Journal of Scientific Computing (2019) 81:1860–1881
https://doi.org/10.1007/s10915-019-01067-1

# An Efficient Boundary Integral Scheme for the Threshold Dynamics Method II: Applications to Wetting Dynamics

**Dong Wang[1] · Shidong Jiang[2] · Xiao-Ping Wang[3]**

Received: 11 January 2019 / Revised: 3 October 2019 / Accepted: 5 October 2019 /
Published online: 19 October 2019


**Abstract**
In this paper, we extend the boundary integral scheme for the threshold dynamics method to treat the case where the material interface is nonsmooth and may undergo topological changes. The scheme is then applied to study the wetting dynamics in both two and three dimensions. Numerical experiments show that the scheme is more efficient as compared with the existing method using uniform grids, making accurate simulation of wetting dynamics on a chemically patterned solid surface in three dimensions within practical reach.

**Keywords** Threshold dynamics method · Nonuniform FFT · Heat equation · Wetting

## 1 Introduction

Wetting and spreading are of critical importance for many applications such as microfluidics, inkjet printing, surface engineering, and oil recovery [2,3]. The way in which a liquid drop spreads on a solid surface is governed by the surface and interfacial interactions, which usually occur at a small scale. The most important quantity in wetting is the contact angle between the liquid surface and the solid surface [14]. When the solid surface is homogeneous, the contact angle for a static drop is given by Young's equation:

$$\cos\theta_Y = \frac{\gamma_{SV} - \gamma_{SL}}{\gamma_{LV}}, \tag{1}$$

✉ Dong Wang
dwang@math.utah.edu

Shidong Jiang
shidong.jiang@njit.edu

Xiao-Ping Wang
mawang@ust.hk

[1] Department of Mathematics, University of Utah, Salt Lake City, UT 84112, USA

[2] Department of Mathematical Sciences, New Jersey Institute of Technology, Newark, NJ 07102, USA

[3] Department of Mathematics, The Hong Kong University of Science and Technology, Kowloon, Hong Kong, China

where $\gamma_{SL}$, $\gamma_{SV}$ and $\gamma_{LV}$ are the solid–liquid, solid–vapor and liquid–vapor surface energy densities, respectively. $\theta_Y$ is the so-called Young's angle [33]. Mathematically, Young's equation (1) can be derived by minimizing the total energy in the solid–liquid–vapor system. If we ignore gravity and the effect of the fluid flow, the total energy in the system can be written as

$$\mathcal{E} = \gamma_{LV}|\Sigma_{LV}| + \gamma_{SL}|\Sigma_{SL}| + \gamma_{SV}|\Sigma_{SV}|, \tag{2}$$

where $\Sigma_{LV}$, $\Sigma_{SL}$ and $\Sigma_{SV}$ are the liquid–vapor, solid–liquid and solid–vapor interfaces respectively and $|\cdot|$ denotes the area of the interface. When the solid surface $\Gamma$ is a homogeneous planar surface, provided that the volume of the liquid drop is fixed, the unique minimizer of the total energy is a domain with a spherical surface in $\Omega$ with constant curvature, and the contact angle between the spherical surface and the solid surface $\Gamma$ is Young's angle $\theta_Y$ [29].

The study of wetting dynamics has attracted much attention in the physics and applied mathematics communities [1,10,12,21,32]. Numerical simulation of wetting on rough surfaces is challenging. One must track the interface motion accurately, and deal with complicated boundary shapes and boundary conditions. There are many different numerical methods for solving interface and contact line problems, including the front-tracking method [18,30], the front-capturing method using the level-set function [34], and the phase-field methods [4,5]. Recently, Xu et al. [28,31] developed an efficient threshold dynamics method for solid wetting problems. The method is based on the minimization of the weighted surface area functional over an extended domain that includes the solid domain as an additional phase. The method is insensitive to the inhomogeneity or roughness of the solid boundary. The heat equation is solved in a rectangular domain with a uniform grid using convolution of the heat kernel with the initial condition. The convolution is evaluated using the fast Fourier transform (FFT) at $O(N \log N)$ cost per time step with $N$ being the total number of uniform grid points in the entire computational domain. However, simulations must choose a spatial mesh size that is of the same order as the time step size, resolve the thin layer around the contact points, and capture the interface accurately. These requirements make $N$ very large and the simulation very expensive, especially for three dimensional problems. The threshold dynamics method has subsequently been extended to deal with many other applications including image processing [24,27], topology optimization [6], and harmonic target-valued maps [20,25,26].

In this paper, we extend the boundary integral scheme for the threshold dynamics method developed in [17] for smooth material interfaces to *nonsmooth* material interfaces. The scheme discretizes the physical space only in a neighborhood of the interfaces and applies non-uniform fast Fourier transform (NUFFT) [8,9,15] to further accelerate the calculation. Unlike many grid-based methods where the spatial mesh size is required to be of the same order as the time step size, the numerical experiments showed that the spatial mesh size can be chosen based on the accuracy consideration and was more or less independent of the time step size for the whole simulation. The combination of these techniques has greatly reduced the computational cost for the threshold dynamics simulation. We then apply the scheme to study the wetting dynamics in both two and three dimensions, where the solid surface might be chemically patterned. The numerical results demonstrate that the scheme offers significant speedup for the simulation of wetting dynamics. Indeed, as compared with the method presented in [31], the scheme requires much less memory and is about 100 times faster for the simulation of wetting dynamics on a chemically patterned solid surface in three dimensions.

The rest of the paper is organized as follows. In Sect. 2, we first introduce the threshold dynamics method for wetting dynamics on a single solid surface and then extend it to the wetting dynamics on a chemically patterned surface. In Sect. 3, we introduce the boundary integral scheme for the pure heat diffusion equation with the initial condition being the characteristic function of some bounded domain with a piecewise smooth boundary. The thresholding scheme is introduced in Sect. 4. Various numerical experiments are presented in Sect. 5. Conclusions and future directions are given in Sect. 6.

## 2 Threshold Dynamics Method for Wetting Dynamics

In this section, we briefly introduce the threshold dynamics method for wetting dynamics on solid surfaces. In Sect. 2.1, we introduce the threshold dynamics method for wetting dynamics on a chemically homogeneous surface. In Sect. 2.2, we extend the threshold dynamics method to deal with wetting dynamics on a chemically patterned surface.

### 2.1 Threshold Dynamics Method for Wetting Dynamics on a Chemically Homogeneous Surface

We first consider a wetting problem in a domain $\Omega \in \mathbb{R}^d$, $d = 2, 3$ (see the left graph in Fig. 1). The solid surface $\Gamma$ (smooth $a.e.$) is part of the domain boundary $\partial\Omega$. Denote the liquid domain by $D_1 \subset \Omega$. We assume that $\partial D_1 \cap \partial\Omega \subset \Gamma$. The volume of the liquid drop is fixed such that $|D_1| = V_0$. We denote by $\Sigma_{LV} = \partial D_1 \cap \Omega$, $\Sigma_{SL} = \partial D_1 \cap \Gamma$ and $\Sigma_{SV} = \Gamma \backslash \partial D_1$ the liquid–vapor, solid–liquid and solid–vapor interfaces respectively.

Xu et al. [31] extend the liquid–vapor domain to a larger domain $\tilde{\Omega}$ containing the solid phase denoted by $D_3$ (see the right graph in Fig. 1). Analogous to that in [11], the energy (2) can be approximated by

$$\begin{aligned}\mathcal{E}^{\delta t}(\chi_{D_1}, \chi_{D_2}) = &\frac{\gamma_{LV}}{\sqrt{\delta t}} \int_{\tilde{\Omega}} \chi_{D_1} G_{\delta t} * \chi_{D_2} \mathrm{d}\mathbf{x} + \frac{\gamma_{SL}}{\sqrt{\delta t}} \int_{\tilde{\Omega}} \chi_{D_1} G_{\delta t} * \chi_{D_3} \mathrm{d}\mathbf{x} \\ &+ \frac{\gamma_{SV}}{\sqrt{\delta t}} \int_{\tilde{\Omega}} \chi_{D_2} G_{\delta t} * \chi_{D_3} \mathrm{d}\mathbf{x}, \end{aligned} \tag{3}$$

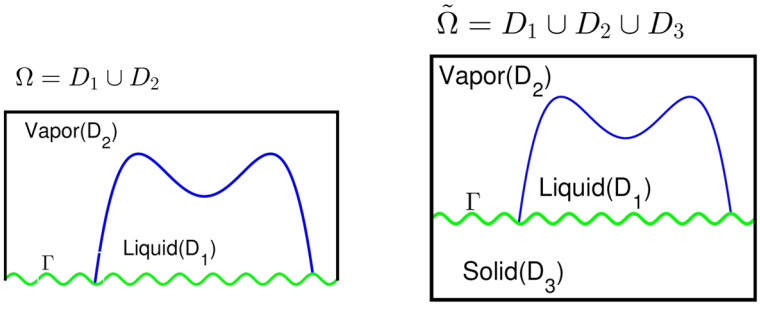


**Fig. 1** Left: the liquid domain and vapor domain in the wetting system. Right: the liquid domain, vapor domain and solid domain in the wetting system

where

$$G_{\delta t}(\mathbf{x}) := \frac{1}{(4\pi\delta t)^{d/2}} \exp\left(-\frac{\|\mathbf{x}\|^2}{4\delta t}\right)$$

is the heat kernel at time $\delta t$ and $\chi_{D_i}$ denotes the indicator function of the domain $D_i$, that is,

$$\chi_{D_i}(x) = \begin{cases} 1 & \text{if } x \in D_i, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Since $D_3$ is always fixed, denote $u_1 = \chi_{D_1}$ and $u_2 = \chi_{D_2}$ and define an admissible set

$$\mathcal{B} = \Big\{(u_1, u_2) \in BV(\Omega) \mid u_i(x) = 0, 1, \text{ and } u_1(x) + u_2(x) = 1, \ a.e.\ x \in \Omega, \int_\Omega u_1 \mathrm{d}\mathbf{x} = V_0\Big\} \tag{5}$$

where $BV$ denotes the space of functions of bounded variation. The wetting problem can now be approximated by

$$\min_{(u_1,u_2)\in\mathcal{B}} \mathcal{E}^{\delta t}(u_1, u_2). \tag{6}$$

Based on the relaxation and linearization procedure developed in [11], Xu et al. [31] first relax the problem to an equivalent minimization problem in a convex admissible set:

$$\min_{(u_1,u_2)\in\mathcal{K}} \mathcal{E}^{\delta t}(u_1, u_2). \tag{7}$$

where $\mathcal{K}$ is the convex hull of the admissible set $\mathcal{B}$:

$$\mathcal{K} = \Big\{(u_1, u_2) \in BV(\Omega) \mid 0 \le u_i \le 1, u_1(x) + u_2(x) = 1, \ a.e.\ x \in \Omega, \int_\Omega u_1 \mathrm{d}\mathbf{x} = V_0\Big\}. \tag{8}$$

Using an iterative method, when the $k^{th}$ iteration $(u_1^k, u_2^k)$ is computed, the approximate energy (3) is linearized around $(u_1^k, u_2^k)$ by

$$\mathcal{E}^{\delta t}(u_1, u_2) \approx \mathcal{E}^{\delta t}(u_1^k, u_2^k) + \hat{\mathcal{L}}(u_1 - u_1^k, u_2 - u_2^k, u_1^k, u_2^k).$$

with

$$\hat{\mathcal{L}}(u_1, u_2, u_1^k, u_2^k) = \frac{1}{\sqrt{\delta t}} \left(\int_{\tilde{\Omega}} u_1 G_{\delta t} * (\gamma_{LV} u_2^k + \gamma_{SL}\chi_{D_3}) + u_2 G_{\delta t} * (\gamma_{LV} u_1^k + \gamma_{SV}\chi_{D_3}) \mathrm{d}\mathbf{x}\right). \tag{9}$$

To find the approximate $(u_1^{k+1}, u_2^{k+1})$, one needs only to solve the following minimization problem of the linear functional on a convex admissible set:

$$\min_{(u_1,u_2)\in\mathcal{K}} \hat{\mathcal{L}}(u_1 - u_1^k, u_2 - u_2^k, u_1^k, u_2^k). \tag{10}$$

This is equivalent to solving

$$\min_{(u_1,u_2)\in\mathcal{K}} \left(\int_{\tilde{\Omega}} u_1\phi_1 + u_2\phi_2 \mathrm{d}\mathbf{x}\right), \tag{11}$$

where $\phi_1 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} u_2^k + \gamma_{SL} \chi_{D_3})$ and $\phi_2 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} u_1^k + \gamma_{SV} \chi_{D_3})$. This problem is reduced to a point-wise minimization problem, which can be efficiently solved with a thresholding step through comparing the values of $\phi_1$ and $\phi_2$, i.e.,

$$u_1^{k+1} = \begin{cases} 1 & \text{if } \phi_1 < \phi_2 + \delta, \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

and $u_2^{k+1} = \chi_\Omega - u_1^{k+1}$. Here $\delta$ is a parameter to be determined so that the volume is preserved, i.e., $\int u_1^{k+1} d\mathbf{x}$ is unchanged. More details will be given in Sect. 4. The method is summarized in Algorithm 1.

**Algorithm 1** Threshold dynamics method for solid wetting problems.

*Given initial $D_1^0, D_2^0 \subset \Omega$, such that $D_1^0 \cap D_2^0 = \emptyset$, $D_1^0 \cup D_2^0 = \Omega$ and $|D_1^0| = V_0$, set a tolerance parameter $\varepsilon > 0$.*

1: For given sets $(D_1^k, D_2^k)$, compute two functions

$$\phi_1 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} \chi_{D_2^k} + \gamma_{SL} \chi_{D_3}), \ \phi_2 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} \chi_{D_1^k} + \gamma_{SV} \chi_{D_3}). \tag{13}$$

2: Find $\delta$ such that the set

$$\tilde{D}_1^\delta = \{x \in \Omega \,|\, \phi_1 < \phi_2 + \delta\} \tag{14}$$

satisfies $|\tilde{D}_1^\delta| = V_0$. Denote $D_1^{k+1} = \tilde{D}_1^\delta$ and $D_2^{k+1} = \Omega \backslash D_1^{k+1}$.

3: If $|D_1^k - D_1^{k+1}| \le \varepsilon$, stop; otherwise, go back to Step 1.

Note that step 1 in Algorithm 1 requires calculating the weighted sum of the convolution between $G_{\delta t}(\mathbf{x})$ and the characteristic functions of some bounded domains with piecewise smooth boundaries. This is equivalent to finding the solution to the pure initial value problem of the heat diffusion equation at $t = \delta t$, where the initial data are the characteristic functions of some domains with piecewise smooth boundaries. This will be discussed in Sect. 3. The thresholding step will be discussed in Sect. 4.

### 2.2 Threshold Dynamics Method for Wetting Dynamics on Chemically Patterned Surfaces

In this section, we will extend the threshold dynamics method in Sect. 2.1 to deal with wetting dynamics on a chemically patterned surface. A chemically patterned solid surface is a surface patterned with hydrophobic and hydrophilic materials (see, for example, Fig. 2).

To extend Algorithm 1 to wetting dynamics on a chemically patterned surface, we need to split the original solid domain into two domains $D_3$ and $D_4$ in Fig. 2, representing materials $\mathcal{A}$ and $\mathcal{B}$, respectively. Denote by $\gamma_{S_\mathcal{A} L}$, $\gamma_{S_\mathcal{A} V}$, $\gamma_{S_\mathcal{B} L}$ and $\gamma_{S_\mathcal{B} V}$ the solid$_\mathcal{A}$-liquid, solid$_\mathcal{A}$-vapor, solid$_\mathcal{B}$-liquid, and solid$_\mathcal{B}$-vapor surface energy densities, respectively. Then, the total interfacial energy (2) becomes

$$\mathcal{E} = \gamma_{LV} |\Sigma_{LV}| + \gamma_{S_\mathcal{B} L} |\Sigma_{S_\mathcal{B} L}| + \gamma_{S_\mathcal{B} V} |\Sigma_{S_\mathcal{B} V}| + \gamma_{S_\mathcal{A} L} |\Sigma_{S_\mathcal{A} L}| + \gamma_{S_\mathcal{A} V} |\Sigma_{S_\mathcal{A} V}|. \tag{15}$$

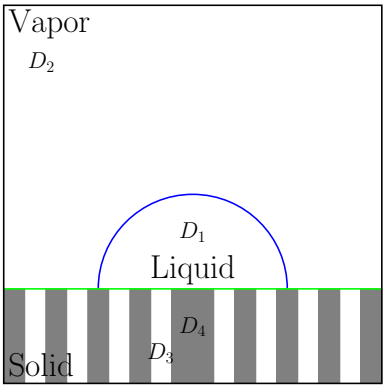


**Fig. 2** A liquid droplet on a chemically patterned solid surface

Similarly, when the $k$th iteration $(u_1^k, u_2^k)$ is computed, the $(k+1)$th iteration $(u_1^{k+1}, u_2^{k+1})$ can be obtained as the solution to the following minimization problem:

$$\min_{(u_1,u_2)\in\mathcal{K}} \left( \int_{\tilde{\Omega}} u_1\psi_1 + u_2\psi_2 \mathrm{d}\mathbf{x} \right), \tag{16}$$

where $\psi_1 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} u_2^k + \gamma_{S_{\mathcal{A}}L}\chi_{D_3} + \gamma_{S_{\mathcal{B}}L}\chi_{D_4})$ and $\psi_2 = \frac{1}{\sqrt{\delta t}} G_{\delta t} * (\gamma_{LV} u_1^k + \gamma_{S_{\mathcal{A}}V}\chi_{D_3} + \gamma_{S_{\mathcal{B}}V}\chi_{D_4})$. This problem can be solved in a point-wise manner and $(u_1^{k+1}, u_2^{k+1})$ can be redefined via the formulas

$$u_1^{k+1} = \begin{cases} 1, & \text{if } \psi_1 < \psi_2 + \delta, \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

and $u_2^{k+1} = \chi_\Omega - u_1^{k+1}$. Once again, $\delta$ is a parameter to be determined so as to preserve the volume $\int u_1^{k+1} \mathrm{d}\mathbf{x}$. The algorithm is almost identical to Algorithm 1, except that we need to change $\phi_1$ and $\phi_2$ to $\psi_1$ and $\psi_2$, respectively.

An important step in both algorithms is to solve the free space heat diffusion equation with the initial data being the weighted sum of the characteristic functions of different bounded domains. In the following, we briefly review and extend an efficient boundary integral scheme for the free space heat solver with the initial data being the characteristic function of a bounded domain. The original scheme is introduced in [17] for smooth boundaries, and we extend the scheme here to treat *nonsmooth* boundaries. Then $\phi_1$, $\phi_2$, $\psi_1$, and $\psi_2$ can be calculated as the weighted sum of these solutions.

## 3 An NUFFT-Based Heat Solver

Consider the following pure initial value problem of the heat equaiton:

$$\begin{aligned} u_t(\mathbf{x}, t) &= \Delta u(\mathbf{x}, t), \\ u(\mathbf{x}, 0) &= \chi_D(\mathbf{x}), \end{aligned} \tag{18}$$

where the initial data $\chi_D(\mathbf{x})$ is the characteristic function of some bounded domain $D \in \mathbb{R}^d$ $(d = 2, 3)$. In [17], we assume that the boundary $\partial D$ of $D$ is smooth. For wetting dynamics, the boundary is often piecewise smooth due to the presence of the (possibly chemically patterned) solid surface. Here, we first summarize the NUFFT based solver for (18) developed

in [17] for smooth boundaries. We then extend the scheme to nonsmooth boundaries. Note that $u(\mathbf{x}, \delta t)$ is given by the formula

$$u(\mathbf{x}, \delta t) = \int \cdots \int_{\mathbb{R}^d} G_d(\mathbf{x}, \mathbf{y}; \delta t)\chi_D(\mathbf{y})d\mathbf{y} = \int \cdots \int_D G_d(\mathbf{x}, \mathbf{y}; \delta t)d\mathbf{y}, \tag{19}$$

where the Green's function (or the fundamental solution) of the heat equation in $\mathbb{R}^d$ is given by the formula

$$G_d(\mathbf{x}, \mathbf{y}; t) = \frac{1}{(4\pi t)^{d/2}} e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{4t}}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{20}$$

It is well known that $G_d$ admits the following Fourier representation:

$$G_d(\mathbf{x}, \mathbf{y}; \delta t) = \frac{1}{(2\pi)^d} \int \cdots \int_{\mathbb{R}^d} e^{-\|\mathbf{k}\|^2 \delta t + i\mathbf{k}\cdot(\mathbf{x}-\mathbf{y})} d\mathbf{k}, \quad \mathbf{k} \in \mathbb{R}^d. \tag{21}$$

Substituting (21) into (19) and exchanging the order of integration, we obtain

$$u(\mathbf{x}, \delta t) = \frac{1}{(2\pi)^d} \int \cdots \int_{\mathbb{R}^d} e^{-\|\mathbf{k}\|^2 \delta t + i\mathbf{k}\cdot\mathbf{x}} f(\mathbf{k})d\mathbf{k}, \tag{22}$$

where $f(\mathbf{k})$ is given by the formula

$$f(\mathbf{k}) = \int \cdots \int_D e^{-i\mathbf{k}\cdot\mathbf{y}} d\mathbf{y}. \tag{23}$$

and $f(\mathbf{k})$ is a $C^\infty$ function since $D$ is a bounded domain in $\mathbb{R}^d$. One may also show that $f(\mathbf{k})$ tends to 0 as $\|\mathbf{k}\| \to \infty$ (see, for example, [22]).

At first glance, Eq. (22) seems to be much more expensive than (19) since it involves integrals in both physical and Fourier spaces. However, the following three observations can be used to greatly reduce the computational cost of (22). First, the area/volume integral in (23) can be converted to a line/surface integral using either Green's theorem or the divergence theorem. Second, the heat kernel $G_d$ admit an efficient spectral Fourier approximation due to the exponential decay of the high frequency modes in its Fourier representation (see, for example, [16,17]). Third, NUFFT can be used to speed up the calculation of the discrete sums.

In [17], we have assumed that the material interface is smooth and thus used a spectrally accurate global discretization. For wetting dynamics, due to the existence of possibly chemically patterned solid surface and the contact angle, such assumptions are no longer held. Instead, we have to consider the case where the boundary is only piecewise smooth.

For 2D problems, the double integral in (23) is converted into a line integral using Green's theorem as follows:

$$f(\mathbf{k}) = \iint_D e^{-i\mathbf{k}\cdot\mathbf{y}} d\mathbf{y} = \begin{cases} \frac{i}{k_1} \int_{\partial D} e^{-i\mathbf{k}\cdot\mathbf{y}} dy_2, & k_1 \neq 0, \\ \int_{\partial D} y_1 e^{-ik_2 y_2} dy_2, & k_1 = 0 \end{cases} \tag{24}$$

with $\mathbf{k} = (k_1, k_2)$. Because the boundary curve $\partial D$ is only *piecewise* smooth, to evaluate the boundary integral in (24), we divide the boundary into, say, $K_S$ chunks with corners belonging to the set of end points of chunks. We then discretize the parameter space for each chunk using a $p$th order Gauss–Chebyshev rule (we set $p = 16$ in our implementation). The discretization of the integrals in (24) is of the $p$th order and the overall heat solver will also be of the $p$th order. Assume we use the $p$th order scaled and shifted Chebyshev nodes

$(s_{k_s0}, s_{k_s1}, s_{k_s2}, \ldots, s_{k_sp})$ with weights $(w_{k_s0}, w_{k_s1}, w_{k_s2}, \ldots, w_{k_sp})$ to discretize the $k_s$th chunk for $k_s = 1, 2, \ldots, K_S$. Then, we have

$$f(\mathbf{k}) \approx \begin{cases} \dfrac{i}{k_1} \displaystyle\sum_{k_s=1}^{K_S} \sum_{j=0}^{p} e^{-i\mathbf{k}\cdot\mathbf{y}(s_{k_sj})} y_2'(s_{k_sj}) w_{k_sj}, & k_1 \neq 0, \\ \displaystyle\sum_{k_s=1}^{K_S} \sum_{j=0}^{p} y_1(s_j) e^{-ik_2 y_2(s_{k_sj})} y_2'(s_{k_sj}) w_{k_sj}, & k_1 = 0. \end{cases} \tag{25}$$

Furthermore, the area bounded by $\partial D$ can be calculated by

$$A = \iint_D d\mathbf{y} = \int_{\partial D} y_1 dy_2 \approx \sum_{k_s=1}^{K_S} \sum_{j=0}^{p} y_1(s_{k_sj}) y_2'(s_{k_sj}) w_{k_sj}. \tag{26}$$

For 3D problems, the volume integral in (23) is converted into a surface integral using the divergence theorem

$$f(\mathbf{k}) = \iiint_D e^{-i\mathbf{k}\cdot\mathbf{y}} d\mathbf{y} = \begin{cases} \frac{i}{k_1} \displaystyle\iint_{\partial D} e^{-i\mathbf{k}\cdot\mathbf{y}} (\hat{i}\cdot\mathbf{n}_y) ds_y, & k_1 \neq 0, \\ \displaystyle\iint_{\partial D} y_1 e^{-ik_2y_2 - ik_3y_3} (\hat{i}\cdot\mathbf{n}_y) ds_y, & k_1 = 0, \end{cases} \tag{27}$$

where $\hat{i}$ is the unit vector along the $x$-axis and $\mathbf{n}_y$ is the unit normal vector at $\mathbf{y}$. For *piecewise* smooth $\partial D$, we divide it into $K_S$ patches. We then parametrize each patch via $[u, v] \in [0, \pi] \times [0, \pi]$, where $u$ is the polar angle and $v$ is the azimuthal angle. We now discretize $u$ using $N_1$ scaled and shifted Chebyshev nodes $(u_{k_s1}, u_{k_s2}, \ldots, u_{k_sN_1})$, and $v$ using $N_2$ scaled and shifted Chebyshev nodes $(v_{k_s1}, v_{k_s2}, \ldots, v_{k_sN_2})$ for $k_s = 1, 2, \ldots, K_S$. Hence, the integrals in (27) are approximated by

$$f(\mathbf{k}) \approx \begin{cases} \dfrac{i}{k_1} \displaystyle\sum_{k_s=1}^{K_S} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} e^{-i\mathbf{k}\cdot\mathbf{y}_{k_sij}} n_1(u_{k_si}, v_{k_sj}) J_{k_sij} w_{k_sij}, & k_1 \neq 0, \\ \displaystyle\sum_{k_s=1}^{K_S} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} y_1(u_{k_si}, v_{k_sj}) e^{-i\mathbf{k}\cdot\mathbf{y}_{k_sij}} n_1(u_{k_si}, v_{k_sj}) J_{k_sij} w_{k_sij}, & k_1 = 0, \end{cases} \tag{28}$$

where $n_1(u_{k_si}, v_{k_sj})$ is the $x$-component of the unit outward normal vector, $J_{k_sij} = |\mathbf{y}_u \times \mathbf{y}_v|$ is the Jacobian at the point $(u_{k_si}, v_{k_sj})$, and $w_{k_sij}$ is the corresponding quadrature weight. Furthermore, the volume bounded by $\partial D$ can be computed as follows:

$$\begin{aligned} V &= \iiint_D d\mathbf{y} = \iint_{\partial D} x(\hat{i}\cdot\mathbf{n}_y) ds_y \\ &\approx \sum_{k_s=1}^{K_S} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} x(u_{k_si}, v_{k_sj}) n_1(u_{k_si}, v_{k_sj}) J_{k_sij} w_{k_sij}. \end{aligned} \tag{29}$$

After $f(\mathbf{k})$ has been computed, the solution $u(\mathbf{x}, \delta t)$ in (22) can be evaluated by approximating the Fourier integral via a spectrally accurate truncated trapezoidal rule. We have

$$u(\mathbf{x}, \delta t) \approx \frac{h^d}{(2\pi)^d} \sum_{m_1=-M}^{M-1} \cdots \sum_{m_d=-M}^{M-1} e^{-\|\mathbf{m}\|^2 h^2 \delta t + ih\mathbf{m}\cdot\mathbf{x}} f(h\mathbf{m}). \tag{30}$$

We refer the readers to Theorem 1 in [17] for the choices of $h$ and $M$ in (30). It is clear that we can apply type-1 NUFFT to evaluate $f(h\mathbf{m})$ defined in (25) or (28), and then apply type-2 NUFFT to evaluate $u(\mathbf{x}, \delta t)$ in (30) efficiently (see, for example, [15] about the definitions of different types of NUFFTs).

We summarize the whole procedure in Algorithm 2. The total cost of Algorithm 2 is $O(N_S + N_T + N_F \log N_F)$, where $N_S$, $N_T$ are the total number of non-equispaced source points on the boundary and the total number of target points near the boundary, respectively, and $N_F = (2M)^d$ is the total number of equispaced points in the Fourier space.

**Algorithm 2** NUFFT-based solver for the initial value problem (18).

*Given the prescribed accuracy $\epsilon$, the time step size $\delta t$, and the boundary $\partial D$, compute $u(\mathbf{x}, \delta t)$ defined in (19) on a set of prescribed target points $\mathbf{x}_i$, $i = 1, \ldots, N_T$.*

1: Divide the boundary $\partial D$ into $K_S$ subsets and discretize each subset via $N_S$ points in the parameter space (scaled and shifted Chebyshev nodes for curves in 2D; scaled and shifted Chebyshev nodes along both the polar direction and the azimuthal direction for surfaces in 3D), and compute the source locations $\mathbf{y}_{k_s j}$ ($k_s = 1, \ldots, K_S$; $j = 0, \ldots p$ in 2D and $j = 1, \ldots N_1 N_2$ in 3D ) via the given parametrization of $\partial D$, and the associated weights $w_{k_s j}$.
2: Compute the derivatives $\mathbf{y}'_j$ or the Jacobians.
3: Compute the mesh spacing $h$ and $M$ for the Fourier spectral approximation of the heat kernel according to Theorem 1 in [17].
4: Use type-1 NUFFT to evaluate $f(h\mathbf{m})$ defined in (25) or (28) for $m_i = -M, \ldots, M-1$ ($i = 1, \ldots, d$).
5: Use type-2 NUFFT to evaluate $u(\mathbf{x}_j, \delta t)$ defined in (30) for $j = 1, \ldots, N_T$.

## 4 The Thresholding Step

We now discuss how thresholding is performed in our algorithm. As mentioned in the preceding section, we use a carefully chosen set of points to represent the interface in both two and three dimensions from which all other geometric quantities such as the tangential derivatives, unit normal vectors, and area elements can be computed efficiently. Hence, we only need to keep track of this set of points in the threshold dynamics.

We note that in Step 2 of Algorithm 1, $D_1$ at the next iteration is determined by comparing the values of $\phi_1$ and $\phi_2$. It can be simplified via computing

$$\begin{aligned}
\gamma_{LV}\tilde{\phi} = \phi_1 - \phi_2 &= \frac{1}{\sqrt{\delta t}} G_{\delta t} * \left(\gamma_{LV}\chi_{D_2^k} + \gamma_{SL}\chi_{D_3} - \gamma_{LV}\chi_{D_1^k} - \gamma_{SV}\chi_{D_3}\right) \\
&= \frac{1}{\sqrt{\delta t}} G_{\delta t} * \left(\gamma_{LV}\chi_{D_2^k} - \gamma_{LV}\chi_{D_1^k} + (\gamma_{SL} - \gamma_{SV})\chi_{D_3}\right) \\
&= \frac{\gamma_{LV}}{\sqrt{\delta t}} G_{\delta t} * \left(\chi_{D_2^k} - \chi_{D_1^k} + \frac{\gamma_{SL} - \gamma_{SV}}{\gamma_{LV}}\chi_{D_3}\right) \\
&= \gamma_{LV}\frac{1}{\sqrt{\delta t}} G_{\delta t} * \left(\chi_{D_2^k} - \chi_{D_1^k} + \cos(\theta_Y)\chi_{D_3}\right)
\end{aligned}$$

where the last equality follows from (1). Then, Step 2 is equivalent to find $x$ such that $\tilde{\phi}(x) < \tilde{\delta}$ where $\tilde{\delta} = \frac{\delta}{\gamma_{LV}}$. For the patterned surface, we can define $\tilde{\psi} = \frac{\psi_1 - \psi_2}{\gamma_{LV}}$ in the same way. Hence, in numerical examples, we only need to set the equilibrium contact angles $\theta_Y$ to determine the dynamics instead of considering the specific choices of surface tensions.

Thresholding begins by finding the level set of a given $\tilde{\phi}$ (or $\tilde{\psi}$) calculated by (13) based on the solution to the initial value problem (18) with some initial sets $D_1$, $D_2$, and $D_3$ (or $D_3$, $D_4$). Note that in Algorithm 1, each iteration starts with an indicator function of a set and ends with the indicator function of a new set. In other words, we start with a set of points $\mathbf{y}_j$ $(j = 1, \ldots, T_S)$ to represent the boundary of the initial domain and end up with another set of points to represent the boundary of the new domain at the next time step.

Before we introduce the algorithm for finding the level set of $\tilde{\phi}$ (or $\tilde{\psi}$), we first introduce the algorithm to allocate the target points in wetting problems. When the boundary is smooth, it is natural to allocate the target points along the normal direction of each source point. And the points on the new boundary are obtained through a rootfinding algorithm along the normal lines, with the target points serving as the interpolation nodes for function evaluation. However, when the boundary is nonsmooth, say, having corners, these normal lines from different source points will intersect with each other even for small diffusion time steps. The subsequent line search algorithm will then change the order of discrete points on the new boundary, leading to instability of the scheme.

An implicit constraint in the threshold dynamics method for wetting dynamics is that source points on the solid surface should always be located on the solid surface. In this case, if we continue to allocate target points along the normal direction for each source point as in [17], the original source points on the solid surface will move away from the solid surface (below or above). Hence, to find the proper direction for allocating target points, we need to "interpolate" between the normal direction at each source point and the tangential direction of the solid surface at the contact points (in the 3D case, the tangential direction is considered to be the direction tangential to the solid surface and normal to the contact line). Generally, at each source point, when it is close to the solid surface, we expect to allocate some of the target points around it along or parallel to the solid surface. When the source point is away from the solid surface, some of the target points along the normal direction of the surface should be allocated. Specifically, we introduce a weight function $W_s : [0, \infty) \to [0, 1]$ (e.g., $W_s(x) = \min(x, 1)$). The direction in which the target points should be allocated is then determined by

$$W_s(d(\mathbf{x}))\mathbf{n}(\mathbf{x}) + (1 - W_s(d(\mathbf{x})))\mathbf{t}(\mathbf{x}),$$

where $d(\mathbf{x})$ is the distance from the source point $\mathbf{x}$ to the solid surface, $\mathbf{n}(\mathbf{x})$ is the normal vector at $\mathbf{x}$, and $\mathbf{t}(\mathbf{x})$ is the tangential direction at $\mathbf{x}$. In Fig. 3, we plot the allocated target points where $W_s$ is a piecewise constant function.

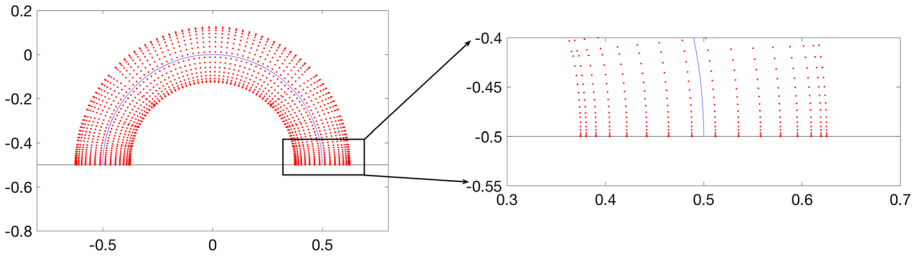


**Fig. 3** Target points allocated for a wetting problem

The algorithm is summarized in Algorithm 3.

**Algorithm 3** Allocating target points in single-droplet wetting problems.

*Given the time step size $\delta t$, the tangential direction $\mathbf{t}_j$ at the source point on the solid surface, and a set of source points $\mathbf{y}_j$ ($j = 1, \ldots, N_s$) describing the interface at the current time, return the target points.*

1: Compute the derivatives $\mathbf{y}'_j$ and the unit normal vector $\mathbf{n}_j$ for $j = 1, \ldots, N_s$.
2: Compute a length $L$ that is proportional to $\sqrt{\delta t}$, for example, $L = 5\sqrt{\delta t}$.
3: Compute $p$ Gauss–Legendre nodes on the standard interval $[-1, 1]$.
4: Select a weight function $W_s$ and compute the distance between each source point and the solid surface to form $d_j$ for $j = 1, \ldots, N_s$.
5: For each point $\mathbf{y}_j$, compute the direction $\mu_j$ for allocating the target points by $\mu_j = W_s(d_j)\mathbf{n}_j + (1 - W_s(d_j))\mathbf{t}_j$.
6: For each point $\mathbf{y}_j$ on the boundary, allocate $p$ scaled Gauss–Legendre nodes on the interval $[-L, L]$ centered at $\mathbf{y}_j$ along the direction $\mu_j$; altogether we obtain $pN_S$ target points $\mathbf{x}_i$ for $i = 1, \ldots, qN_s$.

***Remark 1*** Note that the single subscript of $\mathbf{y}_j$ ($j = 1, \ldots, N_S$) comes from renumerating the set of source points according to the following order:

$$\mathbf{y}_{10}, \mathbf{y}_{11}, \ldots, \mathbf{y}_{1p}, \mathbf{y}_{20}, \mathbf{y}_{21}, \ldots, \mathbf{y}_{2p}, \ldots, \mathbf{y}_{K_S 0}, \mathbf{y}_{K_S 1}, \ldots, \mathbf{y}_{K_S p}$$

for the 2D case where $N_S = K_S p$ and

$$\mathbf{y}_{11}, \mathbf{y}_{12}, \ldots, \mathbf{y}_{1N}, \mathbf{y}_{21}, \mathbf{y}_{22}, \ldots, \mathbf{y}_{2N}, \ldots, \mathbf{y}_{K_S 1}, \mathbf{y}_{K_S 2}, \ldots, \mathbf{y}_{K_S N}$$

for the 3D case where $N_S = K_S N$ and $N = N_1 N_2$. For simplicity, we will use the single subscript later.

Starting from a set of source points $\mathbf{y}_j$ ($j = 1, \ldots, T_s$), we first apply Algorithm 3 to allocate target points on both sides of all source points and apply Algorithm 2 to compute the solution to the initial value problem (18) to obtain the values of $\tilde{\phi}$ (or $\tilde{\psi}$) on the target points. Then, we use a root-finding algorithm (for example, Müller's method in [19]) to find a point whose solution value is equal to $v$ along each given direction computed in Algorithm 3. We summarize the whole scheme in Algorithm 4.

**Algorithm 4** Computing the level set for a given function value $v$.

*Given the prescribed accuracy $\epsilon$, the time step size $\delta t$, a set of source points $\mathbf{y}_j$ ($j = 1, \ldots, T_S$) describing the interface at the current time, and a specified function value $v$, compute the level set for the function value $v$ and return a new set of $T_S$ points representing the new interface after diffusion and thresholding, at which the solution to the initial value problem is equal to $v$. Also return the area or the volume bounded by the new level set.*

1: Apply Algorithm 3 to find the target points and the direction $\mu_j$ for allocating the target points at each source point $\mathbf{y}_j$.
2: Compute the value of $\tilde{\phi}$ (or $\tilde{\psi}$) at these target points by applying Algorithm 1 to solve the initial value problem (18) with the initial data being the indicator function of $D_1$, $D_2$, and $D_3$.
3: For each point $\mathbf{y}_j$ on the boundary along its corresponding direction $\mu_j$, approximate $u(\mathbf{x}(s), \delta t)$ with a $(p-1)$th order Legendre polynomial and use Müller's method [19] to find the parameter value $s \in [-L, L]$ at which the function value is equal to the given value $v$.
4: Calculate the coordinates of the points in the level set by setting $\mathbf{y}_j = \mathbf{y}_j + s \cdot \mu_j$.
5: Use (26) or (29) to compute the area or the volume bounded by the new level set.

In wetting dynamics, we need to find the area- or volume-preserving level set. We can simply add another round of iterations in Algorithm 4. The algorithm is summarized in Algorithm 5. Our numerical experiments show that 4–6 iterations are required to achieve 12-digit accuracy and only 2–3 iterations are needed to reach single precision. Obviously, the cost of the root-finding procedure in Algorithms 4 and 5 is $O(N_s)$.

**Algorithm 5** Calculating the area- or volume-preserving level set.

*Given the prescribed accuracy $\epsilon$, the time step size $\delta t$, and a set of source points $\mathbf{y}_j$ ($j = 1, \ldots, N_S$) describing the interface at the current time, return a new set of $T_S$ points representing the new interface after diffusion and thresholding, whose area or volume is equal to that bounded by the original set of points.*

1: Use (26) or (29) to compute the area or the volume bounded by the original set of source points.
2: Make initial guesses for the solution values $v_0$, $v_1$, $v_2$ randomly but close to 0 for example, and apply Algorithm 4 to find the level sets and the areas or the volumes associated with $v_i$ ($i = 0, 1, 2$).
3: Use Müller's method to find the solution value $v$ and associated level set whose enclosed area or volume is equal to that bounded by the original set of points.
4: Return the new set of points and the solution value $v$.

## 5 Numerical Experiments

We have implemented the aforementioned algorithms in both Fortran and MATLAB. We used the NUFFT library from [15]. We now illustrate the performance of our algorithm via several numerical examples. All results were obtained on a laptop with a 2.3GHz Intel Core i5 processor and 8GB of RAM.

### 5.1 Two-Dimensional Results

***Example 1*** *Non-smooth two-phase interface motion—Mean curvature motion of the hexagram in 2D.* To show the capability of our method to simulate the dynamics of a non-smooth interface in 2D, we consider the mean curvature motion of an initially non-smooth hexagram in two dimensions. We first check the accuracy of the NUFFT-based heat solver in Algorithm 2 in two dimensions for the non-smooth case. For the 2D solver, we use the hexagram in Fig. 5 as the boundary and compute the solution to the initial value problem with $\delta t = 0.0005$. The numerical solutions are evaluated at $N_T = p \cdot N_s = 16 \times 1440$ fixed target points with $p = 16$ points along each proper direction (for example, see Fig. 4) of the source points. Table 1 shows the relative $L^2$ error of the numerical solution of the heat diffusion equation at $t = \delta t$ with various numbers of source points on the boundary.

We set $\delta t = 0.0005$ and use 360 points to discretize the interface with $p = 16$. The computational domain is $[-1, 1]^2$. Figure 5 displays the snapshots of the mean curvature motion of the hexagram at different times. The simulation shows that our algorithm is very robust for the nonsmooth case even when the hexagram shrinks.

***Example 2*** *Non-smooth three-phase interface motion and wetting on a solid surface.* We now apply our algorithms to simulate the dynamics of the liquid–vapor interface when the liquid drop is placed on a solid surface. The computation domain is $[-1, 1]^2$. We use 400 points to discretize the interface with $p = 16$. The target points are distributed as in Fig. 3.

Figure 6 shows the snapshots of the wetting process on a hydrophilic solid material, where the profile of the liquid drop gradually changes from a half-circle to an arc with contact angle

**Table 1** Relative $L^2$ error versus number of discretization points on the boundary for the 2D heat solver

| $N_S$ | $30 \times 12$ | $40 \times 12$ | $50 \times 12$ | $60 \times 12$ | $70 \times 12$ |
|---|---|---|---|---|---|
| Error | 3.396e−6 | 5.917e−8 | 5.847e−10 | 3.373e−12 | 1.629e−14 |

The boundary curve is a non-smooth hexagram shown in Fig. 4. The reference solution is obtained with $N_s = 120 \times 12$ points on the boundary

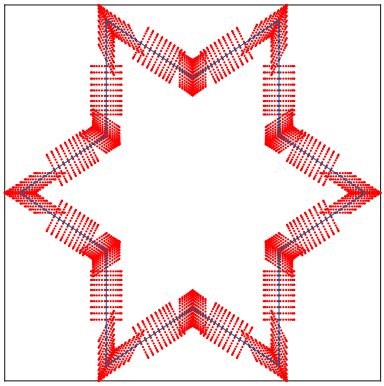

**Fig. 4** Target points generated along a non-smooth hexagram interface

as $\pi/3$. Figure 7 displays the snapshots of the wetting process on a hydrophobic solid material, where the profile of the liquid drop gradually changes from a half circle to an arc with contact angle $2\pi/3$. Both figures are generated with $\delta t = 0.0005$.

The simulation of the hydrophilic solid material using our scheme takes about 98.7 s. We have also run the same case with the same accuracy using the uniform mesh method in [31] with $4096 \times 4096$ grid points ($\delta = 0.001$). It takes 4649 s. For the hydrophobic case, our scheme takes about 101.5 s, whereas the uniform mesh method in [31] takes 4389 s, where the computational domain is again discretized into $4096 \times 4096$ grid points. Therefore, the computational cost is reduced by a factor of about 40 with the current method. The efficiency gain in our scheme mainly comes from limiting the calculation to a neighborhood of the material interfaces and applying the NUFFT to speed up discrete summations.

To show the convergence of Algorithm 1 with respect to $\delta t$, we first check the convergence of dynamics through computing the solution at $t = 0.04$ by using $\delta t = 0.008, 0.004, 0.002, 0.001, 0.0005$ with $\theta_Y = \pi/3$. The reference solution is obtained by setting $\delta t = 0.000125$ and using 400 points to discretize the interface with $p = 16$. Table 2 lists the relative $L^2$ errors of the dynamic solution at $t = 0.04$ with various values of $\delta t$. The convergence order is consistent with the results in [23,31].

We then check the convergence of the equilibrium solutions with respect to $\delta t$. We set $\theta_Y = \pi/3$ and compute the equilibrium solutions by using $\delta t = 0.008, 0.004, 0.002, 0.001, 0.0005$. The exact solution for the equilibrium solution can be computed because the equilibrium solution is a circular curve with the correct contact angle $\theta_Y = \pi/3$ and preserved volume. More details on the equilibrium solution for wetting dynamics can be found in [7]. Table 3 lists the relative $L^2$ errors of the solution at the equilibrium with various values of $\delta t$. The convergence order is consistent with the results in [23,31].

***Example 3*** *Interface motion with topological changes—Two droplets merging in 2D.* In this example, we show that our algorithms can handle the cases involving topological changes as well. We simulate the process of two droplets merging. Initially, we have two droplets on

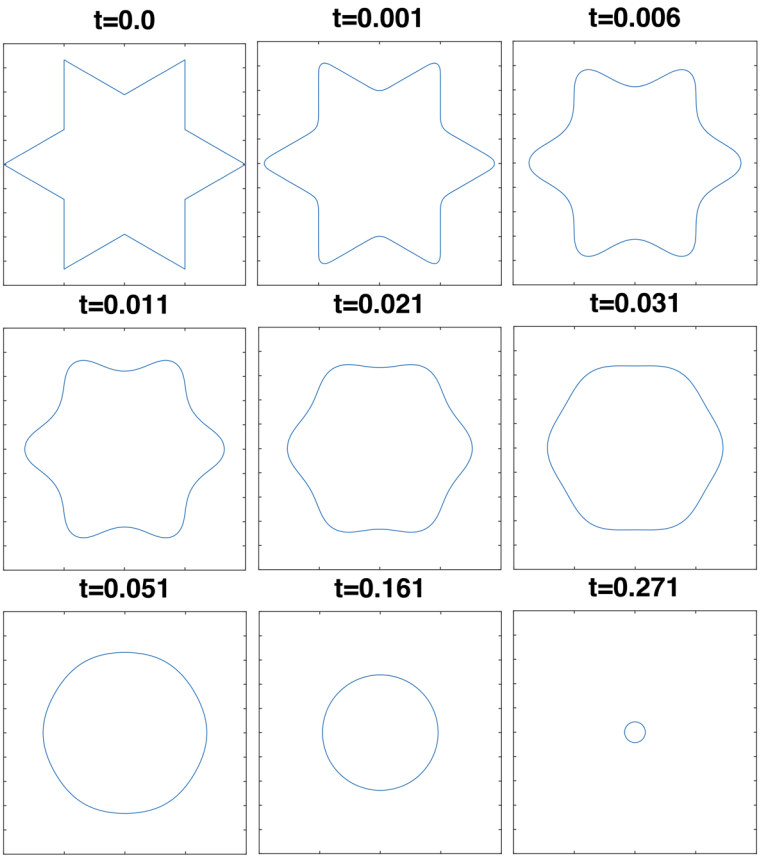


**Fig. 5** Snapshots of the mean curvature motion of a hexagram. $\delta t = 0.0005$. The computational domain is $[-1, 1]^2$

the solid surface as in Fig. 8. We set $\theta_Y = \pi/3$ and the two droplets will spread on the solid surface. Theoretically, the two droplets will merge when the two contact points meet.

In the implementation, we represent the interface of each droplet using 200 points. Because the contact point only moves along the solid surface, we track the distance between the right contact point of the left droplet and the left contact point of the right droplet. If the distance is smaller than a given tolerance $\varepsilon$, we connect the two curves and treat them as one connected curve by merging these two sets of points. Then we distribute the target points along the normal direction of the new single interface and continue the simulation as we did in Example 2.

We set $\delta t = 0.0002$ and use 200 points to discretize each interface with $p = 16$ in Algorithm 3. Figure 9 displays the snapshots of the volume-preserving motion of the two droplets on the solid surface at different times. From these snapshots, we see that the two droplets merge and the profile eventually becomes an arc with equilibrium contact angle $\pi/3$. This is in agreement with the numerical simulation in [31] and the experimental results.

***Example 4*** *Wetting on chemically patterned surfaces in 2D.* To model solid wetting dynamics on chemically patterned surfaces, we use the NUFFT-based heat solver to solve the free space heat diffusion equation to obtain $\psi_1$ and $\psi_2$. The thresholding step is the same as before.

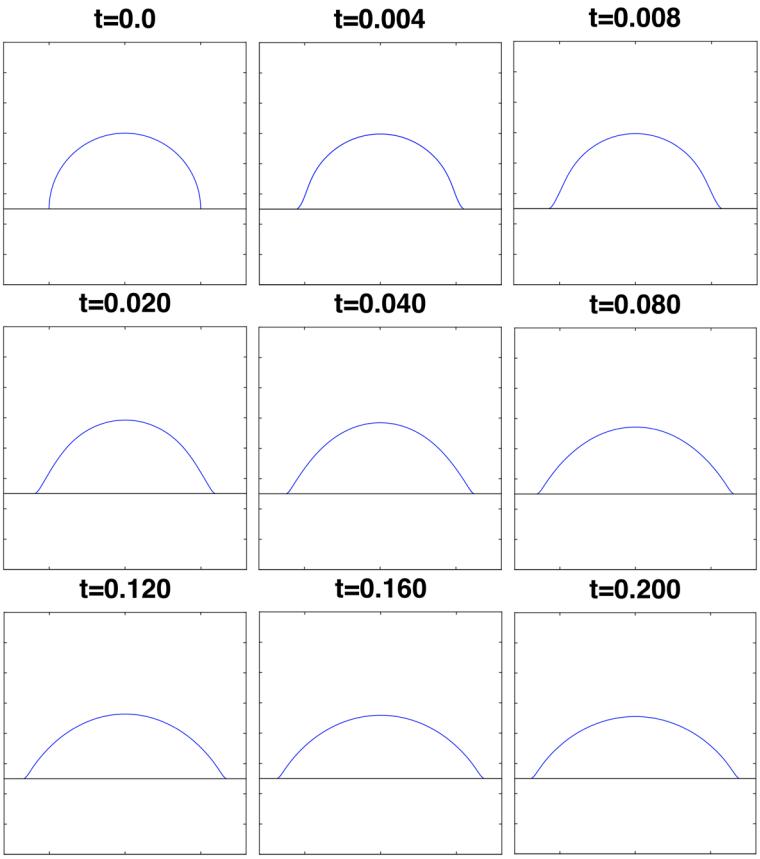


**Fig. 6** Snapshots of the wetting process on a hydrophilic solid surface with Young's angle $\pi/3$. The computational domain is $[-1, 1]^2$

**Table 2** Relative $L^2$ error versus $\delta t$ for the solution in wetting dynamics at $t = 0.04$

| $\delta t$ | 0.008 | 0.004 | 0.002 | 0.001 | 0.0005 |
|---|---|---|---|---|---|
| Error | 1.511e−2 | 9.347e−3 | 5.630e−3 | 3.283e−3 | 1.738e−3 |

The reference solution is obtained with $\delta t = 0.000125$

**Table 3** Relative $L^2$ error versus $\delta t$ for the solution of the equilibrium state

| $\delta t$ | 0.008 | 0.004 | 0.002 | 0.001 | 0.0005 |
|---|---|---|---|---|---|
| Error | 1.631e−2 | 9.056e−3 | 4.509e−3 | 2.217e−3 | 8.803e−4 |

Figure 10 displays the snapshots of the wetting dynamics on the same chemically patterned surface with different initial conditions. For different initial conditions, the contact points are pinned at different positions at the equilibrium displaying the phenomenon of contact angle hysteresis. In this simulation, $\delta t = 0.0005$, Young's angle in material $\mathcal{A}$ (i.e., the red domain

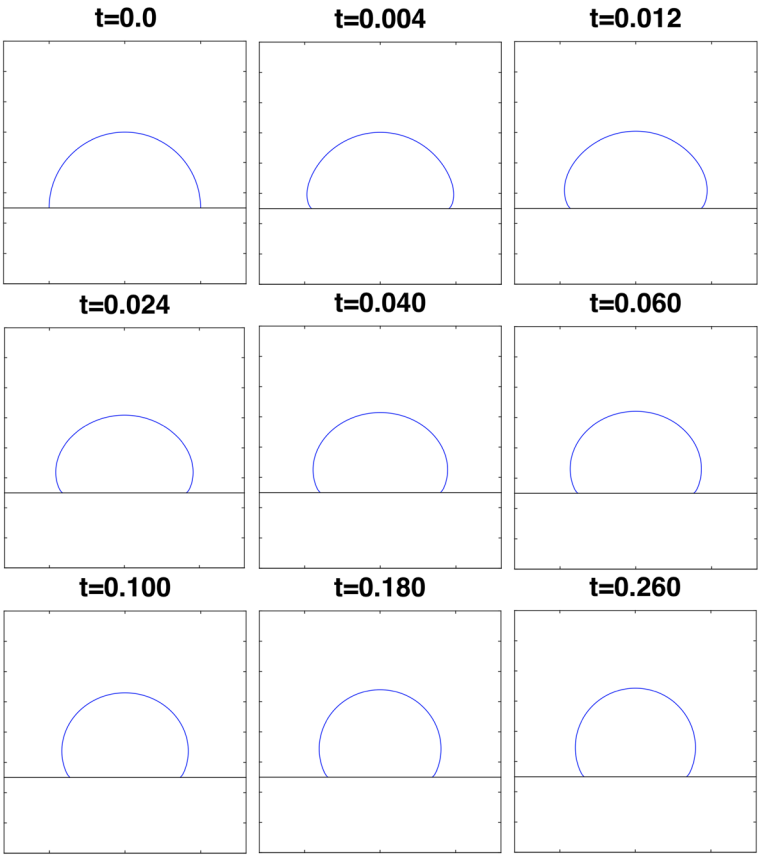


**Fig. 7** Snapshots of the wetting process on a hydrophobic solid surface with Young's angle $2\pi/3$. The computational domain is $[-1, 1]^2$

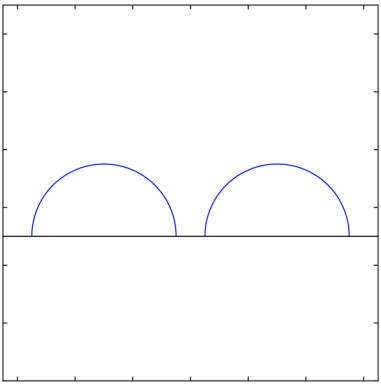

**Fig. 8** Initial profile of two droplets on a solid surface

$D_3$ in Fig. 10) is $2\pi/3$ and Young's angle in material $\mathcal{B}$ (i.e., the white domain $D_4$ in Fig. 10) is $\pi/3$. The interface is discretized with 400 points and $p = 16$.

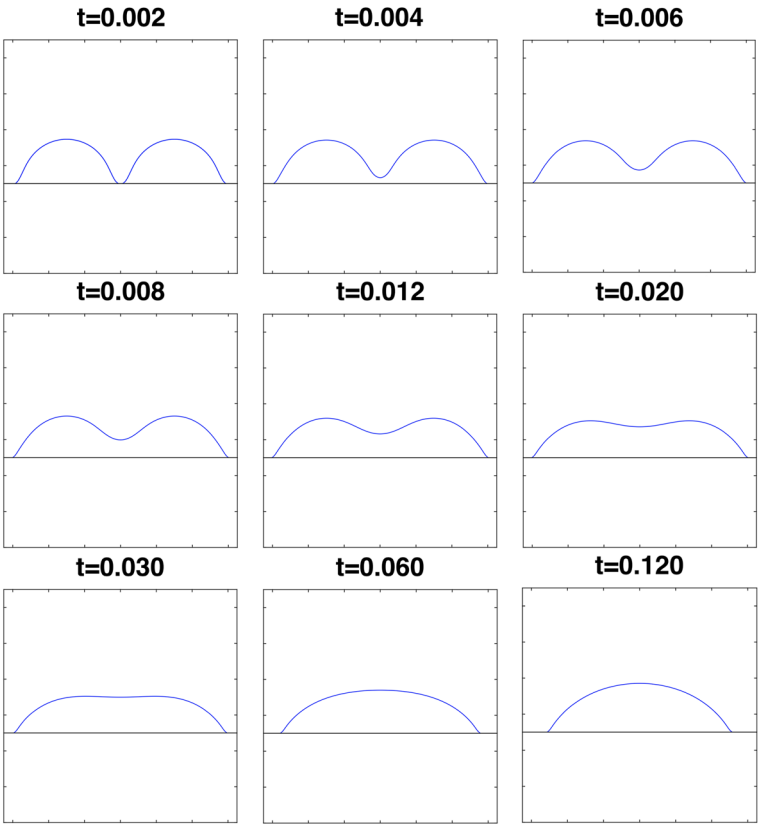


**Fig. 9** Snapshots of two droplets merging on a solid surface with Young's angle $\pi/3$. The computational domain is $[-1, 1]^2$

### 5.2 Three-Dimensional Results

***Example 5*** *Wetting in 3D.* We now consider the wetting of a droplet on the solid surface in three dimensions. Figure 11 displays the snapshots of the solid wetting process on a hydrophobic solid surface. In this simulation, we set $\delta t = 0.002$ and use $96 \times 96$ points to discretize the surface of the droplet with $p = 16$ with Young's angle $\pi/3$. Figure 12 displays the snapshots of the solid wetting process on hydrophilic solid surfaces. In this simulation, we set $\delta t = 0.0005$ and use $96 \times 96$ points to discretize the surface of droplet with $p = 16$ with Young's angle $2\pi/3$. The results in Figs. 11 and 12 show that our numerical results agree strongly with the predicted theoretical results.

Simulating the wetting on a hydrophobic surface takes about 1090 s whereas simulating the wetting on a hydrophilic surface takes about 4111 s. The algorithm in [31] would require at least $1024 \times 1024 \times 1024$ uniform grid points in the computational box to reach a similar level of accuracy for such 3D simulations. Even if we ignore the memory constraint, the whole simulation would take about 600 h on the same laptop to reach equilibrium, making the algorithm in [31] impractical for 3D problems.

***Example 6*** *Solid wetting on chemically patterned surfaces in 3D.*

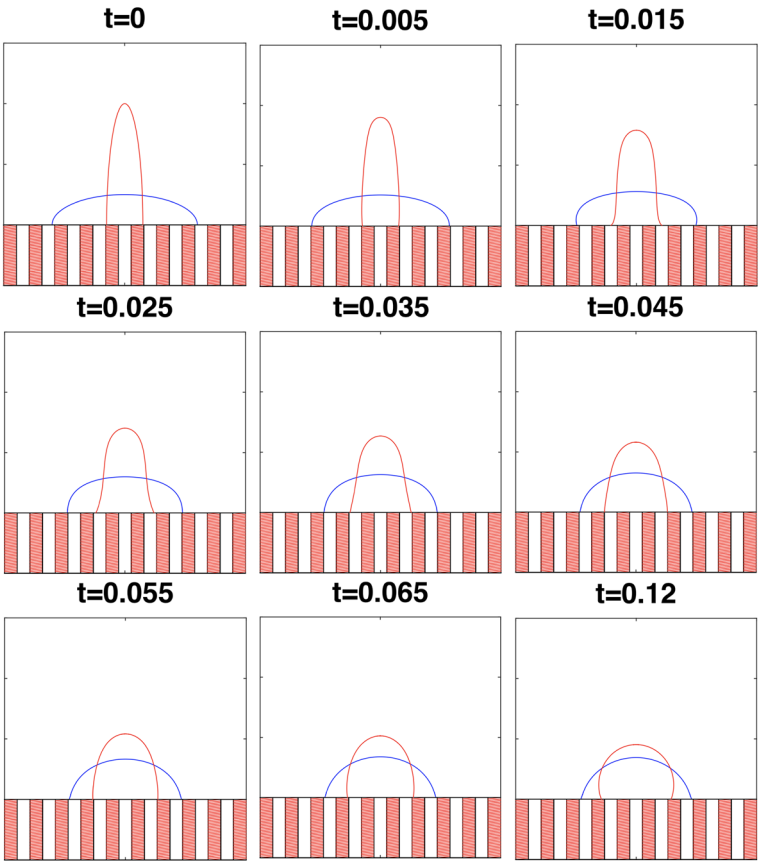


**Fig. 10** Snapshots of wetting dynamics on a chemically patterned surface in 2D. In this simulation, $\delta t = 0.0005$, Young's angle in material $\mathcal{A}$ (i.e., the red domain $D_3$) is $2\pi/3$ and Young's angle in material $\mathcal{B}$ (i.e., the white domain $D_4$) is $\pi/3$. The computational domain is $[-1, 1]^2$

To verify the performance of our algorithm in the 3D simulation, we compare our results with the phase field simulation in [13]. We use the same initial condition and the same pattern on the solid surface. That is, two intersecting hydrophobic strips of width 0.1 with Young's angle $2\pi/3$ are placed at the center of an otherwise hydrophilic background with Young's angle $\pi/4$. Figure 13 displays the snapshots of similar dynamics of the interface as that in [13, Figure 8]. Figure 14 shows that the droplet spreads out on the hydrophilic surface (blue area) and contracts on hydrophobic strips (white area) and eventually reaches an equilibrium state. This is consistent with the result in [13, Figure 9]. In this simulation, we set $\delta t = 0.0005$ and use $96 \times 96$ points to discretize the surface of the droplet with $p = 16$.

## 6 Conclusions and Future Discussions

In this paper, we have extended the algorithm in [17] for smooth curves/surfaces to deal with cases involving non-smooth curves/surfaces and cases involving topological changes. We have also applied the extended algorithm to study the wetting dynamics in both two and three dimensions. The algorithm avoids discretizing of the entire volume and does not need to

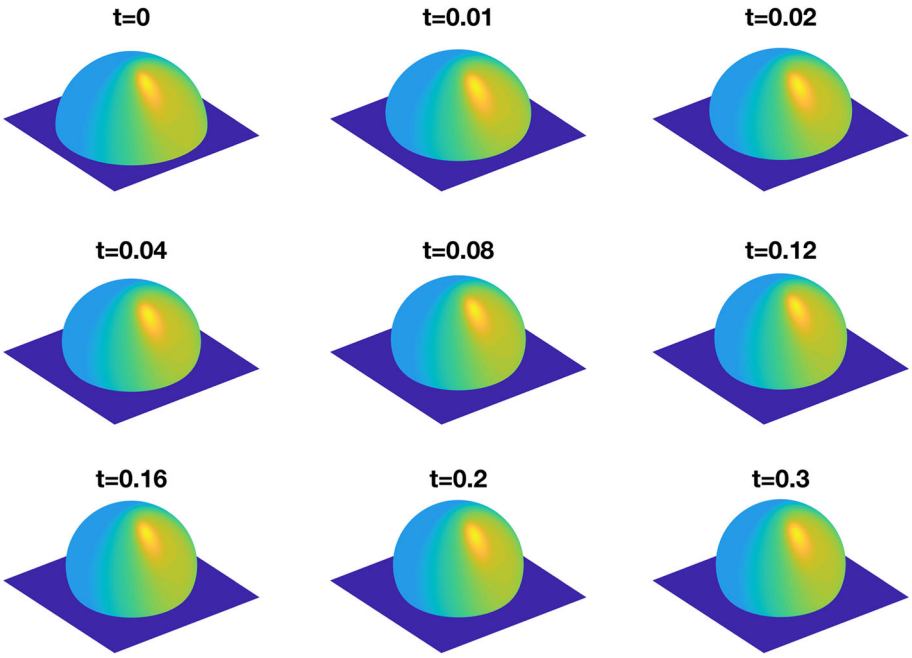


**Fig. 11** Snapshots of the wetting process on a hydrophobic solid material with Young's angle $2\pi/3$. The CPU time for the whole simulation is 1090 s. The computational domain is $[-0.6, 0.6] \times [-0.6, 0.6] \times [-0.5, 0.2]$

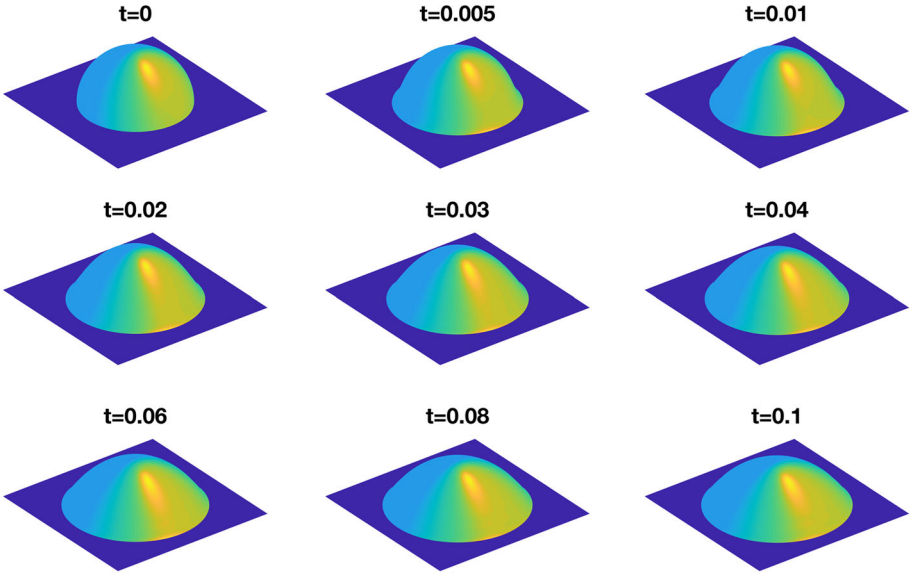


**Fig. 12** Snapshots of the wetting process on a hydrophilic solid material with Young's angle $\pi/3$. The CPU time for the whole simulation is 4111 s. The computational domain is $[-0.8, 0.8] \times [-0.8, 0.8] \times [-0.5, 0.1]$

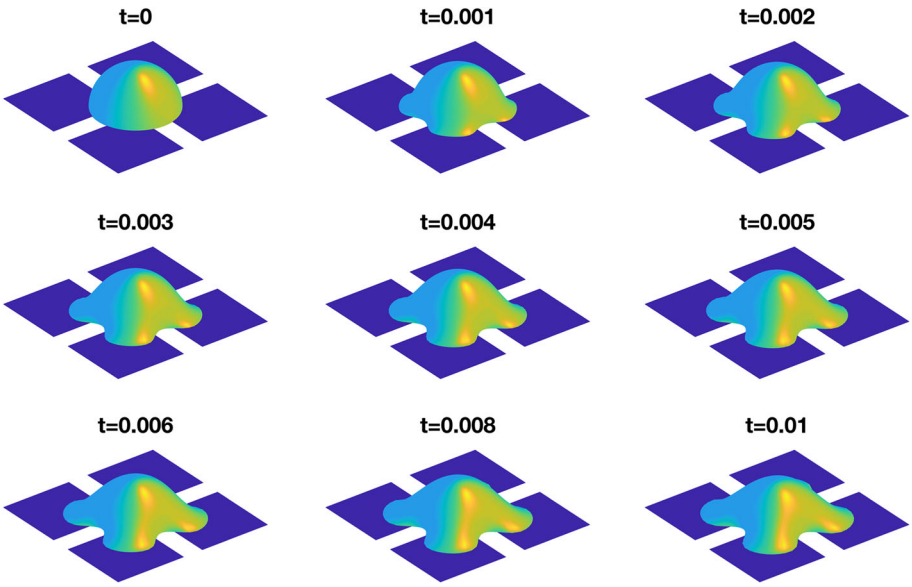


**Fig. 13** Snapshots of wetting dynamics on a chemically patterned surface in 3D. In this simulation, $\delta t = 0.0005$, Young's angle in material $\mathcal{A}$ (i.e., the blue domain $D_3$) is $\pi/4$ and Young's angle in material $\mathcal{B}$ (i.e., the white domain $D_4$) is $2\pi/3$. The computational domain is $[0, 0.8] \times [0, 0.8] \times [-0.8, 0]$

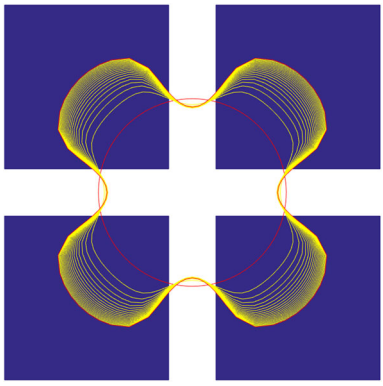

**Fig. 14** Positions of the contact line between time $t = 0$ and $t = 0.01$ at time intervals of 0.0005

truncate the computational domain. Hence, an expensive uniform grid for the discretization of a large volumic domain is not needed and one only needs a nearly optimal number of discretization points in the neighborhood of the material interfaces. When combined with the NUFFT, the algorithm achieves nearly optimal complexity with much fewer number of unknowns. Our numerical experiments demonstrate that the algorithm is highly accurate and efficient compared with the existing algorithms. We expect that the algorithm will be of practical use for the study of wetting dynamics on both single-material solid surfaces and chemically patterned solid surfaces.

**Acknowledgements** S. Jiang was supported by the National Science Foundation under Grant DMS-1720405 and by the Flatiron Institute, a division of the Simons Foundation. The work of X.P. Wang was partially supported by the Hong Kong Research Grants Council (GRF Grants 16302715, 16324416, and 16303318).

Part of the work was performed when the authors were participating in the HKUST-ICERM VI-MSS program 'Integral Equation Methods, Fast Algorithms and Their Applications to Fluid Dynamics and Materials Science' held in 2017.

## References

1. Alberti, G., DeSimone, A.: Wetting of rough surfaces: a homogenization approach. Proc. R. Soc. A **451**, 79–97 (2005)
2. Bonn, D., Eggers, J., Indekeu, J., Meunier, J., Rolley, E.: Wetting and spreading. Rev. Mod. Phys. **81**(2), 739 (2009)
3. Bonn, D., Ross, D.: Wetting transitions. Rep. Prog. Phys. **64**(9), 1085 (2001)
4. Cahn, J.W., Hilliard, J.E.: Free energy of a nonuniform system. I. Interfacial free energy. J. Chem. Phys. **28**(2), 258–267 (1958)
5. Cahn, J.W., Hilliard, J.E.: Free energy of a nonuniform system. III. Nucleation in a two-component incompressible fluid. J. Chem. Phys. **31**(3), 688–699 (1959)
6. Chen, H., Leng, H., Wang, D., Wang, X.P.: An efficient threshold dynamics method for topology optimization for fluids. arXiv preprint arXiv:1812.09437 (2018)
7. Chen, X., Wang, X., Xu, X.: Analysis of the Cahn–Hilliard equation with a relaxation boundary condition modeling the contact angle dynamics. Arch. Ration. Mech. Anal. **213**(1), 1–24 (2014)
8. Dutt, A., Rokhlin, V.: Fast Fourier transforms for nonequispaced data. SIAM J. Sci. Comput. **14**, 1368–1393 (1993)
9. Dutt, A., Rokhlin, V.: Fast Fourier transforms for nonequispaced data. II. Appl. Comput. Harmon. Anal. **2**, 85–100 (1995)
10. Erbil, H.Y.: The debate on the dependence of apparent contact angles on drop contact area or three-phase contact line: a review. Surf. Sci. Rep. **69**(4), 325–365 (2014)
11. Esedoglu, S., Otto, F.: Threshold dynamics for networks with arbitrary surface tensions. Commun. Pur. Appl. Math. **68**(5), 808–864 (2015)
12. Extrand, C.W.: Model for contact angles and hysteresis on rough and ultraphobic surfaces. Langmuir **18**, 7991–7999 (2002)
13. Gao, M., Wang, X.P.: An efficient scheme for a phase field model for the moving contact line problem with variable density and viscosity. J. Comput. Phys. **272**, 704–718 (2014)
14. de Gennes, P., Brochard-Wyart, F., Quere, D.: Capillarity and Wetting Phenomena. Springer, Berlin (2003)
15. Greengard, L., Lee, J.Y.: Accelerating the nonuniform fast Fourier transform. SIAM Rev. **46**(3), 443–454 (2004)
16. Greengard, L., Lin, P.: Spectral approximation of the free-space heat kernel. Appl. Comput. Harmon. Anal. **9**(1), 83–97 (2000)
17. Jiang, S., Wang, D., Wang, X.P.: An efficient boundary integral scheme for the MBO threshold dynamics method via the NUFFT. J. Sci. Comput. **74**, 474–490 (2018)
18. Leung, S., Zhao, H.: A grid based particle method for moving interface problems. J. Comput. Phys. **228**(8), 2993–3024 (2009)
19. Müller, D.E.: A method for solving algebraic equations using an automatic computer. Math. Tables Aids Comput. **10**, 208–215 (1956)
20. Osting, B., Wang, D.: A diffusion generated motion for orthogonal matrix-valued fields. Math. Comp. (2019) (accepted)
21. Quere, D.: Wetting and roughness. Annu. Rev. Mater. Res. **38**, 71–99 (2008)
22. Randol, B.: On the Fourier transform of the indicator function of a planar set. Trans. Am. Math. Soc. **139**, 271–278 (1969)
23. Wang, D.: Threshold dynamics method: theories, algorithms, and applications. Ph.D. thesis, Hong Kong University of Science and Technology (2017)
24. Wang, D., Li, H., Wei, X., Wang, X.P.: An efficient iterative thresholding method for image segmentation. J. Comput. Phys. **350**, 657–667 (2017)
25. Wang, D., Osting, B.: A diffusion generated method for computing Dirichlet partitions. J. Comput. Appl. Math. **351**, 302–316 (2019)
26. Wang, D., Osting, B., Wang, X.P.: Interface dynamics for an Allen–Cahn-type equation governing a matrix-valued field. Multiscale Model. Sim. (2019) (accepted)
27. Wang, D., Wang, X.P.: The iterative convolution-thresholding method (ICTM) for image segmentation. arXiv preprint arXiv:1904.10917 (2019)

28. Wang, D., Wang, X.P., Xu, X.: An improved threshold dynamics method for wetting dynamics. J. Comput. Phys. **392**, 291–310 (2019)
29. Whyman, G., Bormashenko, E., Stein, T.: The rigorous derivative of young, Cassie–Baxter and Wenzel equations and the analysis of the contact angle hysteresis phenomenon. Chem. Phys. Lett. **450**, 355–359 (2008)
30. Womble, D.E.: A front-tracking method for multiphase free boundary problems. SIAM J. Numer. Anal. **26**(2), 380–396 (1989)
31. Xu, X., Wang, D., Wang, X.P.: An efficient threshold dynamics method for wetting on rough surfaces. J. Comput. Phys. **330**, 510–528 (2017)
32. Xu, X., Wang, X.P.: The modified Cassie's equation and contact angle hysteresis. Colloid Polym. Sci. **291**, 299–306 (2013)
33. Young, T.: An essay on the cohesion of fluids. Philos. Trans. R. Soc. Lond. **95**, 65–87 (1805)
34. Zhao, H., Chan, T., Merriman, B., Osher, S.J.: A variational level set approach to multiphase motion. J. Comput. Phys. **127**(1), 179–195 (1996)