ICES REPORT 16-08

March 2016

An implicit boundary integral method for interfaces evolving by Mullins-Sekerka dynamics

by

Chieh Chen, Catherine Kublik, and Richard Tsai

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Chieh Chen, Catherine Kublik, and Richard Tsai, "An implicit boundary integral method for interfaces evolving by Mullins-Sekerka dynamics," ICES REPORT 16-08, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, March 2016.

An implicit boundary integral method for interfaces evolving by Mullins-Sekerka dynamics

Chieh Chen, Catherine Kublik and Richard Tsai

1 Introduction

This paper proposes an algorithm for simulating interfacial motions in Mullins-Sekerka dynamics using a boundary integral equations formulated on implicit surfaces. The proposed method is able to simulate the dynamics on unbounded domains, and as such will provide a tool to better understand the dynamics obtained in these situations. As pointed out in Gurtin's paper [23], the behavior of the dynamics are far more interesting on domains that are unbounded. However, it seems that there is no other computational method that can simulate the nonlinear interfacial motion of Mullins-Sekerka dynamics in three dimensions on unbounded domains, allowing the interfaces to merge, break up, without the need of finding explicit representations.

The Mullins-Sekerka flow is a Stefan-type free boundary problem involving an nonlocal interfacial motion dynamically controlled by the solution of Laplace's equation with appropriate boundary conditions obtained on both sides of the interface. As a result, it is reasonable to consider boundary integral methods, especially for exterior domains, combined with level set methods [40][20], for easy tracking and for being able to handle topological changes.

Various numerical methods have been proposed to solve elliptic problems on irregular domains. We mention here some of the few "usual suspects" that use fi-

Chieh Chen
Goldman Sachs, 6011 Connection Drive, Irving TX, 75039 USA e-mail: Chieh.Chen@gs.com

Catherine Kublik
Department of Mathematics, University of Dayton, 300 College Park, Dayton OH, 45469 USA e-mail: ckublik1@udayton.edu

Richard Tsai
Department of Mathematics and Institute for Computational Engineering and Sciences, University of Texas at Austin, 2515 Speedway, 78712 Austin, USA.
Department of Mathematics, KTH Royal Institute of Technology, SE-100 44, Stockholm, Sweden e-mail: tsai@kth.se

nite element methods [4, 15, 24, 25, 26], finite difference techniques [7, 11, 18, 27, 28, 33], and boundary integral methods [3, 46]. Finite element methods require an explicit representation (e.g. triangulation) of the domain which makes them less tractable for an evolving domain. There is also a wide range of finite difference methods for solving elliptic equations, some of them using the level set method [19, 35, 18]. The Immersed Interface Method (IIM) [33, 34] is a popular technique for solving elliptic equations, particularly if the coefficients in the equation are discontinuous. This technique uses an adaptive finite difference scheme with a locally adaptive stencil. Unlike finite element method, the IIM can be used with an implicit representation of the interface. Nevertheless, both finite element and finite difference based techniques discretize the whole domain and therefore cannot be applied directly to "exterior" problems without imposing additional artificial boundary conditions.

In contrast, boundary integral methods (BIMs) use an integral representation of the solution, namely the solution is defined by an integral of a suitable potential over the interface. If the domain boundaries are fixed and suitable parameterizations are available, BIMs can be a natural choice to solve exterior problems.

In this paper, we present an implicit boundary integral algorithm for solving the following free-boundary problem

$$\begin{cases} \Delta u(\mathbf{x},t) = 0 & \mathbf{x} \in \mathbb{R}^m \setminus \Gamma_t,\ m = 2 \text{ or } 3, \\ u(\mathbf{x}) = -\kappa(\mathbf{x}) & \mathbf{x} \in \Gamma_t, t \geq 0, \\ v_n = -\left[\frac{\partial u}{\partial \mathbf{n}}\right]_{\Gamma_t} & \mathbf{x} \in \Gamma_t, t \geq 0, \\ \lim_{|\mathbf{x}|\to\infty} |u(\mathbf{x})| < \infty, & m = 2, \\ \lim_{|\mathbf{x}|\to\infty} |u(\mathbf{x})| = 0, & m = 3. \\ \Gamma_0 = \partial\Omega_0^-. \end{cases} \tag{1}$$

This problem is sometimes referred to as the Mullins-Sekerka problem [38] which was first studied as a solidification process. In general, this problem models the solidification or liquidation of materials of negligible specific heat, where the function u represents the scaled temperature and Γ_t is the boundary of the bounded domain Ω_t, namely $\Gamma_t = \partial\Omega_t$ at time t. The boundary condition $u = -\kappa$ comes from the Gibbs-Thomson relation and is related to the surface tension effect. In two dimensions, the boundary condition at infinity (assuming that the temperature is uniformly bounded) produces an isolated boundary in which the flux $\frac{\partial u}{\partial n}$ is zero at infinity. In three dimensions, the boundary condition at infinity is an isothermal boundary condition which models the liquidation of material.

In this setup, Ω_t is the solid region and Ω_t^c is the liquid region. $\left[\frac{\partial u}{\partial \mathbf{n}}\right]_{\Gamma_t}$ is the jump in the normal derivative of the function u across the interface Γ_t defined as the sum of the outward normal derivatives of u from each side of Γ_t. More precisely,

$$\left[\frac{\partial u}{\partial \mathbf{n}}\right] := \left(\frac{\partial u}{\partial \mathbf{n}^+}\right)_- - \left(\frac{\partial u}{\partial \mathbf{n}^+}\right)_+, \tag{2}$$

where $+$ and $-$ refer to outside and inside respectively.

We observe that the two dimensional Mullins-Sekerka problem has nontrivial stationary solutions. For example, a collection of disjoint circles with the same radius R will be stationary under the Mullins-Sekerka dynamics. To see this, we note that in this case, both the solution inside and outside will be constant equal to $-\frac{1}{R}$, and thus $v_n = 0$.

This free boundary problem has interesting properties concerning volume and surface area. See for example [13], where coarsening rates in Mullins-Sekerka evolution in bounded domains with periodic conditions are studied.

In the following, we describe some notable differences in evolutions defined in two and three dimensions. Let $A(t)$ denote the surface area of Γ_t at time t and $V(t)$ denote the volume of Ω_t at time t. Then we have

$$\begin{aligned}\frac{dV(t)}{dt} &= -\int_{\Gamma_t} v_n ds \\ &= \int_{\Gamma_t} \left[\frac{\partial u}{\partial n}\right] ds \\ &= \int_{\Omega_t} \Delta u dx - \int_{\Gamma_t} \left(\frac{\partial u}{\partial \mathbf{n}^+}\right)_+ ds.\end{aligned}$$

The first term is zero because u is harmonic inside Ω_t. In two dimensions, the second term turns out to be zero because of the compatibility condition (17) (see Appendix 7), and thus the total volumeenclosed by Γ_t is conserved. This compatibility condition is not valid in three dimensions and hence the motion is not volume preserving in three dimensions.

For the area, we have the following:

$$\begin{aligned}\frac{dA(t)}{dt} &= -\int_{\Gamma_t} \kappa v_n ds \\ &= -\int_{\Gamma_t} u \left[\frac{\partial u}{\partial n}\right] ds \\ &= \int_{\Gamma_t} u (\nabla u)_+ \cdot \mathbf{n}^+ ds - \int_{\Omega_t} |\nabla u|^2 dx - \underbrace{\int_{\Omega_t} \Delta u dx}_{=0}.\end{aligned}$$

In two dimensions, we can use equation (21) derived in Appendix 7 to obtain

$$\frac{dA(t)}{dt} = -\int_{\mathbb{R}^2 \setminus \Gamma_t} |\nabla u|^2 dx, \tag{3}$$

namely the evolution in two dimensions is area preserving while decreasing the length of the boundary.

Another interesting property of the Mullins-Sekerka dynamics is that the two dimensional motion induced by (1) does not preserve convexity, see e.g. [6].

In this paper, we solve (1) using an implicit boundary interface method, namely we define Γ_t implicitly through a level set function φ, namely at any time $t \geq 0$,

$$\Gamma_t = \{\mathbf{x} \in \mathbb{R}^m : \varphi(\mathbf{x},t) = 0\}.$$

The algorithm operates by alternatively solving for the solution u while maintaining the interface fixed, and then propagating the interface according to the normal velocity obtained from the jump in the normal derivative of u across the interface. In the present work, the solution of the boundary value problem given by the first three equations in (1) is solved using an implicit boundary integral formulation [31, 32] that allows the boundary integral to be rewritten as an integral over $\mathbb{R}^m$.

2 Integral equations for Laplace's equation

We present below the boundary integral equation formulations most relevant to this paper and some of their useful properties. Throughout the paper, we consider $\Omega \subset \mathbb{R}^m$ to be a bounded set, $m = 2$ or 3 and the boundary $\partial\Omega = \Gamma$ to be a disjoint collection of closed compact C^2 interfaces. We also denote by $\mathbf{n}^+$ and $\mathbf{n}^-$ the outward and the inward unit normal vectors respectively. The exterior is defined to be the unbounded domain.

We consider the Dirichlet problem (interior and exterior) in an $L+1$ connected region (bounded or unbounded) with L holes labeled $\{S_i\}_{i=1}^L$ each having boundary $\Gamma_i = \partial S_i$:

$$\begin{cases} \Delta u(\mathbf{x}) = 0 & \mathbf{x} \in \Omega \\ u(\mathbf{x}) = f(\mathbf{x}), & \mathbf{x} \in \Gamma, \end{cases} \tag{4}$$

and

$$\begin{cases} \Delta u(\mathbf{x}) = 0 & \mathbf{x} \in \mathbb{R}^m \setminus \bar{\Omega}, \\ u(\mathbf{x}) = f(\mathbf{x}), & \mathbf{x} \in \Gamma, \\ \lim_{|\mathbf{x}|\to\infty} |u(\mathbf{x})| < \infty, & m = 2 \\ \lim_{|\mathbf{x}|\to\infty} |u(\mathbf{x})| = 0, & m = 3. \end{cases} \tag{5}$$

Since Dirichlet boundary conditions are imposed, we introduce an unknown density β defined on Γ and represent the solution of (4) using the double layer potential formulation

$$u(\mathbf{x}) = \int_\Gamma \beta(\mathbf{y}) \frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n}_\mathbf{y}^+} dS(\mathbf{y}), \quad \mathbf{x} \in \Omega,$$

where Φ is the fundamental solution of Laplace's equation defined as

$$\Phi(\mathbf{x},\mathbf{y}) = \begin{cases} \frac{1}{2\pi} \ln|\mathbf{x}-\mathbf{y}| & \text{for } m = 2, \\ -\dfrac{1}{m(m-2)\rho_m |\mathbf{x}-\mathbf{y}|^{m-2}} & \text{for } m \geq 3, \end{cases} \tag{6}$$

where ρ_m is the volume of the unit ball in $\mathbb{R}^m$.

In this context, the naive integral equation is ill-posed, both for the interior and the exterior problem, since there are L nontrivial homogeneous solutions that span the nullspace, see e.g. [17]. To alleviate this issue, we follow the approach first suggested by Mikhlin [36] and developed by Greenbaum etal [21]. The procedure amounts to adding a linear combination $\sum_{i=1}^{L} A_i \Phi(\mathbf{x}-\mathbf{z_i})$ to the integral equation, where $\mathbf{z_i} \in S_i$, and the coefficients A_i play the role of Lagrange multipliers. These L constants give L degrees of freedom that are used to make sure the constructed solution satisfies the appropriate equation and boundary conditions. For the full derivation of the modified integral equation, we refer the reader to the work of Mikhlin and Greenbaum etal [21, 36].

The interior problem is solved as follows:

1. Find the density β and the constants $\{A_i\}_{i=1}^{L}$ such that

$$\int_{\Gamma} \beta(\mathbf{y}) \frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n}_\mathbf{y}^+} dS(\mathbf{y}) + \frac{1}{2}\beta(\mathbf{x}) + \sum_{i=1}^{L} A_i \Phi(\mathbf{x}-\mathbf{z_i}) = f(\mathbf{x}), \text{ for } \mathbf{x} \in \Gamma. \tag{7}$$

$$\int_{\Gamma_i} \beta(\mathbf{y}) dS)\mathbf{y}) = 0, \text{ for } 1 \le i \le L. \tag{8}$$

2. Reconstruct the solution u in Ω using the double layer potential formulation

$$u(\mathbf{x}) = \int_{\Gamma} \beta(\mathbf{y}) \frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n}_\mathbf{y}^+} dS(\mathbf{y}) + \sum_{i=1}^{L} A_i \Phi(\mathbf{x}-\mathbf{z_i}), \quad \text{for } \mathbf{x} \in \Omega.$$

Note that for simply connected Ω the constants A_i are not needed since when $L = 0$ the term $\sum_{i=1}^{0} A_i \Phi(\mathbf{x}-\mathbf{z_i}) = 0$ and equation (8) also removed.

For the exterior problem it is necessary to modify the integral equation in order to obtain an invertible system. It can be shown, e.g. [30], that if the kernel is modified using $\dfrac{1}{|\mathbf{x}-\mathbf{y}|^{m-2}}$, the system becomes well-posed. The exterior problem is therefore solved as follows:

1. Find the density β and the constants $\{A_i\}_{i=1}^{L}$ such that

$$\begin{aligned}
&\int_{\Gamma} \beta(\mathbf{y}) \left(\frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n}_\mathbf{y}^+} - \frac{1}{|\mathbf{x}-\mathbf{y}|^{m-2}} \right) dS(\mathbf{y}) - \frac{1}{2}\beta(\mathbf{x}) + \sum_{i=1}^{L} A_i \Phi(\mathbf{x}-\mathbf{z_i}) = f(\mathbf{x}), \\
&\int_{\Gamma_i} \beta(\mathbf{y}) dS(\mathbf{y}) = 0, \text{ for } 1 \le i \le L-1, \quad \text{for } \mathbf{x} \in \Gamma \\
&m = 2: \quad \sum_{i=1}^{L} A_i = 0, \\
&m = 3: \quad \int_{\Gamma_L} \beta(\mathbf{y}) dS(\mathbf{y}) = 0.
\end{aligned} \tag{9}$$

2. Reconstruct the solution u in $\mathbb{R}^m \setminus \bar{\Omega}$ using the double layer potential formulation

$$u(\mathbf{x}) = \int_{\Gamma} \beta(\mathbf{y}) \frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n_y}} dS(\mathbf{y}) + \sum_{i=1}^{L} A_i \Phi(\mathbf{x} - \mathbf{z_i}), \quad \text{for } \mathbf{x} \in R^m \setminus \bar{\Omega}.$$

The condition $\sum_{i=1}^{L} A_i = 0$ necessary in the two dimensional case ensures that the solution is bounded. This condition is not needed in three dimensions.

3 Overview of existing numerical methods

In this section, we give a brief overview of numerical methods for solving integral equations. Boundary integral methods typically provide highly accurate numerical solutions. However, they require highly accurate parameterizations of the boundaries, high order quadratures and specialized dense matrix solvers.

For each of the boundary integral equations obtained in the previous section, we need to solve a Fredholm equation of the second kind. In other words, we need to find a function γ defined on Γ, such that

$$q(x) = \int_{\Gamma} \gamma(y(s)) K(x, y(s)) ds + C_0 \gamma(x),$$

where C_0 is a constant and K is the normal derivative of the fundamental solution of Laplace's equation to Γ. To solve these equations numerically it is necessary to discretize the above integrals. Three discretization methods are typically used: the Nyström method [2, 39], the collocation method [2] and the Galerkin method [2, 12]. Each of these discretization methods leads to a discrete system of the form

$$(I + \mathtt{K}\Lambda)\gamma = q,$$

where I is the identity matrix, $\mathtt{K}$ is a dense matrix, Λ is a diagonal matrix (for example containing the quadrature weights of the Nyström method), γ is the vector of unknowns, and q is a known vector obtained from the boundary conditions.

Since $\mathtt{K}$ is dense this system is usually solved using an iterative procedure. In addition, low rank approximations may be constructed to improve the efficiency of the numerical solver. One very successful approach is the Fast Multipole Method introduced by Greengard and Rokhlin in 1987 [22]. The use of hierarchical matrices [8] to solve this dense system is also popular.

In this paper, we use an exact integral formulation that allows the boundary integral to be rewritten exactly as a volume integral over a thin tubular neighborhood of the boundary. This enables us to perform the computations on a fixed gird regardless of the location of the interface. Should the interface evolve in time, all computations will be performed on the mesh that is used by the level set function at each time step. This makes the algorithm easy to implement for evolving interfaces in two and three dimensions.

4 Exact integral formulations using signed distance functions

The exact integral formulation we use in this work was first proposed in [31] and extended in [32]. This formulation allows the computation of integrals of the form

$$\int_{\Gamma} v(\mathbf{x}) dS, \tag{10}$$

in the level set framework, namely when the domain Ω is represented implicitly by a level set function.

In [31], with the choice of $\varphi = d_{\partial\Omega}$ being a signed distance function to Γ, the integral (10) is expressed as an average of integrals over nearby level sets of d_Γ, where these nearby level sets continuously sweep a thin tubular neighborhood around the boundary Γ of radius ε. Consequently, (10) is *equivalent* to the volume integral shown on the right hand side below:

$$\int_{\Gamma} v(\mathbf{x}) dS = \int_{\mathbb{R}^n} v(\mathbf{x}^*) J(\mathbf{x}; d_\Gamma) \delta_\varepsilon(d_\Gamma(\mathbf{x})) d\mathbf{x}, \tag{11}$$

where δ_ε is an averaging kernel, $\mathbf{x}^*$ is the closest point on Γ to $\mathbf{x}$ and $J(\mathbf{x}; d_\Gamma)$ accounts for the change in curvature between the nearby level sets and the zero level set. We remark that the author in [37] used a similar formulation to study the heat equation defined in tubular neighborhoods of Γ.

We briefly give a justification for (11). Suppose that Γ is a smooth closed hypersurface in $\mathbb{R}^m$ and assume that $\mathbf{x}$ is sufficiently close to Ω so that the closest point mapping

$$\mathbf{x}^* = P_\Gamma(\mathbf{x}) = \operatorname{argmin}_{y\in\Gamma} |\mathbf{x} - \mathbf{y}|$$

is continuously differentiable. Then the restriction of P_Γ to Γ_η is a diffeormorphism between Γ_η and Γ, where $\Gamma_\eta := \{\mathbf{x} : d_\Gamma(\mathbf{x}) = \eta\}$. As a result, it is possible to write integrals over Γ using points on Γ_η as:

$$\int_{\Gamma} v(\mathbf{x}) dS = \int_{\Gamma_\eta} v(P_\Gamma(\mathbf{x})) J(\mathbf{x}; \eta) dS,$$

where the Jacobian $J(\mathbf{x}, \eta)$ comes from the change of variable defined by P_Γ restricted on Γ_η. Averaging the above integrals respectively with a kernel, δ_ε, compactly supported in $[-\varepsilon, \varepsilon]$, we obtain

$$\int_{\Gamma} v(\mathbf{x}) dS = \int_{-\varepsilon}^{\varepsilon} \delta_\varepsilon(\eta) \int_{\Gamma_\eta} v(P_\Gamma(\mathbf{x})) J(\mathbf{x}; \eta) dS \, d\eta.$$

Formula (11) then follows from the coarea formula [16] applied to the integral on the right hand side. Because the distance function is used, it is possible to obtain a closed form for the Jacobian $J(\mathbf{x}, \eta)$. See [31] and [32] for proofs.

We can now use this formulation to evaluate integral equations. Consider a general integral equation of the form

$$\int_{\Gamma} \beta(\mathbf{y}) K(\mathbf{x},\mathbf{y}) dS(\mathbf{y}) + \lambda \beta(\mathbf{x}) = f(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Gamma,$$

where $\lambda \in \mathbb{R}$ and Γ is a closed C^2 hypersurface embedded in $\mathbb{R}^m$. We define for any function $u : \Gamma \mapsto \mathbb{R}$ its constant extension along the normals $\bar{u}$ as

$$\forall \mathbf{x} \in \mathbb{R}^m, \qquad \bar{u}(\mathbf{x}) = u(P_\Gamma(\mathbf{x})).$$

Analytically, the formulation in [31] gives us the following

$$\int_{|d_\Gamma| \leq \varepsilon} \bar{\beta}(\mathbf{y}) \tilde{K}(\mathbf{x},\mathbf{y}) \delta_\varepsilon(d_\Gamma(\mathbf{y})) dy + \lambda \bar{\beta}(\mathbf{x}) = \bar{f}(\mathbf{x}),$$

where $\tilde{K}(\mathbf{x},\mathbf{y}) = K(\mathbf{x},\mathbf{y}) J(\mathbf{x};\Gamma)$ can be thought of as a weighted restriction.

On a uniform cartesian grid with grid spacing h, the above expression can be discretized as: $\forall \mathbf{x}_j s.t. |d_\Gamma(\mathbf{x}_j)| < \varepsilon$

$$\sum_{k:|d_\Gamma(\mathbf{x}_k)|<\varepsilon} \bar{\beta}(\mathbf{x}_k) \tilde{K}(\mathbf{x}_j,\mathbf{x}_k) \delta_\varepsilon(d_\Gamma(\mathbf{x}_k)) h^m + \lambda \bar{\beta}(\mathbf{x}_j) = \bar{f}(\mathbf{x}_j). \tag{12}$$

The number of equations in the corresponding system is the number of grid points trapped inside the tubular neighborhood $|d_\Gamma(\mathbf{x})| < \varepsilon$.

5 Algorithm for Mullins-Sekerka dynamics

In this section, we highlight the main ideas of the algorithm and provide a discussion of the interesting and specific features of the algorithm.

5.1 Algorithm

There is variety of work that has been done on the computation of Mullins-Sekerka flows. Zhu etal [46] proposed a boundary integral method for computing the two space dimensional Mullins-Sekerka free boundary problem. In [29], Karali etal used a set of ODE to simulate the interactions of circular bubbles under the Mullins-Sekerka flow both in two and three dimensions. In [9] the authors propose a level set based approach using finite difference to compute the two dimensional Mullins-Sekerka dynamics. Finally, in [5] the authors proposed a parametric finite element approach for two and three dimensions. In this paper, we propose an implicit boundary algorithm for simulating the resulting interfacial motion from Mullins-Sekerka dynamics on unbounded domains in both two and three dimensions. Our algorithm is based on the technique described in [31]. However, because of the complexity of the Mullins-Sekerka dynamics, it is necessary to expand the basic method to more general configurations.

The overall structure of the proposed algorithm is defined by the general level set framework:

1. Given d_{Γ_n} on the grid, solve the Laplace problems (5)-(6). Output: the density β in $\{d_{\Gamma_n}(x_j) < \varepsilon\}$.
2. Compute a suitable extension $\tilde{v}_n$ of v_n in $\{d_{\Gamma_n}(x_j) < \varepsilon\}$.
3. Solve the level set equation in $\{d_{\Gamma_n}(x_j) < \varepsilon\}$ for Δt amount of time

$$\varphi_t + \tilde{v}_n |\nabla \varphi| = 0, \tag{13}$$

   using $\varphi(x,0) = d_{\Gamma_n}$ as initial condition.
4. Redistance: Given $\varphi(x,\Delta t)$, compute the signed distance function d_{Γ_n} to the zero level set of φ.
5. Repeat the above steps until the desired final time.

In the following, we shall describe what we think are important details that are specific to the Mullins-Sekerka simulations, leaving the other steps to the standard level set literature.

5.1.1 Dealing with multiply connected regions in Step 1

One complexity is the fact that the regions can be multiply connected. Numerically, we first need to identify the connected components and how multiply connected they are before solving the correct system. See Figure 1 for an illustration of a typical configuration with simply and multiply connected components. To perform this task, we use a connected component labeling algorithm which was first proposed in the context of computer graphics. See Appendix 7.

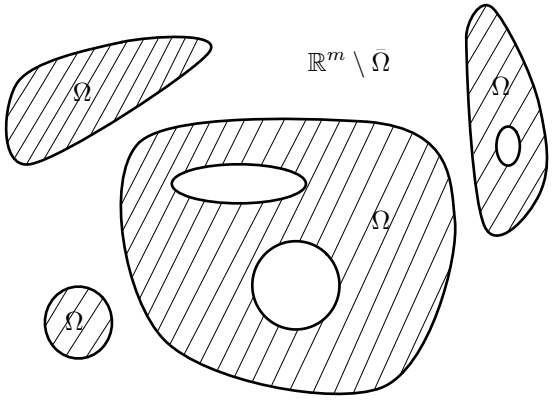

Fig. 1 Non-simply connected domain.

5.1.2 Velocity extension in Step 2

Another difficulty in the context of level set methods, is the velocity extension. In order to propagate an interface moving with normal velocity v_n, it is first necessary to extend $v_n(\mathbf{x},t)$ defined on $\Gamma_t \times [0,\infty)$ to $\tilde{v}_n(\mathbf{x},\tau)$ defined on $\mathbb{R}^m \times [0,\infty)$. This is typically done by solving the system

$$\begin{cases} \frac{\partial \tilde{v}_n}{\partial \tau} + sign(\varphi)\frac{\nabla \varphi}{|\varphi|} \cdot \nabla \tilde{v}_n = 0, \\ \tilde{v}_n(\mathbf{x},0) = v_n(\mathbf{x},t), \end{cases} \tag{14}$$

for a sufficient amount of time so that $\nabla \tilde{v}_n \cdot \nabla \varphi = 0$ in a neighborhood of Γ_t. This extension extends v_n in a constant fashion along streamlines normal to the level sets of φ, which in practice is performed in a neighborhood of the interface, see [10] for a more extensive discussion on this topic. However, this extension requires the knowledge of the normal velocity on the interface which in this case necessitates the evaluation of a hypersingular integral

$$\frac{\partial u}{\partial \mathbf{n}_\mathbf{x}}(P_\Gamma(\mathbf{x})) = \int_\Gamma \frac{\partial^2 \Phi}{\partial_{\mathbf{n}_\mathbf{x}} \partial_{\mathbf{n}_\mathbf{y}}}(P_\Gamma(\mathbf{x}),\mathbf{y})\beta(\mathbf{y})dS(\mathbf{y}).$$

We circumvent this issue by approximating the normal velocity v_n at the grid points $\mathbf{x}$ closest to the interface Γ_t (namely in a very narrow band around the interface taken to be a few grid points wide) using the equation

$$v_n(\mathbf{x}) := sign(d_{\Gamma_t}(\mathbf{x}))\left(\frac{\partial u}{\partial \mathbf{n}^+}(\mathbf{x}) - \frac{\partial u}{\partial \mathbf{n}^+}(\mathbf{x} - 2d_{\Gamma_t}(\mathbf{x})\nabla d_{\Gamma_t}(\mathbf{x}))\right), \tag{15}$$

where the normal derivatives $\frac{\partial u}{\partial \mathbf{n}^+}$ are computed using the interface integral

$$\frac{\partial u}{\partial \mathbf{n}_\mathbf{x}}(\mathbf{x}) = \int_{\Gamma_t} \frac{\partial^2 \Phi}{\partial_{\mathbf{n}_\mathbf{x}} \partial_{\mathbf{n}_\mathbf{y}}}(\mathbf{x},\mathbf{y})\beta(\mathbf{y})dS(\mathbf{y}),$$

where $\mathbf{x} \notin \Gamma_t$. The idea behind equation (15) is to compute the jump in the normal derivative of u at a point $\mathbf{x}$ close to Γ_t by using it symmetrical point (or mirror point) $\mathbf{x} - 2d_{\Gamma_t}(\mathbf{x})\nabla d_{\Gamma_t}(\mathbf{x})$ with respect to Γ_t. This is illustrated in Figure 2.

Note that the extension $\tilde{v}_n$ in (15) is continuous and agrees with v_n on Γ_t. We point out that this step can be interpreted as the initialization step for the PDE in equation (14). Thus, once the normal velocity $\tilde{v}_n$ is obtained in this very narrow band, we extend it to the next layer of grid points using the standard extension (14).

Finally the normal velocity may tend to ∞ as parts of Γ_t vanish in finite time. Numerically, we regularize the computed normal velocity imposing a maximum normal velocity v_∞. This means we replace v_n by $\max(v_n, v_\infty)$.

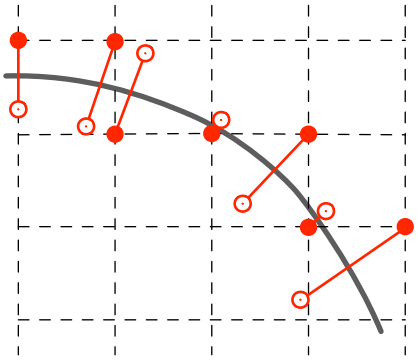

Fig. 2 Velocity extension. The solid dots represent the grid points $\mathbf{x}$ and the empty ones their corresponding mirror points across the interface.

5.1.3 Literature on redistancing and involving the level set equations

To maintain the completeness of this exposition, we briefly mention some established literature on the level set equation (13) and the construction of signed distance functions (Step 4 in the algorithm).

The level set equation (13), first proposed by Osher and Sethian in their seminal paper [40], convects the values of the level set function φ with the velocity field $\tilde{v}_n$ which has been extended off of Γ_t. Typically, the extension is performed in a neighborhood of the interface since we are only interested in the location of the interface. Such techniques are referred to as local level set methods[1, 41].

The construction of signed distance functions (or distance reinitialization) is a procedure that replaces a general level set function by the signed distance function $d(\mathbf{x},t)$ which is the value of the distance from $\mathbf{x}$ to Γ_t taken to be positive inside and negative outside (or vice versa). This assures that φ does not become too flat nor too steep near the interface. To construct the signed distance function, one needs to find a function d such that

$$|\nabla d| = 1 \text{ almost everywhere,} \qquad \text{subject to } \{\mathbf{x} : \varphi(\mathbf{x},t) = 0\} = \{\mathbf{x} : d(\mathbf{x},t) = 0\}.$$

Such construction can be performed very efficiently using fast sweeping or fast marching algorithms. See e.g. [10, 42, 43, 44, 45].

6 Numerical simulations

In this section, we show the results of some numerical simulations using the algorithm. The aim of the following simulations is to show the computational properties of the algorithm. We discretize the integral equations (7), (7) and (9) using finite differences and compute them using simple Riemann sums over uniform grids. Af-

ter discretization, we obtain a linear system of the form $Ax = b$ where the vector x consists of the unknown density β and the constants A_i. See (7), (8) and (9).

To solve equation (14), we use an upwind scheme with a WENO-3 scheme for the spatial derivatives, and third order TVD-RK3 method for time. We propagate enough to ensure the velocity information is transmitted throughout an ε wide band of the boundary Γ_t. For the level set equation, we use Godunov scheme with WENO-3 discretization for the eikonal term, and TVD-RK3 method for time.

6.1 Two dimensions

We start by illustrating that the simulated two dimensional Mullins-Sekerka does indeed preserve area and decreases length. In this test case, we computed the evolution of the perimeter and area of an ellipse under the Mullins-Sekerka dynamics. The plots for the perimeter and area versus time are shown in Figure 3.

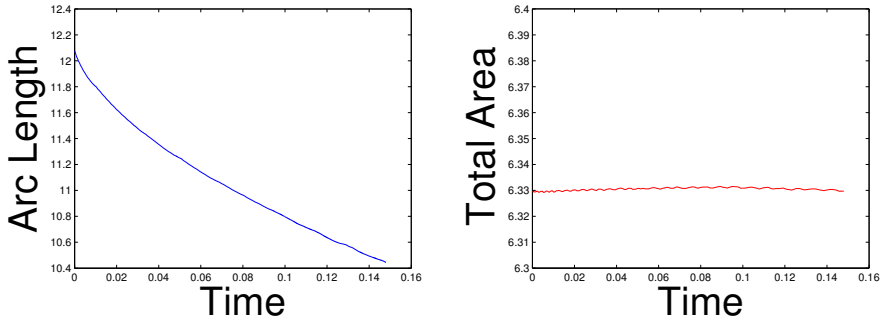

Fig. 3 Evolution of the perimeter and area of an ellipse under the Mullins-Sekerka model.

As a second test problem, we simulated the evolution of an elongated tube to corroborate the fact that the two dimensional Mullins-Sekerka does not preserve convexity at all times during the evolution. This is illustrated in Figure 4.

Finally, we demonstrate the capability of our algorithm to handle mergings of connected components. In this particular test example, we started with two ellipses that were close enough to finally merge into one connected component. Figure 5 shows the evolution of these two ellipses and Table 1 shows the numerical errors in the conservation of total area before and after merging.

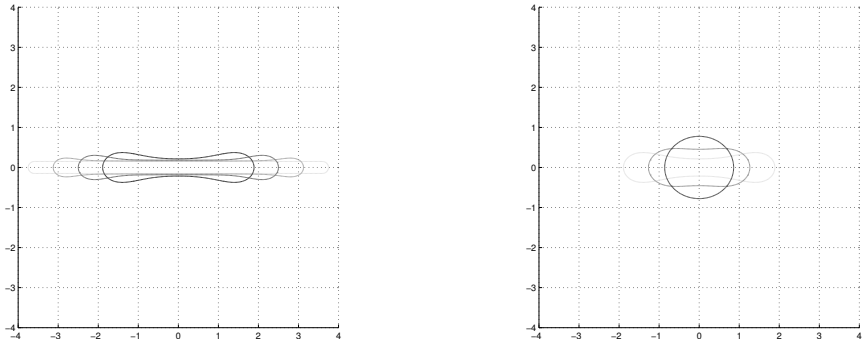

Fig. 4 Evolution of a thin tube under the two dimensional Mullins-Sekerka dynamics.

h	4/128	4/256	4/512
Initial Area	2.6965	2.69611	2.695495
Start Merge Area	2.696655	2.691792	2.686751
End Merge Area	2.804205	2.724255	2.694007
Area Jump	0.107550	0.032463	0.007256
Relative Area Error	0.03988	0.01206	0.00270

Table 1 Merging time and area jump for the merging of two ellipses evolving under the two-dimensional Mullins-Sekerka dynamics.

6.2 Three dimensions

In these simulations, we assume that the far-field condition is $\lim_{|\mathbf{x}|\to\infty}|u(\mathbf{x})| = u_\infty$ which allows the solution to converge to a non-zero constant. This u_∞ is referred to as the far-field environment which corresponds to the temperature at infinity. The dynamics of the Mullins-Sekerka model depend on this far-field environment and give rise to interesting behaviors that are illustrated in these simulations. While it is difficult to obtain rigorous analytical results on the general 3D Mullins-Sekerka model on unbounded domains with a general far-field environment u_∞, we can simulate these different situations. We note that to obtain the so-called Mullins-Sekerka instability (or dendritic growth) it is necessary that $u_\infty < 0$.

We observe that in section 2, we only provided a method for solving the three dimensional exterior problem with $u_\infty = 0$. We thus briefly provide the equations for solving the **general 3D exterior problem**:

1. Find the density β and the constants $\{A_i\}_{i=1}^L$ such that

$$\begin{cases} \int_\Gamma \beta(\mathbf{y})\left(\frac{\partial\Phi(\mathbf{x},\mathbf{y})}{\partial\mathbf{n}_\mathbf{y}^+} - \frac{1}{|\mathbf{x}-\mathbf{y}|^{m-2}}\right)dS(\mathbf{y}) - \frac{1}{2}\beta(\mathbf{x}) + \sum_{i=1}^L A_i\Phi(\mathbf{x}-\mathbf{z_i}) = f(\mathbf{x}) - u_\infty, \\ \int_{\Gamma_i}\beta(\mathbf{y})dS(\mathbf{y}) = 0, \quad 1\le i\le L, \ \ \mathbf{x}\in\Gamma. \end{cases}$$

2. Reconstruct the solution u in $\mathbb{R}^m\setminus\bar{\Omega}$ using the double layer potential formulation

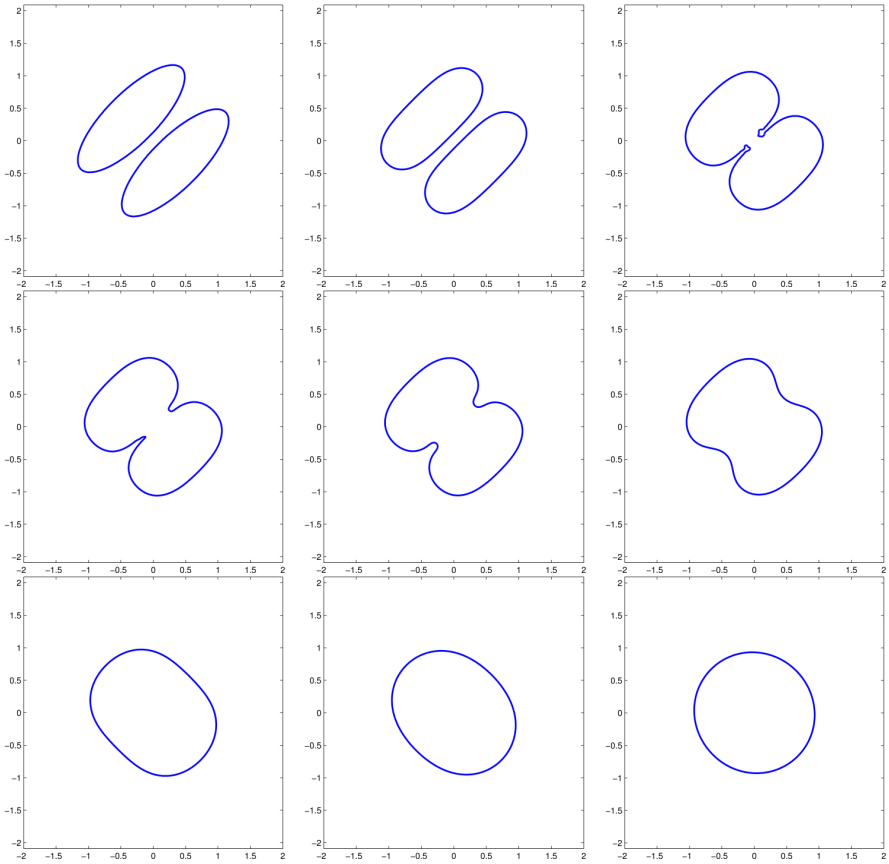

Fig. 5 Two ellipses merging in a two-dimensional Mullins-Sekerka flow simulation.

$$u(\mathbf{x}) = \int_{\Gamma} \beta(\mathbf{y}) \frac{\partial \Phi(\mathbf{x},\mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} dS(\mathbf{y}) + \sum_{i=1}^{L} A_i \Phi(\mathbf{x} - \mathbf{z}_{\mathbf{i}}) + u_{\infty}, \qquad \text{for } \mathbf{x} \in R^m \setminus \bar{\Omega}.$$

We focus on two features of the three dimensional Mullins-Sekerka dynamics. First, we look at the isothermal process and show that it depends on the far-field environment u_∞. Second, we simulate a case where instabilities develop. See Figure 7.

In Figure 6 we simulate the dynamics of two initial spheres with different radii using various far-field value u_∞. If u_∞ is small (i.e. low temperature), both spheres grow. If u_∞ is large (high temperature), both spheres shrink (i.e. melt). If u_∞ is set at phase transition point, we see that the larger sphere grows at the expense of the smaller one, as expected in the two dimensional case.

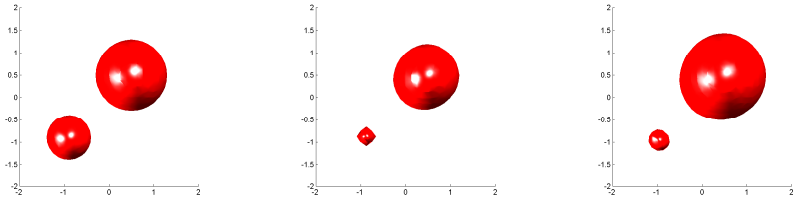

Fig. 6 Effect of the far-field value u_∞ on the dynamics of two spheres. On the left plot, both spheres grow in a low temperature environment. In the middle, both shrink in a high temperature environment. On the right, the larger sphere grows and absorbes the mass of the smaller sphere.

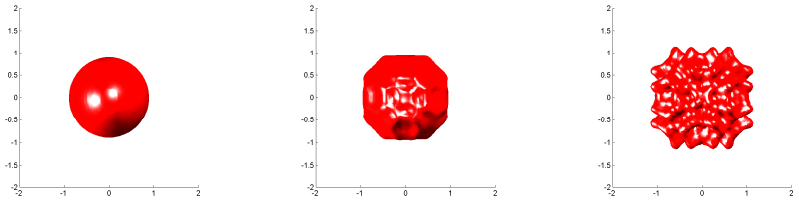

Fig. 7 Instability of dendritic growth in the 3D Mullins-Sekerka while maintaining crystal symmetry.

7 Conclusion

We described an implicit boundary integral algorithm for computing the dynamics of the Mullins-Sekerka model on unbounded domains in two and three dimensions. We used a formulation described in [31] and further investigated in [32] to compute integral equations using a signed distance function. We showed that our algorithm was able to handle topological changes, such as mergings of connected components, and was able to simulate the so-called Mullins-Sekerka instability (or dentritic growth).

This paper demonstrates the versatility our computational method and in particular its capability to simulate on unbounded domains in three dimensions. Coupled with fast multipole methods, this algorithm has the potential to be a powerful tool to better understand the dynamics of the Mullins-Sekerka model on unbounded domains.

Acknowledgements Kublik's research was partially funded by a University of Dayton Research Council Seed Grant and a Dr. Schraut Faculty Research Award. Chen's and Tsai's research is partially supported by Simons Foundation, NSF Grants DMS-1318975, DMS-1217203, and ARO Grant No. W911NF-12-1-0519.

Appendix

Connected component labeling

As explained in section 2, the system of equations to solve depends on many properties including the number of connected components of the region, whether the region is bounded or not (exterior vs interior), and the orientation of the region. In our numerical simulations, we adapted a technique called connected component labeling (CCL), see e.g. [14], to find necessary topological information needed to obtain the correct set of equations. These include:

1. The boundedness of the connected component C_i. This decides which formulation (interior vs exterior) to use.
2. The orientation of C_i which determines the normal direction.
3. The total number of connected boundary components of the boundary of C_i, denoted by $\Gamma^i = \partial C_i$.
4. The selection of $\mathbf{z}_i$ in (7), (8) and (9) for each component Γ^j. This involves
   - the separation of Γ^i into boundary components Γ^i_j, each bounding a hole in the region except for the most exterior one. We denote the exterior boundary as Γ^i_0.
   - For each hole delimited by Γ^i_j , $j \neq 0$, find a point $\mathbf{z}_i$ inside the hole that gives the least singular value of $\Phi(\mathbf{x} - \mathbf{z}_i)$. This means that $\mathbf{z_i}$ should be as far from the boundary as possible.

Since we deal with closed interfaces, every grid point belongs to a unique connected component and has a well defined component label. Note that a connected component C_i may belong to Ω or $\mathbf{R}^m \setminus \Omega$.

We use the algorithm as follows:

- We label the unbounded component as C_0. This is the only true exterior component for the boundary integral formulations.
- Each interior component ($d_\Gamma > 0$, i.e. the solid phase) will have positive label i and each exterior component ($d_\Gamma < 0$ i.e. the liquid phase) will take negative label $-i$ (except for the unbounded one).
- Each boundary piece of each component will have label j.
- The point $\mathbf{z_i}$ in C_i is chosen to be the point with largest signed distance function in absolute value.

We adopt the two-pass CCL algorithm with $2m$-connectivity in $\mathbb{R}^m$. This algorithm uses equivalence classes for labels: after the first pass, points in the same connected component may not have the same label, but the labels of points in the same connected component will be assigned to the same equivalent class in the second pass. The root of the equivalence class denotes the smallest label (in absolute value) in the equivalence class. The scanning process works as follows:

1. First Pass

a. With begin with label 0 which denotes the unbounded component C_0.
b. At the current point scanned, we look at the sign of the signed distance function of its $2m$ neighbors already visited (in 2D, west and north of the current point)
   - If none of the neighbors have the same sign as the current point, we create a new label (positive or negative, based on the sign of the distance function at the current point) and a new equivalence class.
   - If only one neighbor has the same sign, we pick the label for the current point to be the root of that neighboring point's equivalence class.
   - If more than one neighbor has the same sign as the current point, we pick the label for the current point to be the smallest root of the neighbors that have the same sign's equivalence classes. Furthermore, we combine the equivalence classes of the neighboring points that have the same sign since they are connected through the current point.
c. The largest root of all equivalence classes with a given sign (interior or exterior) will give the total number of connected components that have that sign. Thus, the sum is the total number of connected components.

2. Second Pass

a. At each point, we assign its label to be the root of its equivalence class.
b. We update and store the points with largest absolute distance within each equivalence class. These are the points $\mathbf{z_i}$.
c. For each point $\mathbf{x}$ within the ε tubular neighborhood of the boundary ($|d_\Gamma| < \varepsilon$), we look for the root of its equivalence class (say i) and the root of its projection point's ($P_\Gamma(\mathbf{x})$) equivalence class (say j) that takes the opposite sign. To obtain j, we look at the vertices of the cell $P_\Gamma(\mathbf{x})$ falls in and scan for the label with opposite sign. This step identifies the boundary piece Γ_j^i and collects points within the ε neighborhood of the boundary Γ_j^i.
d. We store the total number of connected boundary pieces.

Compatibility conditions

Consider Ω an open bounded domain in $\mathbb{R}^m$ with smooth boundary $\partial\Omega = \Gamma$ for $m = 2,3$. Let $R > 0$ be sufficiently large so that $\bar{\Omega} \subset B_R(x)$ where $B_R(x)$ is the ball centered at $x \in \Omega$ with radius R. Define the interior problem

$$\begin{cases} \Delta u = 0 & \text{in } \Omega \\ u = f & \text{on } \Gamma, \end{cases}$$

and the exterior problem

$$\begin{cases} \Delta v = 0 & \text{in } B_R(x) \setminus \bar{\Omega} \\ v = u & \text{on } \Gamma \\ \text{"far field condition"}. \end{cases}$$

We discuss what "far field condition" should be in the following. Using Green's identity in $B_R(x) \setminus \bar{\Omega}$ for $x \in \Omega$, we obtain

$$\begin{aligned} 0 &= \int_{B_R(x)\setminus\bar{\Omega}} (v(y)\Delta_y \Phi(x,y) - \Phi(x,y)\Delta_y v(y))\, dy \\ &= \int_\Gamma \left(v(y)\frac{\partial \Phi(x,y)}{\partial \mathbf{n}_y} - \Phi(x,y)\frac{\partial v(y)}{\partial \mathbf{n}_y} \right) dS(y) \\ &\quad - \int_{\partial B_R(x)} \left(v(y)\frac{\partial \Phi(x,y)}{\partial \mathbf{n}_y} - \Phi(x,y)\frac{\partial v(y)}{\partial \mathbf{n}_y} \right) dS(y). \end{aligned}$$

Thus

$$\int_\Gamma \left(v(y)\frac{\partial \Phi(x,y)}{\partial \mathbf{n}_y} - \Phi(x,y)\frac{\partial v(y)}{\partial \mathbf{n}_y} \right) dS(y) = \frac{\partial \Phi(R)}{\partial r}\int_{\partial B_R(x)} v ds - \Phi(R)\int_{\partial B_R(x)} \frac{\partial v}{\partial r} ds \tag{16}$$

with $\Phi(x,y) = \Phi(|x-y|) = \Phi(r)$. In addition, notice that since v is harmonic in $B_R(x) \setminus \bar{\Omega}$,

$$\int_\Gamma \left(\frac{\partial v}{\partial n} \right)_+ ds = \int_{\partial B_R(x)} \frac{\partial v}{\partial r} ds.$$

Now, since formula (16) holds for any $R > 0$, we see that necessarily for $m = 2$

$$\int_\Gamma \left(\frac{\partial v}{\partial n} \right)_+ ds = 0, \tag{17}$$

otherwise the second integral on the right hand side will not be bounded uniformly in R. For $m = 3$, it suffices that the integral $\int_\Gamma \left(\frac{\partial v}{\partial n} \right)_+ ds$ be bounded uniformly in R, since $\Phi(R) \sim O(1/R)$.

Consequently "the far field condition" should be

$$m = 2: \quad \lim_{R\to\infty} |v| < \infty,$$

since $\frac{\partial \Phi(r)}{\partial r}|_{r=R}$ is $O(R^{-1})$ and $ds \sim O(R)$, and

$$m = 3: \quad \lim_{R\to 0} v = 0,$$

since $\Phi(R) \sim O(R^{-1})$ and $ds \sim O(R^2)$.

Now applying the divergence theorem in the "annulus" we obtain

$$\int_{\partial B_R(x)} v\nabla v\cdot\mathbf{n}^+ ds - \int_\Gamma v\nabla v\cdot\mathbf{n}^+ ds = \int_{B_R(x)\setminus\bar{\Omega}} \nabla\cdot(v\nabla v)\,dx \tag{18}$$

$$= \int_{B_R(x)\setminus\bar{\Omega}} |\nabla v|^2 dx + \int_{B_R(x)\setminus\bar{\Omega}} \Delta v dx \tag{19}$$

$$= \int_{B_R(x)\setminus\bar{\Omega}} |\nabla v|^2 dx \tag{20}$$

since v is harmonic in $\mathbb{R}^m\setminus\bar{\Omega}$. Now in two dimensions and by the previous argument, we know that $\lim_{R\to\infty}\frac{\partial v}{\partial n}|_{\partial B_R(x)} = 0$ which implies that for $m = 2$

$$\lim_{R\to\infty}\int_{\partial B_R(x)} v\nabla v\cdot\mathbf{n}^+ ds = \lim_{R\to\infty}\int_{\partial B_R(x)} v\frac{\partial v}{\partial \mathbf{n}^+} ds = 0.$$

Thus, taking R to infinity in equation (20), we obtain

$$\int_\Gamma v\nabla v\cdot\mathbf{n}^+ ds = -\int_{\mathbb{R}^2\setminus\bar{\Omega}} |\nabla v|^2 dx. \tag{21}$$

References

1. D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *J. of Comput. Phys.*, 118(2):269–277, 1995.
2. K. E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1997.
3. K. E. Atkinson and G. Chandler. Boundary integral equation methods for solving laplace's equation with nonlinear boundary conditions: the smooth boundary case. *Mathematics of Computation*, 55(191):451–472, 1990.
4. I. Babuška. The finite element method for elliptic equations with discontinuous coefficients. *Computing*, 5:207–213, 1970.
5. J. W. Barrett, H. Garcke, and R. Nrnberg. On stable parametric finite element methods for the stefan problem and the mullinssekerka problem with applications to dendritic growth. *Journal of Computational Physics*, 229(18):6270 – 6299, 2010.
6. P. W. Bates, X. Chen, and X. Deng. A numerical scheme for the two phase Mullins-Sekerka problem. *Electronic Journal of Differential Equations*, 1995.
7. J. Bedrossian, J. J. von Brecht, S. Zhu, E. Sifakis, and J. Teran. A second order virtual node method for elliptic problems with interfaces and irregular domains. *J. Comput. Phys.*, 229:6405–6426, 2010.
8. S. Börn, L. Grasedyck, and W. Hackbusch. Hierarchical matrices. Technical report, Max-Planck Institut fur Mathematik in den Naturwissenschaften, Leipzig, 2003.
9. S. Chen, B. Merriman, S. Osher, and P. Smereka. A simple level set method for solving Stefan problem. *J. Comput. Phys.*, 135, 1997.
10. L.-T. Cheng and Y.-H. Tsai. Redistancing by flow time dependent Eikonal equation. *J. Comput. Phys.*, 227(2):4002–4017, 2008.
11. I.-L. Chern and Y.-C. Shu. A coupling interface method for elliptic interface problems. *J. of Comput. Physics*, 225:2138–2174, 2007.
12. P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. SIAM, 1978.
13. S. Conti, B. Niethammer, and F. Otto. Coarsening rates in off-critical mixtures. *SIAM J. Math. Anal.*, 37(6):1732–1741, 2006.

14. M. Dillencourt, H. Samet, and M. Tamminen. A general approach to connected-component labeling for arbitrary image representations. *J. ACM*, 39, 1992.
15. J. Dolbow and I. Harari. An efficient finite element method for embedded interface problems. *J. Numer. Methods Eng.*, 78:229–252, 2009.
16. H. Federer. *Geometric Measure Theory*. Springer-Verlag, 1969.
17. G. B. Folland. *Introduction to Partial Differential Equations*. Princeton University Press, 1976.
18. F. Gibou and R. Fedkiw. A fourth order accurate discretization for the laplace and heat equations on arbitrary domains, with applications to the stefan problem. *J. Comput. Phys.*, 202:577–601, 2005.
19. F. Gibou, R. Fedkiw, L. Cheng, and M. Kang. A second order accurate symmetric discretization of the poisson equation on irregular domains. *J. Comput. Phys.*, 176:1–23, 2002.
20. Y. Giga. *Surface evolution equations: A level set approach*, volume 99 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 2006.
21. A. Greenbaum, L. Greengard, and G. B. MFadden. Laplace's equation and the Dirichlet-Neumann map in multiply connected domains. *J. Comput. Phys.*, 105:267–278, 1993.
22. L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.
23. M. Gurtin. On the two-phase stefan problem with interfacial energy and entropy. *Archive for Rational Mechanics and Analysis*, 96, 1986.
24. A. Hansbo and P. Hansbo. An unfitted element method, based on nitsche's method for elliptic interface problems. *Comput. Methods Appl. Mech. Eng.*, 191:5537–5552, 2002.
25. A. Hansbo and P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Comput. Methods Appl. Mech. Eng.*, 193:3523–3540, 2004.
26. J. Huang and J. Zou. A mortar element method for elliptic problems with discontinous coefficients. *IMA J. Numer. Anal.*, 22:549–576, 2002.
27. H. Johansen. *Cartesian grid embedded boundary finite difference methods for elliptic and parabolic differential equations on irregular domains*. PhD thesis, University of California, Berkeley, 1997.
28. H. Johansen and P. Colella. A cartesian grid embedded boundary method for poisson's equation on irregular domains. *J. Comput. Phys.*, 147:60–85, 1998.
29. G. Karali and P. Kevrekidis. Bubble interactions for the Mullins-Sekerka problem: some case examples. *Math. Comput. Simul.*, 80, 2009.
30. R. Kress. *Linear Integral Equations*. Springer-Verlag, New York, second edition, 1999.
31. C. Kublik, N. M. Tanushev, and R. Tsai. An Implicit Interface Boundary Integral Method for Poisson's Equation on Arbitrary Domains. *J. Comput. Phys.*, 247:269–311, 2013.
32. C. Kublik and R. Tsai. Integration over curves and surfaces defined by the closest point mapping. *Research in the Mathematical Sciences*, 2016.
33. R. Leveque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.
34. Z. Li and K. Ito. The immersed interface method: numerical solutions of pdes involving interfaces and irregular domains (frontiers in applied mathematics). *Society for Industrial and Applied Mathematics*, 2006.
35. X. Liu, R. Fedkiw, and M. Kang. A boundary condition capturing method for poisson's equation on irregular domains. *J. Comput. Phys.*, 160(1):151–178, 2000.
36. S. G. Mikhlin. *Integral Equations*. Pergammon, London, 1957.
37. T.-H. Miura. Zero width limit of the heat equation on moving thin domains. *UTMS Preprint Series, The University of Tokyo*, 2015.
38. W. W. Mullins and R. F. Sekerka. Morphological stability of a particle growing by diffusion or heat flow. *J. of Applied Physics*, 34:323–329, 1963.
39. E. Nyström. Über die praktische Auflösung von Integralgleichungen mit Anwendungen auf Randwertaufgaben. *Acta Math.*, 54:185–204, 1930.
40. S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *J. Comp. Phys.*, 79:12–49, 1988.

41. D. Peng, B. Merriman, S. Osher, H.-K. Zhao, and M. Kang. A pde-based fast local level set method. *J. Comput. Phys.*, 155(2):410–438, 1999.
42. G. Russo and P. Smereka. A remark on computing distance functions. *J. Comput. Phys.*, 163:51–67, 2000.
43. J. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
44. Y.-H. Tsai, L. Cheng, S. Osher, and H.-K. Zhao. Fast sweeping methods for a class of hamilton-jacobi equations. *SIAM Journal on Numerical Analysis*, 41(2):673–694, 2003.
45. J. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40:1528–1538, 1995.
46. J. Zhu, X. Chen, and T. Y. Hou. An efficient boundary integral method for the Mullins-Sekerka problem. *J. Comput. Phys.*, 127:246–267, 1996.