boldsymbol{\vartheta}) = 0 \quad (21)$$

yields the following seven real eigenvalues λ_k for the 3D case:

$$\begin{aligned} \lambda_1 = \lambda_2 = \lambda_3 &= 0; \quad \lambda_4 = \lambda_5 = \mathbf{u} \cdot \boldsymbol{\vartheta}; \\ \lambda_6 &= \mathbf{u} \cdot \boldsymbol{\vartheta} + \sqrt{\frac{(1 - F_T)k}{\tau}}; \quad \lambda_7 = \mathbf{u} \cdot \boldsymbol{\vartheta} - \sqrt{\frac{(1 - F_T)k}{\tau}}. \end{aligned} \quad (22)$$

This implies that there exist two characteristic speeds in our proposed convection-diffusion model, one is the velocity of the background material ($\mathbf{u}$), and the other is the propagation speed of diffusion ($c = \sqrt{(1 - F_T)k/\tau}$), which is also referred to as the *second sound speed* in the area of heat transfer [Chandrasekharaiah 1986; Sellitto et al. 2020]. Moreover, when $F_T = 0$, the diffusion speed c reverts to the classical wave speed of hyperbolic diffusion.

In other words, we assume that the physics of diffusion in a moving system, such as tea diffusing in water, is a competition between the material speed ($\mathbf{u}$) and the diffusive speed (c). To better describe this, we define the following non-dimensional number:

$$H_c = \frac{\|\mathbf{u}\|}{c}, \quad (23)$$

which can be used to classify the various diffusion types. Specifically:

$$\begin{aligned} H_c < 1 &\Leftrightarrow \text{Sub-critical Diffusion,} \\ H_c > 1 &\Leftrightarrow \text{Super-critical Diffusion,} \\ H_c = 1 &\Leftrightarrow \text{Critical Diffusion.} \end{aligned} \quad (24)$$

The role of H_c in diffusion is similar to the *Mach number* in compressible fluid dynamics [Anderson 2002], which characterizes how perturbations (or “shocks”) propagate in a medium. In particular, in the absence of Fourier-type fast processes (i.e., $F_T = 0$), the diffusive quantity can travel both downstream and upstream, corresponding to *sub-critical diffusion*. In contrast, under conditions of *super-critical diffusion*, the diffusive quantity can only travel downstream since the diffusive speed c is slower than the material speed $\mathbf{u}$. In the general non-Fourier case ($F_T < 1$) with $H_c > 1$, the diffusive pattern merges both Fourier-type and Cattaneo-type processes. That is to say, the diffusive quantity exists both downstream and upstream,

but a “conical regime” emerges with higher density concentration, with cone angle $\alpha = \sin^{-1}(c/\|\mathbf{u}\|) = \sin^{-1}(1/H_c)$ (see Figure 3).

6 COUPLING WITH CONTINUUM MECHANICS

For coupling our convection-diffusion formulation using the C-F diffusion model to solid mechanics, or incompressible flow, we use the following equation of motion along with equation (9):

$$\rho \frac{D\mathbf{u}}{Dt} - \nabla \cdot \boldsymbol{\sigma} = \mathbf{b}, \quad (25)$$

where D/Dt is the material derivative. For coupling with incompressible flow, we must also consider the divergence-free constraint $\nabla \cdot \mathbf{u} = 0$, and the stress tensor is defined as $\boldsymbol{\sigma} = -p\mathbf{I}$, where $\mathbf{I}$ is the second order identity tensor, and p is the pressure (note that we consider *inviscid* flow, omitting the term $\nabla \cdot \mu \nabla \mathbf{u}$ from the stress).

For coupling diffusion-driven dynamics with solid mechanics, the stress tensor is described by the elastic constitutive relation [Bonet and Wood 1997]:

$$\boldsymbol{\sigma} = \frac{1}{J} \frac{\partial \Psi}{\partial \mathbf{F}_E} \mathbf{F}_E^T, \quad (26)$$

where Ψ is the hyperelastic potential energy density, $\mathbf{F}_E$ is the elastic part of the deformation gradient $\mathbf{F}$ and $J = \det(\mathbf{F})$. We make the assumption that the *internal* stress driven by the change in saturation can be described by the simplified expression $\alpha\phi$, where ϕ is the quantity being diffused, and α is an amplification factor.

7 GENERALIZED PHASE-FIELD MODELING

Dendrite growth is a fascinating example of spontaneous pattern formation in nature. Accurate modeling and simulation of this phenomenon requires careful attention to the associated non-equilibrium dynamics. We first reformulate the classical phase-field theory for modeling dendritic growth [Kobayashi 1994], such that it can be integrated with our convection-diffusion formulation using the generalized C-F diffusion model. The governing equations are as follows:

$$\begin{aligned} \tau_\eta \frac{D\eta}{Dt} + \nabla \cdot \mathbf{q}_\eta &= \eta(1-\eta)(\eta + \psi - 0.5), \\ \mathbf{q}_\eta &= \mathbf{q}_\eta^C + \mathbf{q}_\eta^F, \quad \mathbf{q}_T = \mathbf{q}_T^C + \mathbf{q}_T^F, \\ \mathbf{q}_\eta^F + F_\eta \epsilon^2 \nabla \eta &= 0, \quad \mathbf{q}_T^F + F_T \nabla T = 0, \\ \tau_1 \frac{D\mathbf{q}_\eta^C}{Dt} + \mathbf{q}_\eta^C + (1-F_\eta) \epsilon^2 \nabla \eta &= 0, \\ \frac{DT}{Dt} + \nabla \cdot \mathbf{q}_T &= W + \frac{\partial \eta}{\partial t}, \\ \tau_2 \frac{D\mathbf{q}_T^C}{Dt} + \mathbf{q}_T^C + (1-F_T) \nabla T &= 0, \end{aligned} \quad (27)$$

where η is the phase-field value, $\mathbf{q}_\eta$ is the corresponding flux, ϵ^2 is the coefficient of diffusion for the phase-field, T is the temperature, $\mathbf{q}_T$ is the temperature flux, F_η and F_T are the Fourier-type fast process parameters for diffusing the phase-field and temperature, and $\tau_\eta, \tau_1, \tau_2$ are phase-field and diffusion parameters. The variable ψ is a function of T and v , defined as $\psi(T, v) = -(\alpha/\pi) \tan^{-1}(\gamma\beta(v)T)$, where $v = -\nabla \eta$, $\beta(v) = 1 - \delta(1 - a/b)$, α, δ and γ are user-defined parameters, $a = \sum_{i=1}^3 v_i^4$, and $b = \left(\sum_{i=1}^3 v_i^2\right)^2$. Note that the traditional phase-field equation with Fick’s diffusion law is a parabolic

system. However, our proposed enhancement to the phase-field equation with the generalized C-F diffusion model makes it a *hyperbolic* system for the specific case when $F_\eta = F_T = 0$, allowing our formulation to generate a wider range of patterns for dendrites than previously possible (see Figure 8 for examples).

7.1 Anisotropic Phase-field Modeling for Snowflakes

Motivated by the recent work of [Demange et al. 2017], we propose the following model for simulating snowflakes using the generalized C-F diffusion model, that also accommodates anisotropic growth:

$$\begin{aligned} \tau_\eta \frac{D\eta}{Dt} + \nabla_\star \cdot \mathbf{q}_\eta &= \delta^2 \nabla_\star \cdot [\|v\|^2 A(v)D] + \eta(1-\eta)(\eta + \psi - 0.5), \\ \mathbf{q}_\eta &= \mathbf{q}_\eta^C + \mathbf{q}_\eta^F, \quad \mathbf{q}_T = \mathbf{q}_T^C + \mathbf{q}_T^F, \\ \mathbf{q}_\eta^F + F_\eta A^2(v) \nabla_\star \eta &= 0, \quad \mathbf{q}_T^F + F_T \nabla_\star T = 0, \\ \tau_1 \frac{D\mathbf{q}_\eta}{Dt} + \mathbf{q}_\eta + (1-F_\eta) A(v)^2 \nabla_\star \eta &= 0, \\ \frac{DT}{Dt} + \nabla_\star \cdot \mathbf{q}_T &= W + \frac{\partial p}{\partial t}, \\ \tau_2 \frac{D\mathbf{q}_T^C}{Dt} + \mathbf{q}_T^C + (1-F_T) \nabla_\star T &= 0, \end{aligned} \quad (28)$$

where $\eta, T, \mathbf{q}_\eta, \mathbf{q}_T, \tau_\eta, \tau_1, \tau_2, F_\eta, F_T, \delta$ and v have the same definitions as that in equation (27). The differential operator $\nabla_\star$ is defined as $\nabla_\star = [\partial/\partial x_1, \partial/\partial x_2, \Gamma \partial/\partial x_3]^T$, where Γ is a parameter that controls the vertical/horizontal preference of diffusion. The variable A determines the orientation of the snowflake and is a function of v_i , defined as $A(v) = 1 + \epsilon_p \cos(\omega\theta(v)_p) + \epsilon_a \cos(\zeta\theta(v)_a)$, where $\theta(v)_p = \tan^{-1}(v_2/v_1)$ and $\theta(v)_a = -\tan^{-1}\left(\sqrt{v_1^2 + v_2^2}/v_3\right)$ represent the polar and azimuthal angles, respectively. $A(v)$ accounts for both the horizontal ω -fold symmetry, and the vertical planar ζ -fold symmetry of snowflakes, ϵ_p and ϵ_a are anisotropy constants, and ψ is a function of T , defined as $\psi(T) = -(\alpha/\pi) \tan^{-1}(\gamma T)$, where α and γ are user-defined parameters. The vector $D = \partial A(v)/\partial v$.

Table 1. Physical quantities stored on particles and grid nodes.

Particle	Description	Grid
$\mathbf{x}_p$	position	$\mathbf{x}_i$
$\mathbf{v}_p$	velocity	$\mathbf{v}_i$
$\mathbf{F}_p$	deformation gradient	—
—	force	$\mathbf{f}_i$
V_p	volume	—
m_p	total mass	m_i
m_p^f	mass of fluid	m_i^f
m_p^s	fully saturated mass of fluid	m_i^s
ϕ_p	porosity	ϕ_i
S_p	saturation	S_i
$\mathbf{q}_p^C$	C-process diffusion flux	$\mathbf{q}_i^C$

8 INTEGRATION WITH MPM

We now describe the integration of our convection-diffusion formulation (9) using the C-F diffusion model [Anderson and Tamma

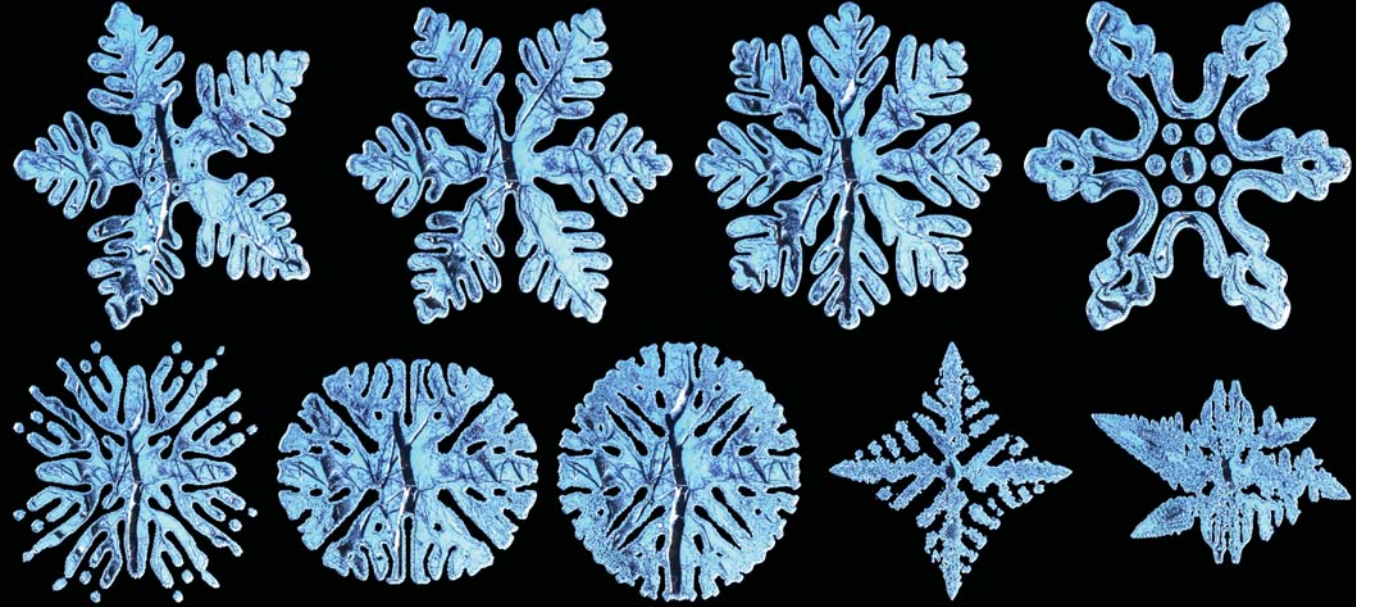


Fig. 8. Our generalized diffusion solver generates snowflake patterns that can be described by a traditional Fourier-type diffusion solver (top row), but can also capture patterns that cannot be described by the traditional Fourier modeling approach (bottom row), as in this case the governing equations become purely hyperbolic (corresponding to the case $F_T = 0$) (256^2 grid). However, as pointed out in Section 9, some of these patterns are indeed observed in the real world.

2006; Xue et al. 2018] with the Material Point Method (MPM) to simulate diffusion-driven poroelastic material dynamics. Our method is inspired by the recent work of Ding et al. [2019], with new modifications to accommodate our non-Fourier diffusion solver.

We reserve subscripts p, q, r for quantities stored on particles, and subscripts i, j, k for quantities stored on grid nodes. Table 1 summarizes the various quantities stored on particles and grid nodes. We treat each particle p as *porous*, where the porosity ϕ_p is defined as the volume fraction of the interconnected void space at p . Thus, $\phi_p V_p$ represents the void particle volume. Each particle can hold an absorbed fluid mass $m_p^f \leq m_p^s = \rho_f \phi_p V_p$, where ρ_f is the fluid density. The particle saturation $S_p = m_p^f / m_p^s$.

8.1 Algorithm Description

We now describe our method in detail. Figure 9 gives a high-level overview. At the beginning of each time step, we rasterize quantities $m_p, \phi_p, \nabla \cdot \mathbf{q}_p^C, S_p$, and V_p from particles to the grid as follows:

$$\begin{aligned} \phi_i^n &= \sum_p \phi_p^n w_{ip}^n, & m_i^n &= \sum_p (S_p^n m_p^s \phi_p^n + m_{p0}) w_{ip}^n, \\ \mathbf{v}_i^n &= \sum_p \mathbf{v}_p^n m_p w_{ip}^n / m_i^n, & (\nabla \cdot \mathbf{q}_i^C)^n &= \sum_p (\nabla \cdot \mathbf{q}_p^C)^n w_{ip}^n, \end{aligned} \quad (29)$$

where w_{ip} is the quadratic B-spline weight of particle p at grid node i . Similar to the velocity normalization in equation (29) to conserve momentum, following [Stomakhin et al. 2013], we rasterize the saturation to conserve the fluid mass in particles, as follows:

$$\begin{aligned} V_i^n &= \sum_p V_p^n J_p^n w_{ip}^n, & (m_i^f)^n &= \sum_p S_p^n m_p^s \phi_p^n J_p^n w_{ip}^n, \\ (m_i^s)^n &= \sum_p (m_p^s)^n J_p^n w_{ip}^n, & S_i^n &= (m_i^f)^n / (m_i^s)^n, \end{aligned} \quad (30)$$

where J_p denotes the change in the particle volume, i.e., $J_p = \|F_p\|$, and F_p is the deformation gradient. The particle volumes V_p are computed in the very first time step, as described in [Stomakhin et al. 2013]. After rasterizing the diffusive quantities, the saturation and flux values are updated to time t^* as follows:

$$\begin{aligned} S_i^* &= S_i^n - \Delta t \mathbf{v}_i^n \cdot \nabla S_i^n, \\ (\mathbf{q}_i^C)^* &= (\mathbf{q}_i^C)^n - \Delta t \mathbf{v}_i^n \cdot \nabla (\mathbf{q}_i^C)^n. \end{aligned} \quad (31)$$

Subsequently, the stress is computed as:

$$\boldsymbol{\sigma}_p = 2\mu \left(\mathbf{F}_{E_p}^n - \mathbf{R}_{E_p}^n \right) \left(\mathbf{F}_{E_p}^n \right)^T + \lambda (J_E - 1) J_E \mathbf{I} - \chi S_p^n \mathbf{I}, \quad (32)$$

where $\mathbf{F}_{E_p}^n$ is the elastic part of the deformation gradient, and $\mathbf{R}_{E_p}$ is obtained by the polar decomposition of $\mathbf{F}_{E_p} = \mathbf{R}_{E_p} \mathbf{S}_{E_p}$. Following [Stomakhin et al. 2014], μ and λ are defined as $\mu = \mu_0 e^{\xi(1-J_{E_p})}$, $\lambda = \lambda_0 e^{\xi(1-J_p)}$, $J_{E_p} = \det \mathbf{F}_{E_p}$, and $J_p = \det \mathbf{F}_p$, where μ_0 and λ_0 are the Lamé parameters, ξ is a hardening parameter, and χ is an amplification coefficient linking saturation and internal stress. Then the internal stress-based force per grid node is computed as $\mathbf{f}_i^n = -\sum_p V_p \boldsymbol{\sigma}_p \nabla w_{ip}^n$, and used to update the nodal velocity as

$$\mathbf{v}_i^* = \mathbf{v}_i^n + \Delta t (\mathbf{f}_i^n + \mathbf{g}) / m_i, \quad (33)$$

where $\mathbf{g}$ is the acceleration due to gravity. Collisions with objects are treated as described in [Stomakhin et al. 2013]. For symplectic Euler integration, we set $\mathbf{v}_i^{n+1} = \mathbf{v}_i^*$. We compute S_i^{n+1} and $(\mathbf{q}_i^C)^{n+1}$ using our non-Fourier solve in equations (17) and (18), as follows:

$$\begin{aligned} \left[(\tau + \Delta t) \mathbf{I} - \left(\Delta t^2 + F_T \Delta t \tau \right) k \nabla \cdot \nabla \right] S_i^{n+1} &= (\tau + \Delta t) S_i^* - \tau \Delta t \nabla \cdot (\mathbf{q}_i^C)^*, \\ (\mathbf{q}_i^C)^{n+1} &= \frac{\tau}{\tau + \Delta t} (\mathbf{q}_i^C)^* - (1 - F_T) \frac{\Delta t}{\tau + \Delta t} k \nabla S_i^{n+1}. \end{aligned} \quad (34)$$

For semi-implicit time integration, $\mathbf{v}_i^{n+1}$ is computed as

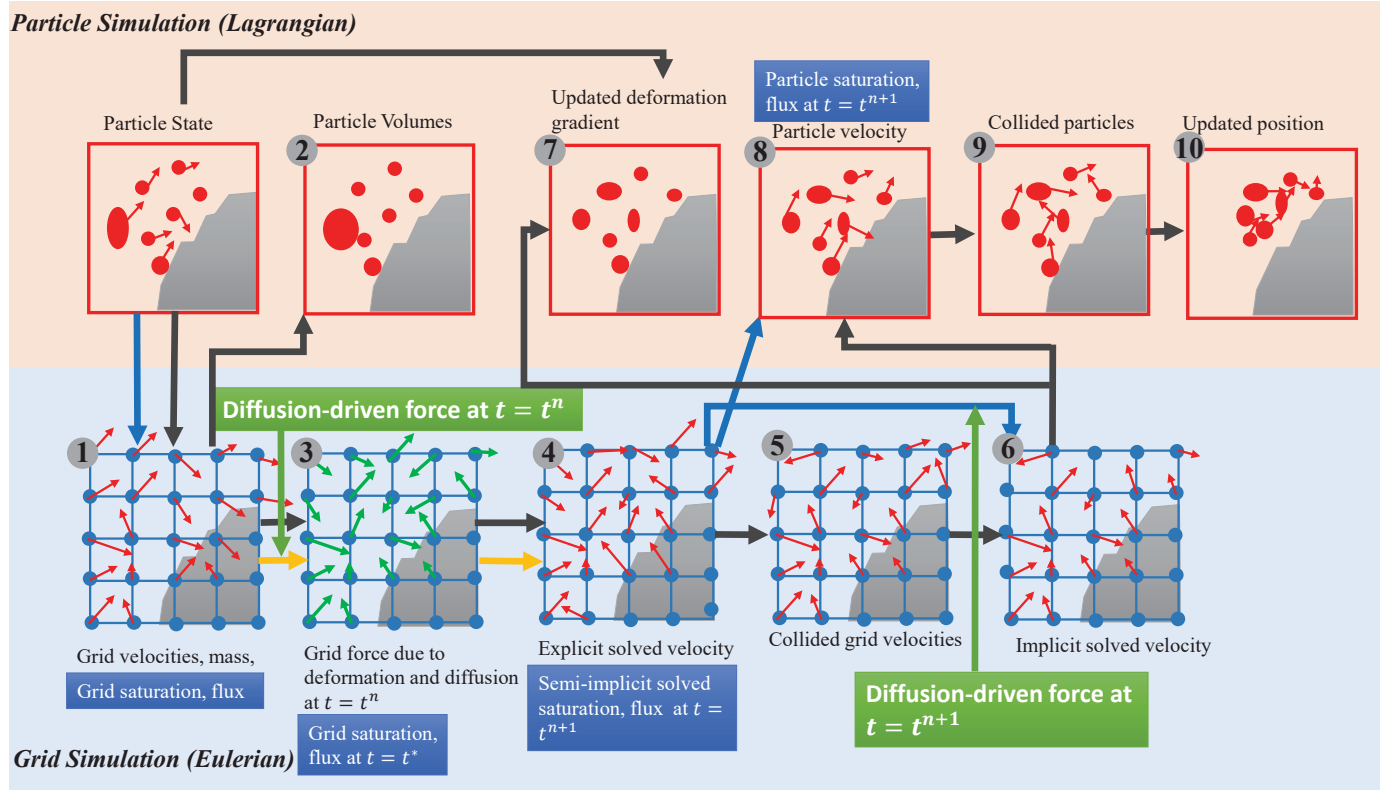


Fig. 9. Overview of our proposed method in conjunction with the Material Point Method (MPM) for simulating diffusion-driven poroelastic material dynamics. The short red and green arrows represent the velocity and force vectors at the nodes of the background grid. The blue boxes represent the different steps that involve diffusion processes. The green boxes illustrate the particular interventions where we add diffusion-driven forces in the MPM simulation framework.

$$\mathbf{v}_i^{n+1} = \mathbf{v}^* + \Delta t \beta (\mathbf{f}_i^{n+1} - \mathbf{f}_i^n) / m_i, \quad (35)$$

where $\mathbf{f}_i^{n+1} = \mathbf{f}_d^{n+1} + \chi \nabla S_i^{n+1}$ and $\mathbf{f}_d^{n+1}$ is the part of internal force due to elasticity [Stomakhin et al. 2013]. The parameter $\beta \in [0, 1]$ allows for transitioning between fully explicit and semi-implicit integration. We update the deformation gradient $\mathbf{F}_p^{n+1}$ and $(\nabla \cdot \mathbf{q}_p^C)^{n+1}$ based on $\mathbf{v}_p^{n+1}$ and $(\mathbf{q}_p^C)^{n+1}$, and compute $\mathbf{F}_{E_p}^{n+1}$ and $\mathbf{F}_{P_p}^{n+1}$. Then $\mathbf{v}_p^{n+1}$, S_p^{n+1} , and $(\mathbf{q}_p^C)^{n+1}$ are updated considering the PIC and FLIP components as follows:

$$\begin{aligned} \kappa^{n+1} &= (1 - \alpha) \kappa_{\text{PIC}}^{n+1} + \alpha \kappa_{\text{FLIP}}^{n+1}, \\ \kappa_{\text{PIC}}^{n+1} &= \sum_i \kappa_i^{n+1} w_{ip}^n, \quad \kappa_{\text{FLIP}}^{n+1} = \kappa^n + \sum_i (\kappa_i^{n+1} - \kappa_i^n) w_{ip}^n, \end{aligned} \quad (36)$$

where κ is a generalized variable, and the parameter $\alpha \in [0, 1]$ allows for transitioning between fully PIC and fully FLIP update formulas. Once post-collision velocities $\mathbf{v}_p^{n+1}$ have been computed, we use them to update the particle positions $\mathbf{x}_p^{n+1}$.

9 RESULTS

Timings for the simulations in Figures 2 and 5, including machine specifications, are summarized in Table 2. The better conditioning of our non-Fourier system in equation (17) allows the diffusion step to be $1.04 - 5.62\times$ faster, compared to the standard Fourier model.

When $F_T = 1$, our model reduces to the standard Fourier model and thus, has equivalent computational cost as Fourier-based diffusion.

9.1 Implementation Details

Accompanying this article, we open-source our code for running all examples with our generalized diffusion solver. We use SPGrid [Setaluri et al. 2014] as the sparse grid structure in our simulator, with 16 channels for incompressible flow (Figure 5), and 32 channels for MPM and phase-field examples (Figures 2, 7 and 8). Our Multigrid solver only uses 4 channels. Since the background grid is used as a scratchpad in MPM, we rasterize the grid within a slightly thickened bounding box of the particles, saving memory overhead.

9.2 Diffusion Patterns in 2D

To demonstrate the effect of the parameters F_T and H_c in equations (7) and (24), we simulated a point source that diffuses material in a rectangular 2D domain with uniform background velocity $\mathbf{u}$. Figure 4 shows the effect of varying the thermal parameter F_T . When $F_T = 1$, the diffusive quantity exists both upstream and downstream. However, as F_T decreases, Cattaneo's diffusion process starts to dominate and the diffusive regime is gradually constrained to a "conical" region when $F_T = 0$. To investigate this limited diffusive regime, we set $F_T = 0$ and vary H_c from critical diffusion to super-critical diffusion. As shown in Figure 3, the cone angle in the diffusive region is consistent with the value $\sin^{-1}(1/H_c)$, as proposed by our model.

Table 2. Timing information for four cases. All run times cited are in seconds. Specific Notations: **F**: Fourier-based diffusion model, **NF**: non-Fourier diffusion model, **DM**: Diffusion model, F_T : Diffusion parameters as described in equation (7), **DiffQC**: Diffusive quantities convection, **VA**: Velocity Advection, **CFluxAU**: C-Flux (q_C) advection & update, **SM**: Source modification, **MEM**: Memory usage in GB, **MPMSol**: MPM solver runtime, **TotDiffTime**: Total diffusion time, **PCG Iteration**: Preconditioned Conjugate Gradients iterations, **TotTime**: Total compute time. For simplicity, we organize the CPU time for all the computational procedures and show the major steps, particularly, the MPMSol time includes all the computational cost of pure evaluation of deformation, and TotDiffTime sums up all the procedures that involve our proposed non-Fourier diffusion solver. In smoke/ink and hydrogel swelling simulations, our proposed non-Fourier solver shows great computational performance (less PCG iterations and Total Time) in comparison to classical Fourier-based diffusion.

	Smoke/Ink Simulations ($256 \times 512 \times 256$)						Hydrogel Swelling ($512^3 + 4 \times 10^5$)					
	Intel®Xeon®Gold 6128 CPU @ 3.40GHz						Intel®Xeon® CPU E5-1620 v4 @ 3.50GHz					
	Figure 5 (a)-(f)						Figure 6		Figure 2 (Top)		Figure 2 (Bottom)	
DM	NF			F			NF	F	NF	F	NF	F
F_T	0	0.25	0.5	0.75	1	N/A	0	N/A	0	N/A	0	N/A
DiffQC	1.00	0.63	0.62	0.66	0.69	0.59	121.71	302.16	N/A	N/A	N/A	N/A
VA	4.90	4.99	4.91	4.90	4.88	4.88	N/A	N/A	N/A	N/A	N/A	N/A
CFluxAU	6.01	4.82	4.89	4.97	4.82	N/A	7.7	N/A	N/A	N/A	N/A	N/A
SM	0.14	0.14	0.14	0.14	0.14	0.13	0.15	0.15	N/A	N/A	N/A	N/A
Projection	168.10	167.27	166.01	160.19	164.08	164.55	N/A	N/A	N/A	N/A	N/A	N/A
Threads	8	8	8	8	8	8	8	8	8	8	8	8
MEM	3.456	3.456	3.456	3.456	3.456	3.456	0.576	0.576	0.192	0.224	0.192	0.192
MPMSol	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	5.43	13.2	7.21	16.3
TotDiffTime	98.93	131.06	166.99	185.66	193.61	196.01	16.92	297.18	0.03997	0.224	0.03807	0.118552
PCG Iteration	8	28	38	43	45	45	8	42	1	5	1	5
TotTime	279.52	309.36	343.98	362.96	368.66	366.45	131.07	303.68	5.47	13.46	5.78	16.42

9.3 Ink Dispersion and Smoke Simulation

We simulated ink dispersing in water, as shown in Figure 5. Our method produces a more vivid behavior, similar to the observed behavior in the real world¹, in contrast to incompressible flow with no diffusion, and a traditional Fourier-type diffusion solver, which produces an overly diffusive profile. To highlight differences in the super-critical diffusive regime, Figure 6 shows smoke diffusion in a medium with uniform background velocity. The traditional Fourier-type solver propagates smoke both in the upstream and downstream directions, dissipating significantly in the process. In contrast, our generalized diffusion solver produces a limited “conical” diffusion region, as predicted by our diffusion taxonomy derived in Section 5.

9.4 Hydrogel Swelling

We simulated poroelastic material dynamics of a hydrogel that are driven by fluid absorption, by integrating our convection-diffusion formulation using the generalized C-F diffusion model with MPM, as described in Section 8. As fluid diffuses through the solid, its saturation level affects material elasticity, making the resulting dynamics an interaction between the diffusion wave and stress wave. As shown in Figure 2, our method produces rich anisotropic 3D deformations that are not captured with Fick’s law. Moreover, our method can qualitatively reproduce the characteristic folding behavior of the hydrogel that is observed in the real world, as shown in Figure 10 (see here² for a video of the real world experiment).

9.5 Snowflake Formation and Coral Reef Growth

Using our generalized diffusion solver, we simulated growth patterns for snowflakes (256×256 grid) starting from an initial circular shape,

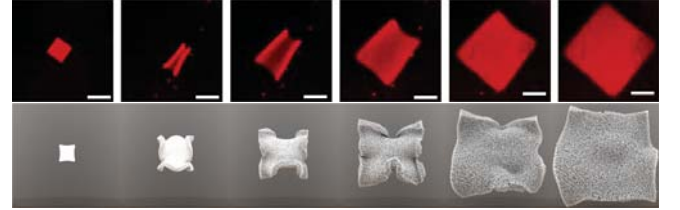


Fig. 10. (Top) Time-lapse fluorescence micrographs of a hydrogel in polymerizing hairpins (reproduced with permission from [Cangialosi et al. 2017]). (Bottom) Simulated hydrogel using our proposed convection-diffusion formulation (9) using the generalized C-F diffusion model. Our method can qualitatively reproduce the folding behavior observed in the real world, while the standard Fourier model produces uniform expansion (see Figure 2(top)).

obtaining symmetric fern-like shapes that can also be described by a traditional Fourier-type diffusion solver (see Figure 8 (top row)), but also more vivid and asymmetric ones that cannot be described by the traditional Fourier-based modeling approach (see Figure 8 (bottom)), as they correspond to the case $F_\eta = F_T = 0$, when the governing equations become purely hyperbolic. However, as shown in Figure 12, such kinds of simulated growth patterns are indeed observed in the real world [Libbrecht 2005a] (see also here³ for the real world snowflake photos). For this example, we set $\tau_1 = 1$ and $\tau_2 = 10^{-4}$, obtaining significant non-Fourier effects in the phase-field in comparison to the temperature. The 6-fold branch pattern that is preset in our formulation (and obtained with standard Fourier-type diffusion, see Figure 11(far right)) degenerates to 4 branches.

In our simulations, we have observed that such large differences in the values of τ_1 and τ_2 can result in various anisotropies. To empirically verify this hypothesis, we conducted a parameter study for this example. We set $F_\eta = F_T = 0$ and ran simulations where

¹<https://www.youtube.com/watch?v=bWgGvANnO0U>

²<https://www.youtube.com/watch?v=dUEQ4lV0iyQ>

³<http://snowcrystals.com/>

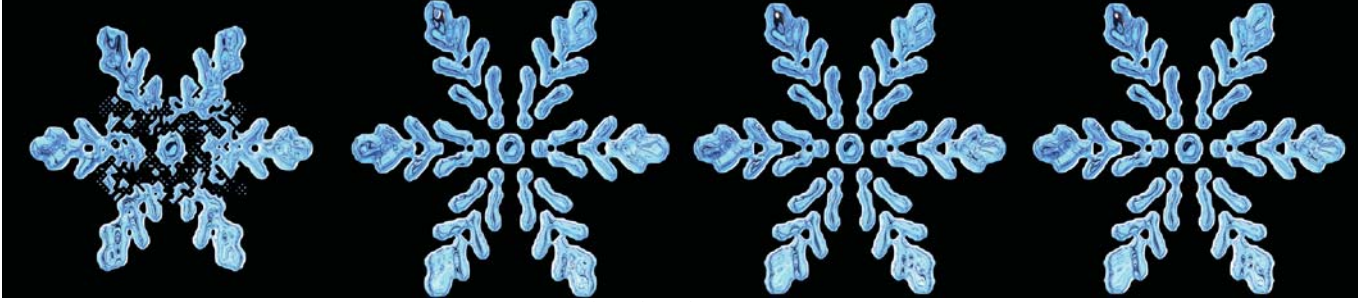


Fig. 11. The first three patterns use $F_\eta = F_T = 0$ and $\tau_1 = \tau_2 \in \{10^{-3}, 10^{-4}, 10^{-5}\}$ in Eq. (28). The last pattern was obtained with the standard Fourier model.

$\tau_1 = \tau_2$ and also where $\tau_1 \neq \tau_2$. In the former case, particularly when these values are small, the deviations from standard Fourier-based diffusion are minor (see Figure 11). This observation is also in agreement with our diffusion taxonomy (see Section 5), the diffusive speed for both phase-field and temperature are very high when τ_1, τ_2 are small, such that the resulting snowflake growth is very close to the standard Fourier model, where the diffusive speed is infinite.

When $\tau_1 \neq \tau_2$, we fix $\tau_2 = 10^{-4}$ and only increase τ_1 to observe non-Fourier effects due to phase-field values. The growth of branches is limited, and the principal branches in the predefined 6-fold pattern degenerates to 4 when $\tau_1 = 10^{-2}$. However, our model provides a wide range of snowflake patterns and the results are supported by real world data (see the case $\tau_1 = 10^{-3}$ in Figure 13).

Figure 7 shows a coral reef ($120^2 \times 200$ grid) that was simulated using our diffusion solver, producing a growth pattern that is similar to those observed in the real world (see here⁴ for a real image).

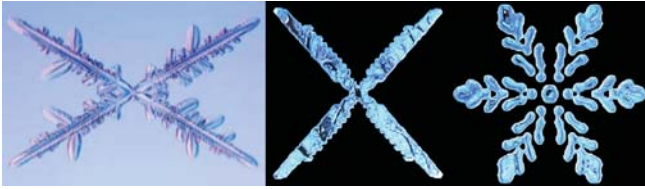


Fig. 12. (Left) Real world snowflake (reproduced with permission from [Libbrecht 2019]). (Middle) Simulated snowflake using our generalized diffusion solver with $F_\eta = F_T = 0$, $\tau_\eta = 10^{-5}$, $\tau_1 = 1$, $\tau_2 = 10^{-4}$, $\delta = 10^{-2}$, $\gamma = 10$, $\alpha = 0.9$, $\omega = 6$, $\epsilon_p = 0.02$, $\epsilon_a = 0$, and $\Gamma = 0$. (Right) Snowflake simulated using traditional Fourier-based diffusion. This example highlights that the C-F diffusion model can describe snowflake patterns that are observed in the real world, but which cannot be described by Fourier-based diffusion.

9.6 Snowfall Simulation

To showcase the various snowflake patterns obtained using our generalized phase-field modeling approach, we used them as textures for animating a procedural simulation of snowfall (see Figure 1). Note that more comprehensive simulation approaches are required for capturing effects such as *snow accumulation* [Gissler et al. 2020].

10 CONCLUSION AND FUTURE WORK

We proposed a novel two-field convection-diffusion formulation to describe diffusion-driven phenomena with the C-F diffusion model, which is a strict generalization of Fourier's law. To simulate the proposed model on staggered MAC grids, we proposed a novel semi-implicit discretization that leads to an implicit linear system that

has better conditioning compared to the Fourier system, and allows for efficient numerical solutions using Multigrid-preconditioned Conjugate Gradients. Our method easily integrates with MPM and phase-field modeling, and allows for relatively large time steps.

Our method presents a more fundamental and physical interpretation of the basic convection-diffusion phenomena in nature, that rely heavily on the interaction between the diffusion process and the background material dynamics. A limited diffusion regime was proven to exist when the propagation speed of diffusion is smaller than the speed of the background material, as shown by our smoke simulations with uniform background flow. We also demonstrated that our method captures a wide range of diffusion processes and produces more detailed and realistic diffusion-driven effects in practical applications than Fourier's law, such as anisotropic dynamic swelling of an elastic hydrogel immersed in water, sharper ink profile, and rich dendrite and snowflake growth patterns.

There are many interesting avenues for future work. Since our method describes a hyperbolic problem (wave propagation), numerical oscillations exist near the discontinuous wave front. Investigation into high order accurate upwind discretization techniques, such as weighted essentially non-oscillatory (WENO) schemes [Jiang and Shu 1996], and their integration with MPM, would be useful. This may also help mitigate the memory overhead of our two-field convection-diffusion formulation that comes with the use of high resolution grids, which is roughly four times that of the traditional single-field Fourier diffusion model. The parameters F_T , τ , and H_c provide a mechanism to two-way couple diffusion dynamics with the background material dynamics, in stark contrast to Fourier-based diffusion where only one-way coupling is achieved. However, the resulting effect very much depends on the problem being simulated and domain knowledge. It would be interesting to find parameters that are more intuitive to control and allow for desired visual results. Our focus in this paper was on simulating dissipative dynamics. The diffusion step happens after advection, and is not conservative in general because this constraint is not being explicitly enforced in our Eulerian framework. Thus, it would be beneficial to design schemes that conserve mass with non-Fourier diffusion. Finally, our snowflake simulations would benefit from the introduction of more advanced orientation field models, such as those proposed in [Ren et al. 2018], that allow the artist to customize the crystal growth by controlling its growth directions and final shape.

⁴<https://www.jpl.nasa.gov/news/news.php?feature=6793>

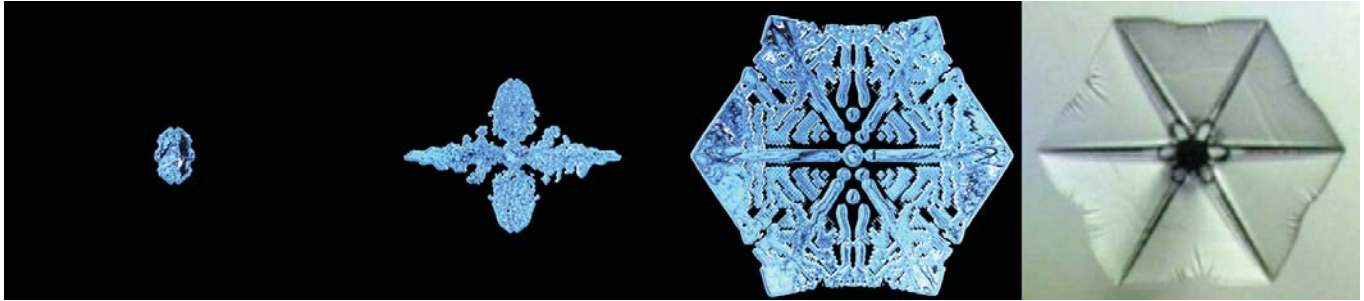


Fig. 13. The first three patterns are simulations which use $F_\eta = F_T = 0$, $\tau_2 = 10^{-4}$ and $\tau_1 \in \{10^{-1}, 10^{-2}, 10^{-3}\}$ in Eq. (28), with the same formulation as that in Figure 11. The rightmost image is another example of a snowflake pattern observed in the real world (reproduced with permission from [Libbrecht 2005b]).

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their feedback, and the Mel Fisher Maritime Museum for open-sourcing their bedrock model under a CC license (used in Figure 7). This research was supported in part by NSF CAREER (IIS-1943199), CCF-1813624, ECCS-2023780, DOE ORNL contract 4000171342, the Rutgers University start-up grant, and the Ralph E. Powe Junior Faculty Enhancement Award.

REFERENCES

- Christianne VDR Anderson and Kumar K Tamma. 2006. Novel heat conduction model for bridging different space and time scales. *Physical review letters* 96, 18 (2006), 184301.
- John Anderson. 2002. *Modern Compressible Flow: With Historical Perspective*. McGraw-Hill Education; 3 edition.
- KJ Baumeister and TD Hamill. 1969. Hyperbolic heat-conduction equation—a solution for the semi-infinite body problem. (1969).
- Javier Bonet and Richard D Wood. 1997. *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press.
- Jose A. Canabal, Miguel A. Otaduy, Byungmoon Kim, and Jose Echevarria. 2020. Simulation of Dendritic Painting. *Computer Graphics Forum* 39, 2 (2020).
- A. Cangialosi, C. Yoon, J. Liu, Q. Huang, J. Guo, T. D Nguyen, D. H Gracias, and R. Schulman. 2017. DNA sequence-directed shape change of photopatterned hydrogels via high-degree swelling. *Science* 357, 6356 (2017), 1126–1130.
- Mark Carlson, Peter J Mucha, R Brooks Van Horn III, and Greg Turk. 2002. Melting and flowing. In *Symposium on Computer Animation*. 167–174.
- Carlo Cattaneo. 1948. Sulla conduzione del calore. *Atti Sem. Mat. Fis. Univ. Modena* 3 (1948), 83–101.
- DS Chandrasekharaiiah. 1986. Thermoelasticity with second sound: a review. (1986).
- Gang Chen. 2001. Ballistic-diffusive heat-conduction equations. *Physical Review Letters* 86, 11 (2001), 2297.
- Hsiao-Yu Chen, Arnav Sastry, Wim M van Rees, and Etienne Vouga. 2018. Physical simulation of environmentally induced thin shell deformation. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–13.
- Nelson S-H Chu and Chiew-Lan Tai. 2005. MoXi: real-time ink dispersion in absorbent paper. *ACM Transactions on Graphics (TOG)* 24, 3 (2005), 504–511.
- Pascal Clausen, Martin Wicke, Jonathan R Shewchuk, and James F O'Brien. 2013. Simulating liquids and solid-liquid interactions with lagrangian meshes. *ACM Transactions on Graphics (TOG)* 32, 2 (2013), 1–15.
- Cassidy J Curtis, Sean E Anderson, Joshua E Seims, Kurt W Fleischer, and David H Salesin. 1997. Computer-generated watercolor. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 421–430.
- G. Demange, H. Zapolsky, R. Patte, and M. Brunel. 2017. A phase field model for snow crystal growth in three dimensions. *Computational Materials* 3, 1 (2017), 1–7.
- Mengyuan Ding, Xuchen Han, Stephanie Wang, Theodore F Gast, and Joseph M Teran. 2019. A thermomechanical material point method for baking and cooking. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 192.
- Y. Fei, C. Batty, E. Grinspun, and C. Zheng. 2018. A multi-scale model for simulating liquid-fabric interactions. *ACM Transactions on Graphics* 37, 4 (2018), 1–16.
- Adolph Fick. 1855. V. On liquid diffusion. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 10, 63 (1855), 30–39.
- Jean-Baptiste Joseph Fourier. 1878. *Théorie analytique de la chaleur*. Paris 1822. *Engl. Übersetzung: "Theory of Heat Transfer", Cambridge* (1878).
- Ming Gao, Andre Pradhana, Xuchen Han, Qi Guo, Grant Kot, Eftychios Sifakis, and Chenfanfu Jiang. 2018a. Animating fluid sediment mixture in particle-laden flows. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–11.
- Ming Gao, Xinlei Wang, Kui Wu, Andre Pradhana, Eftychios Sifakis, Cem Yuksel, and Chenfanfu Jiang. 2018b. GPU optimization of material point methods. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–12.
- Christoph Gissler, Andreas Henne, Stefan Band, Andreas Peer, and Matthias Teschner. 2020. An Implicit Compressible SPH Solver for Snow Simulation. *ACM Trans. Graph.* 39, 4, Article 36 (2020), 16 pages.
- Francis H. Harlow and J. Eddie Welch. 1965. Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *Physics of Fluids* 8, 12 (1965), 2182–2189.
- Xiaowei He, Huamin Wang, Fengjun Zhang, Hongan Wang, Guoping Wang, Kun Zhou, and Enhua Wu. 2015. Simulation of fluid mixing with interface control. In *Symposium on Computer Animation*. 129–135.
- Markus Huber, Simon Pabst, and Wolfgang Straßer. 2011. Wet cloth simulation. In *ACM SIGGRAPH 2011 Posters*. 1–1.
- K. Iwasaki, H. Uchida, Y. Dobashi, and T. Nishita. 2010. Fast particle-based visual simulation of ice melting. In *Computer graphics forum*, Vol. 29. 2215–2223.
- Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. 2016. The material point method for simulating continuum materials. In *ACM SIGGRAPH 2016 Courses*. 1–52.
- Guang-Shan Jiang and Chi-Wang Shu. 1996. Efficient Implementation of Weighted ENO Schemes. *J. Comput. Phys.* 126, 1 (1996), 202 – 228.
- Daniel D Joseph and Luigi Preziosi. 1989. Heat waves. *Reviews of Modern Physics* 61, 1 (1989), 41.
- N. Kang, J. Park, J. Noh, and S. Y Shin. 2010. A hybrid approach to multiple fluid simulation using volume fractions. In *Computer Graphics Forum*, Vol. 29. 685–694.
- Theodore Kim, David Adalsteinsson, and Ming C Lin. 2006. Modeling ice dynamics as a thin-film stefan problem. In *Symposium on Computer animation*. 167–176.
- Theodore Kim, Michael Henson, and Ming C Lin. 2004. A hybrid algorithm for modeling ice formation. In *Symposium on Computer Animation*. 305–314.
- Theodore Kim and Ming Lin. 2007. Stable advection-reaction-diffusion with arbitrary anisotropy. *Computer Animation and Virtual Worlds* 18, 4-5 (2007), 329–338.
- Theodore Kim and Ming C Lin. 2003. Visual simulation of ice crystal growth. In *Symposium on Computer Animation*. Eurographics Association, 86–97.
- Ryo Kobayashi. 1994. A numerical approach to three-dimensional dendritic solidification. *Experimental mathematics* 3, 1 (1994), 59–81.
- Toon Lenaerts, Bart Adams, and Philip Dutré. 2008. Porous flow in particle-based fluid simulations. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 1–8.
- Kenneth G Libbrecht. 2005a. The physics of snow crystals. *Reports on progress in physics* 68, 4 (2005), 855.
- Kenneth G Libbrecht. 2005b. The physics of snow crystals. *Reports on progress in physics* 68, 4 (2005), 855.
- Kenneth G Libbrecht. 2019. Snow crystals. *arXiv preprint arXiv:1910.06389* (2019).
- W-C Lin. 2014. Coupling hair with smoothed particle hydrodynamics fluids. (2014).
- Wei-Chin Lin. 2015. Boundary handling and porous flow for fluid-hair interactions. *Computers & Graphics* 52 (2015), 33–42.
- A. McAdams, E. Sifakis, and J. Teran. 2010. A Parallel Multigrid Poisson Solver for Fluids Simulation on Large Grids. In *Symposium on Computer Animation*. 65–74.
- Michael B Nielsen and Ole Østerby. 2013. A two-continua approach to Eulerian simulation of water spray. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.
- Stanley Osher, Ronald Fedkiw, and K Piechor. 2004. Level set methods and dynamic implicit surfaces. *Appl. Mech. Rev.* 57, 3 (2004), B15–B15.
- Saket Patkar and Parag Chaudhuri. 2013. Wetting of porous solids. *IEEE transactions on visualization and computer graphics* 19, 9 (2013), 1592–1604.
- B. Ren, J. Huang, M. C Lin, and S-M Hu. 2018. Controllable Dendritic Crystal Simulation Using Orientation Field. In *Computer Graphics Forum*, Vol. 37. 485–495.
- Bo Ren, Chenfeng Li, Xiao Yan, Ming C Lin, Javier Bonet, and Shi-Min Hu. 2014. Multiple-fluid SPH simulation using a mixture model. *ACM TOG* 33, 5 (2014), 1–11.
- Witawat Rungtirananon, Yoshihiro Kanamori, and Tomoyuki Nishita. 2012. Wetting effects in hair simulation. In *Computer Graphics Forum*, Vol. 31. 1993–2002.

- W. Rungtirananon, Z. Szego, Y. Kanamori, and T. Nishita. 2008. Real-time animation of sand-water interaction. In *Computer Graphics Forum*, Vol. 27. 1887–1893.
- A. Sellitto, VA Cimmelli, and D Jou. 2020. Nonlinear propagation of coupled first-and second-sound waves in thermoelastic solids. *Journal of Elasticity* 138, 1 (2020), 93–109.
- Rajsekhar Setaluri, Mridul Aanjaneya, Sean Bauer, and Eftychios Sifakis. 2014. SPGrid: A sparse paged grid structure applied to adaptive smoke simulation. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–12.
- Jos Stam. 1999. Stable Fluids. In *Proc. of ACM SIGGRAPH (SIGGRAPH '99)*. 121–128.
- Johan Stefan. 1891. Über die Theorie der Eisbildung, insbesondere über die Eisbildung im Polarmeere. *Annalen der Physik* 278, 2 (1891), 269–286.
- A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. 2013. A material point method for snow simulation. *ACM Transactions on Graphics* 32, 4 (2013), 1–10.
- A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for phase-change and varied materials. *ACM TOG* 33, 4 (2014), 1–11.
- Dan Stora, Pierre-Olivier Agliati, Marie-Paule Cani, Fabrice Neyret, and Jean-Dominique Gascuel. 1999. Animating lava flows.
- Andre Pradhana Tampubolon, Theodore Gast, Gergely Klár, Chuyuan Fu, Joseph Teran, Chenfanfu Jiang, and Ken Museth. 2017. Multi-species simulation of porous sand and water mixtures. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–11.
- U. Trottenberg, C. W. Oosterlee, and A. Schuller. 2001. *Multigrid*. Academic Press.
- Da Yu Tzou. 2014. *Macro-to microscale heat transfer: the lagging behavior*. John Wiley & Sons.
- Kiwon Um, Tae-Yong Kim, Youngdon Kwon, and JungHyun Han. 2013. Porous deformable shell simulation with surface water flow and saturation. *Computer Animation and Virtual Worlds* 24, 3–4 (2013), 247–254.
- V. R Voller and CR Swaminathan. 1991. ERAL Source-based method for solidification phase change. *Numerical Heat Transfer, Part B Fundamentals* 19, 2 (1991), 175–189.
- Joshuah Wolper, Yu Fang, Minchen Li, Jiecong Lu, Ming Gao, and Chenfanfu Jiang. 2019. CD-MPM: continuum damage material point methods for dynamic fracture animation. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–15.
- Joel Wrethorn, Rickard Armiento, and Ken Museth. 2017. Animation of crack propagation by means of an extended multi-body solver for the material point method. *Computers & Graphics* 69 (2017), 131–139.
- Tao Xue, Xiaobing Zhang, and Kumar K Tamma. 2018. Generalized heat conduction model in moving media emanating from Boltzmann Transport Equation. *International Journal of Heat and Mass Transfer* 119 (2018), 148–151.
- X. Yan, Y-T Jiang, C-F Li, R. R Martin, and S-M Hu. 2016. Multiphase SPH simulation for interactive fluids and solids. *ACM TOG* 35, 4 (2016), 1–11.
- Xiao Yan, C-F Li, X-S Chen, and S-M Hu. 2018. MPM simulation of interacting fluids and solids. In *Computer Graphics Forum*, Vol. 37. 183–193.
- Tao Yang, Jian Chang, Ming C Lin, Ralph R Martin, Jian J Zhang, and Shi-Min Hu. 2017. A unified particle system framework for multi-phase, multi-material visual simulations. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 1–13.
- T. Yang, J. Chang, B. Ren, M. C Lin, J. J Zhang, and S-M Hu. 2015. Fast multiple-fluid simulation using Helmholtz free energy. *ACM TOG* 34, 6 (2015), 1–11.
- Yonghao Yue, Breannan Smith, Peter Yichen Chen, Maytee Chantharayukhonthorn, Ken Kamrin, and Eitan Grinspun. 2018. Hybrid Grains: adaptive coupling of discrete and continuum simulations of granular media. *ACM TOG* 37, 6 (2018), 1–19.

A APPENDIX: USING A SINGLE-FIELD C-F MODEL TO DERIVE A CONVECTION-DIFFUSION MODEL

The single-field C-F diffusion equation is given by:

$$\tau \frac{\partial^2 \phi}{\partial t^2} + \tau \frac{\partial \phi}{\partial t} = k \frac{\partial^2 \phi}{\partial x^2} + \tau F_T k \frac{\partial}{\partial t} \left[\frac{\partial^2 \phi}{\partial x^2} \right] + W, \quad \text{for } F_T < 1$$

$$\frac{\partial \phi}{\partial t} = k \frac{\partial^2 \phi}{\partial x^2} + W, \quad \text{for } F_T = 1$$

Taking the case with $F_T < 1$ in incompressible flow as an example and replacing the partial time derivative ($\partial/\partial t$) in the diffusion equation by the material derivative ($D\phi/Dt = \partial\phi/\partial t + \mathbf{u} \cdot \nabla \phi$) yields:

$$\frac{\partial \phi}{\partial t} \rightarrow \left(\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi \right); \quad \tau \frac{\partial^2 \phi}{\partial t^2} \rightarrow \tau \frac{\partial}{\partial t} \left(\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi \right)$$

and

$$\tau \frac{\partial^2 \phi}{\partial t^2} + \tau \frac{\partial(\mathbf{u} \cdot \nabla \phi)}{\partial t} + \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = k \nabla \cdot \nabla \phi + \tau F_T k \frac{\partial}{\partial t} \frac{\partial^2 \phi}{\partial x^2} + \tau F_T k \mathbf{u} \cdot \nabla \frac{\partial^2 \phi}{\partial x^2}$$

where W is omitted for simplicity of exposition. Assuming $\mathbf{u} = V$ as a constant in the one-dimensional case gives:

$$\tau \frac{\partial^2 \phi}{\partial t^2} + \tau V \frac{\partial^2 \phi}{\partial x \partial t} + \frac{\partial \phi}{\partial t} + V \frac{\partial \phi}{\partial x} - k \frac{\partial^2 \phi}{\partial x^2} - \tau F_T k \frac{\partial}{\partial t} \frac{\partial^3 \phi}{\partial x^2} - V \tau F_T k \frac{\partial^3 \phi}{\partial x^3} = 0$$

with the following expression for the wave speed:

$$c = \frac{1}{2} \left[V^2 \pm \sqrt{V^2 - 4(1 - F_T) \frac{k}{\tau}} \right]$$

It follows from above that the diffusive speed c depends on the background material velocity V , assuming different values in different moving media. However, the correct diffusive speed is $c = \sqrt{(1 - F_T)k/\tau}$ (see Section 5). **In other words, using the classical single-field formulation yields a nonphysical wave speed.**

B APPENDIX: DERIVATION OF JACOBIAN MATRICES A_i

Our convection-diffusion formulation (9) using the generalized C-F diffusion model considers the following variables in 3D:

$$\mathbf{u} = [u^1 \quad u^2 \quad u^3]^T \quad \mathbf{q}_F = [q_F^1 \quad q_F^2 \quad q_F^3]^T \quad \mathbf{q}_C = [q_C^1 \quad q_C^2 \quad q_C^3]^T$$

$$\mathbf{U} = [\phi \quad 0 \quad 0 \quad 0 \quad \tau q_C^1 \quad \tau q_C^2 \quad \tau q_C^3]^T$$

$$\mathbf{W} = [W \quad -q_F^1 \quad -q_F^2 \quad -q_F^3 \quad -q_C^1 \quad -q_C^2 \quad -q_C^3]^T$$

$$\mathbf{G} = \begin{bmatrix} u^1 \phi + q_F^1 + q_C^1 & u^2 \phi + q_F^2 + q_C^2 & u^3 \phi + q_F^3 + q_C^3 \\ F_T k \phi & 0 & 0 \\ 0 & F_T k \phi & 0 \\ 0 & 0 & F_T k \phi \\ (1 - F_T)k \phi + \tau q_C^1 u^1 & \tau q_C^1 u^2 & \tau q_C^1 u^3 \\ \tau q_C^2 u^1 & (1 - F_T)k \phi + \tau q_C^2 u^2 & \tau q_C^2 u^3 \\ \tau q_C^3 u^1 & \tau q_C^3 u^2 & (1 - F_T)k \phi + \tau q_C^3 u^3 \end{bmatrix}$$

where the superscripts 1, 2, 3 represent different Cartesian axes. Therefore, we can compute the Jacobian matrices A_i as follows:

$$A_1 = \frac{\partial G_1}{\partial \mathbf{U}} = \begin{bmatrix} u^1 & 0 & 0 & 0 & 1/\tau & 0 & 0 \\ F_T k & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ (1 - F_T)k & 0 & 0 & 0 & u^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & u^1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u^1 \end{bmatrix}$$

$$A_2 = \frac{\partial G_2}{\partial \mathbf{U}} = \begin{bmatrix} u^2 & 0 & 0 & 0 & 0 & 1/\tau & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ F_T k & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & u^2 & 0 & 0 \\ (1 - F_T)k & 0 & 0 & 0 & 0 & u^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u^2 \end{bmatrix}$$

$$A_3 = \frac{\partial G_3}{\partial \mathbf{U}} = \begin{bmatrix} u^3 & 0 & 0 & 0 & 0 & 0 & 1/\tau \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ F_T k & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & u^3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & u^3 & 0 \\ (1 - F_T)k & 0 & 0 & 0 & 0 & 0 & u^3 \end{bmatrix}$$