

Article

Isogeometric Analysis-Based Solidification Simulation and an Improved Way to Apply Supercooling on Droplet Boundary

Jiangping Xu, Tingyu Yan, Yang Li, Zhenyuan Yu, Yun Wang and Yuan Wang *

School of Mechanical Engineering, Jiangsu University, 301 Xuefu Road, Zhenjiang 212013, China

* Correspondence: wangyuan@ujs.edu.cn

Abstract: The classic finite difference method (FDM) has been successfully adopted in the simulation of dendritic solidification, which is based on phase-field theory. Nevertheless, special strategies of boundary integral and projection are required for applying a supercooling rate to a droplet surface. In the present study, isogeometric analysis (IGA) is employed to discretize the phase-field equation due to the two advantages of Non-Uniform Rational B-Splines (NURBS) basis functions, namely an arbitrary order of derivatives and exact description of complex geometry. In addition, an improved, easy way to apply the supercooling rate on a melt droplet surface is proposed to avoid the integral and projection of the cellular boundary required in FDM. Firstly, dendrite growth in a square computational domain is simulated to verify the performance of IGA. Then, the influences of latent heat, anisotropic mode and initial angle on the dendrite shapes are studied by the presented IGA, FDM and finite element method (FEM). Finally, dendritic solidification in a droplet under different cooling rates along irregular boundaries is performed by the proposed IGA.

Keywords: dendrite growth; isogeometric analysis; mathematical model; phase field



Citation: Xu, J.; Yan, T.; Li, Y.; Yu, Z.; Wang, Y.; Wang, Y. Isogeometric Analysis-Based Solidification Simulation and an Improved Way to Apply Supercooling on Droplet Boundary. *Metals* **2022**, *12*, 1836. <https://doi.org/10.3390/met12111836>

Academic Editor: Zhaoyang Hou

Received: 12 October 2022

Accepted: 24 October 2022

Published: 28 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Dendrite growth is an essential phenomenon of phase transition in material science [1,2]. In the casting of metals and alloys, molten metals first generate tiny embryos in the liquid phase and then gradually grow into a stable crystal nucleus. Under the effect of supercooling, they grow up continuously into different structures and finally complete the process of solidification [3]. The final morphology of crystal growth has a direct impact on the properties of materials, such as mechanical strength, corrosives, electrical conductivity and ductility [4]. Numerical technology could provide the ability to gain insight into the interior quality analysis of casting products. It assists engineers in observing the evolution of microstructures in the metal solidification process under different conditions and parameters and thus improves product quality. Furthermore, it helps to control the dendrite growth to obtain the required material properties. Based on the finite difference method (FDM), a special way to calculate the boundary cooling term for a droplet surface was proposed by Miura et al. [5] to avoid overestimated surface length, as shown in Figure 1. Inspired by their work to study the effects of supercooling rates on solidification in droplets, we conducted similar research via IGA with an improved reaction equation for temperature dynamics by introducing an extra term for supercooling on the droplet boundary. Thus, additional calculations of projection lengths of cellular boundary edges and integrals over surfaces are avoided.

Up to now, the phase field method has been commonly applied in material science [6,7], crystal nucleation and growth [8–10], fracture mechanics [11,12], heat transfer [13–17], fluid-structure interaction [18,19] and biomechanics [20–28]. Taking the example in solidification of pure metal, there are two phases in the whole studied domain, namely liquid and solid phases whose order parameters ϕ take values of 0 and 1, respectively. The value of the interface phase between these two phases varies in the interval (0, 1). The dynamic of the

interface is driven by the free energy difference between the two phases [29]. As shown in Figure 2, the non-conservative phase field ϕ has two local stable phases with $\phi = 0$ and $\phi = 1$, respectively. However, the interface phase $0 < \phi < 1$ is meta-stable and has a tendency to jump from high energy to low energy. As a consequence, the interface moves from one phase to the other, and phase transition is achieved automatically without tracking the interface front. Nevertheless, the phase field equation involves the computation of high-order derivatives with respect to order parameter. Usually, the high-order finite element method (FEM) and finite difference method [30–33] are used to solve the high-order partial differential equation. The former brings about more computational efforts due to extra freedoms. The latter is particularly limited to the regular domain. In order to solve the above problems, IGA is adopted to simulate the dendrite growth in this study [34].

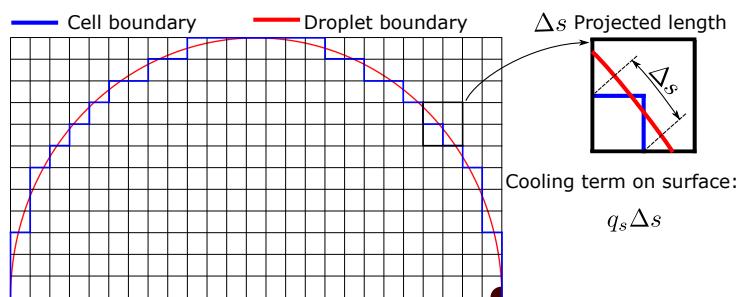


Figure 1. Special strategy for applying supercooling boundary conditions on droplet surface in FDM [5]. Δs is the projected effective length, and q_s is the normal component to the droplet boundary.

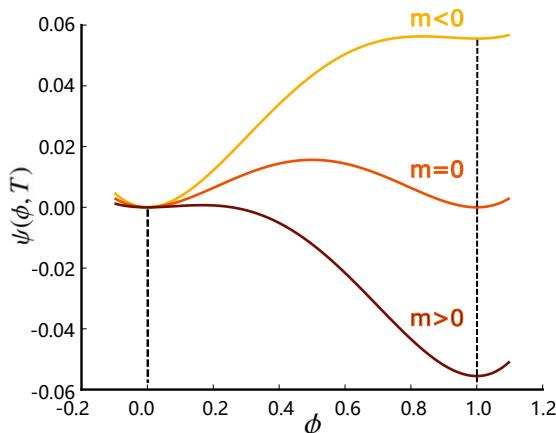


Figure 2. Chemical free energy of the phase-field equation for solidification. When temperature is less than melting temperature, namely $m > 0$, the interfacial phase tends to be a solid phase, which suggests dendrite growth. The opposite situation can be concluded if $m < 0$. In the case of $m = 0$, the interface might not move. Definition of element domains in IGA. Physical elemental domain Ω^e (Top), elemental domain $\hat{\Omega}^e$ in parametric space (bottom left) and Gaussian integration domain $\bar{\Omega}^e$ (bottom right).

Isogeometric analysis was first proposed by Hughes [35] in 2005 for the seamless integration of computer-aided design (CAD) and computer-aided engineering (CAE) and has been widely used in crack [36–38], vibration [39–41], heat transfer and fluid–structure interaction [18,42–44], and optimization of practical engineering problems [45–48]. It can not only accurately describe the complex geometric shape of the object but also has an arbitrary-order partial derivative in theory due to the use of NURBS basis functions and has been proven very successful for phase-field equations [43,44,49–51]. In the work of reference [50], the application of IGA in phase-field is reviewed and a general example of solidification is given. In the present work, the computational framework of IGA is proposed to analyze solidification and study the effects of several parameters on dendrite growth.

In addition, a special strategy for boundary supercooling is presented, and corresponding findings with different cooling rates are compared with experimental observations.

This paper is organized as follows: Section 2 describes the phase-field based mathematical model for dendrite growth and the reaction-diffusion equation for temperature dynamics. The NURBS basis functions of IGA, their derivatives and the integration scheme are introduced in Section 3. Section 4 presents the IGA-based computational framework of the phase-field and reaction-diffusion equation of temperature dynamics for solidification simulation. Then, the process of the generalized- α method for solving the equations is illustrated. In Section 5, a square domain is used to test the performance of the presented method for dendrite growth, and the effects of several parameters on crystal growth are discussed. The computational efficiency of the proposed method is then compared with FDM and FEM. Moreover, an extra term is added to the dynamic equation of temperature to study the effects of supercooling rate on solidification. Three values of cooling rates are tested, and the results are validated with the work of references [5,52]. Finally, we conclude with a summary in Section 6.

2. Mathematical Model

In this study, the phase-field equation is applied to describe the process of dendrite growth, and a reaction–diffusion equation is adopted for the dynamics of temperature. The variable $\phi(x, y, t)$ is taken as the order parameter in the phase-field model, representing the solid phase with $\phi(x, y, t) = 1$ and liquid phase with $\phi(x, y, t) = 0$ at position (x, y) at time t . There is automatically a smoothing interface $0 < \phi(x, y, t) < 1$ between the two phases. For the purpose of convenience, $\phi(x, y, t)$ is simply written as ϕ . The free energy functional for driving the phase transition of the system is defined as

$$F(\phi, m) = \int_{\Omega} \left[\frac{1}{2} \varepsilon^2 |\nabla \phi|^2 + \psi(\phi, T) \right] d\Omega \quad (1)$$

where ε is a constant related to the interface width. The first term in the integral of Equation (1) is the interface free energy and the second term $\psi(\phi, T)$ is the chemical free energy, a double-well potential with two local minima at $\phi = 0$ and $\phi = 1$. The specific expression of $\psi(\phi, T)$ is chosen as

$$\psi(\phi, T) = \frac{1}{4} \phi^4 - \left(\frac{1}{2} - \frac{1}{3} m \right) \phi^3 + \left(\frac{1}{4} - \frac{1}{2} m \right) \phi^2 \quad (2)$$

where m is a function of supercooling, acting as a thermodynamic driving force, and is defined as

$$m(T) = \frac{\alpha}{\pi} \arctan(\gamma(T_{eq} - T)) \quad (3)$$

where T_{eq} is the melting temperature, taking the value of 1 in this paper. Temperature field T is normalized to the interval $[0, 1]$. α and γ are constants. As shown in Figure 2, the $\psi(\phi, T)$ has two local minima at $\phi = 0$ and $\phi = 1$. The dynamic system tends to be stable, which leads to a decrease in energy. The metastable state of phase at the interface turns out to be the local stable phase. When the temperature T at interfacial phase $0 < \phi < 1$ is less than the balance temperature T_{eq} , namely $m > 0$, the interface phase will be driven to be solid phase $\phi = 1$ since the interface phase changes from a high energy state to a low energy state. If the temperature T is greater than T_{eq} , the opposite situation happens, and the liquid phase grows.

In practice, the morphology of dendrite growth is anisotropic. In order to capture the anisotropic characteristic of crystal, the function of interfacial width ε is introduced as follows

$$\varepsilon(\theta) = \bar{\varepsilon}[1 + \delta \cos(j(\theta - \theta_0))] \quad (4)$$

where $\bar{\varepsilon}$ is the phase width and determines the thickness of the interface layer; δ is the strength of interface width; j is the mode number of anisotropy; θ is the angle between

the outward normal direction of the interface and the x axis; θ_0 is the initial angle that represents the orientation of the anisotropic axis.

The free energy of the system Equation (1) is required to decrease monotonically with time. Considering the concept of gradient dynamics [53], the non-conserved phase-field equation for solidification is defined as

$$\tau \frac{\partial \phi}{\partial t} = - \frac{\delta F}{\delta \phi} \quad (5)$$

Substituting Equations (2)–(4) into Equation (1), then, Equation (5) can be derived as

$$\tau \frac{\partial \phi}{\partial t} = - \frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi}{\partial y} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi}{\partial x} \right) + \nabla \cdot \varepsilon^2 \nabla \phi - \mu(\phi, T) \quad (6)$$

where $\varepsilon' = d\varepsilon/d\theta$; $\mu(\phi, T)$ is the derivative of chemical free energy with respect to ϕ and given by

$$\mu(\phi, T) = \phi^3 - (1.5 - m)\phi^2 + (0.5 - m)\phi = -\phi(1 - \phi)(\phi - 0.5 + m) \quad (7)$$

According to references [54,55], the dynamics of the temperature field are described by a reaction–diffusion equation can be rewritten as follows

$$\frac{\partial T}{\partial t} = \nabla \cdot \nabla T + K \frac{\partial \phi}{\partial t} \quad (8)$$

where K is the dimensionless latent heat. The second term on the right side of Equation (8) accounts for the release of latent heat at the interface. Equations (6) and (8) are in dimensionless units. The modified equation of temperature dynamic for applying supercooling on a droplet surface is presented in Section 5.2.

3. NURBS Shape Functions in Isogeometric Analysis

As an extension of FEM, IGA aims to accomplish the seamless integration of CAD and CAE [35]. In the implementation of IGA, the NURBS basis functions are used instead of the classical Lagrangian shape functions in FEM. The unknowns to be solved are the physical variables at control points, no longer at nodes of elements. In this section, the NURBS bases and numerical integration are introduced.

3.1. B-Splines

B-spline is a basis spline function that has local support with a given degree, knot partition and control points. The knot vector in a B-spline is defined as $U = \{u_1, u_2, \dots, u_{n+p+1}\}$ in a non-descending order. u_i is the i th knot, n is the number of basis functions and p is the polynomial order [35]. The knot interval could be uniform, non-uniform, or even 0 where more than one knot repeats at the same partition position i . According to the Cox-de Boor recursion formula [56], the B-Spline basis functions are given for a polynomial order $p = 0$

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

and for a polynomial order $p \geq 1$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (10)$$

where $N_{i,p}(u)$ is i th B-spline basis function of order p . The ratio in the form of 0/0 is defined as zero in the process of calculating the bases. Some important properties of B-spline basis functions are listed as

- $N_{i,p}(u)$ is a polynomial over u_i ;
- $N_{i,p}(u)$ is non-negativity $N_{i,p}(u) \geq 0$;
- Each p order function $N_{i,p}(u)$ has $p - 1$ continuous derivatives across the element boundary when the knot multiplicity is one for all inner knots. The first derivative of a B-spline basis function can be easily calculated from the Cox-de Boor recursion formula

$$\frac{d}{du} N_{i,p}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (11)$$

- If k is the multiplicity of the i th knot, $N_{i,p}(u)$ is C^{p-k} continuous.

Then, a B-spline curve can be a linear combination of B-spline basis functions and is described as follows

$$C(u) = \sum_{i=1}^n N_{i,p}(u) P_i \quad (12)$$

where P_i is the i th control point ($i = 1, 2, \dots, n$).

With two known knot vectors $U = \{u_1, u_2, \dots, u_{n+p+1}\}$ and $V = \{v_1, v_2, \dots, v_{l+q+1}\}$, and $(n + 1) \times (l + 1)$ control points, a B-spline surface is given by

$$S(u, v) = \sum_{i=1}^n \sum_{j=1}^l N_{i,p}(u) N_{j,q}(v) P_{ij} \quad (13)$$

where P_{ij} is a control point.

3.2. NURBS Shape Function

Although B-spline possesses good properties, it can not represent circles, spheres, ellipsoids and paraboloids exactly. In order to overcome this disadvantage, a NURBS curve of order p is presented as a linear combination of B-spline bases by introducing weighting function

$$C(u) = \sum_{i=1}^n R_i(u) P_i \quad (14)$$

where $R_i(u)$, the NURBS basis function, is written as

$$R_i(u) = \frac{N_{i,p}(u) w_i}{\sum_{i=1}^n N_{i,p}(u) w_i} \quad (15)$$

w_i is the weighting function in 1D.

Similarly, the NURBS surface can be expressed as

$$S(u, v) = \sum_{i=1}^n \sum_{j=1}^l R_{i,j}(u, v) P_{ij} \quad (16)$$

The bivariate basis function $R_{i,j}(u, v)$ is written as

$$R_{i,j}(u, v) = \frac{N_{i,p}(u) N_{j,q}(v) w_{ij}}{\sum_{i=1}^n \sum_{j=1}^l N_{i,p}(u) N_{j,q}(v) w_{ij}} \quad (17)$$

where w_{ij} is the weighting function in 2D. In what follows, NURBS shape function R is symbolically replaced with N .

3.3. Numerical Integration in IGA

For the purpose of simplicity, only numerical integration over a 2D computational domain is presented herein. The integration scheme for 3D is not addressed in this work and can be easily obtained by just adding variables in the third dimension. As shown in

Figure 3, there are three domains, namely the physical domain Ω^e , parametric domain $\hat{\Omega}^e$ and parent domain $\bar{\Omega}^e$. The numerical integration over the whole physical domain ($\Omega = \sqcup \Omega^e$, Ω^e is the elemental domain) can be calculated through two main mapping steps, which are expressed as

$$\begin{aligned} & \int_{\Omega} f(x, y) d\Omega \\ &= \sum_{e=1}^{n_e} \int_{\Omega^e} f(x, y) d\Omega^e \\ &= \sum_{e=1}^{n_e} \int_{\hat{\Omega}^e} f(x(\hat{\xi}), y(\hat{\eta})) |\hat{J}| d\hat{\Omega}^e \\ &= \sum_{e=1}^{n_e} \int_{\bar{\Omega}^e} f(x(\bar{\xi}), y(\bar{\eta})) |\hat{J}| |\bar{J}| d\bar{\Omega}^e \end{aligned} \quad (18)$$

where $f(\cdot, \cdot)$ is the function to be integrated, and n_e is the number of elements (different from those in FEM, elements in IGA are related with a knot partition). $|\hat{J}|$ and $|\bar{J}|$ are determinants of the two Jacobian matrices for geometry mapping. The first Jacobian \hat{J} between the physical elemental domain and the parametric domain is given as

$$\hat{J} = \begin{bmatrix} x_{,\hat{\xi}} & x_{,\hat{\eta}} \\ y_{,\hat{\xi}} & y_{,\hat{\eta}} \end{bmatrix} \quad (19)$$

and the other Jacobian between the parametric and parent domains $\bar{J} = 0.25(\hat{\xi}_{i+1} - \hat{\xi}_i)(\hat{\eta}_{i+1} - \hat{\eta}_i)$. Then, a numerical integration scheme in FEM can be applied in the last term in Equation (18).

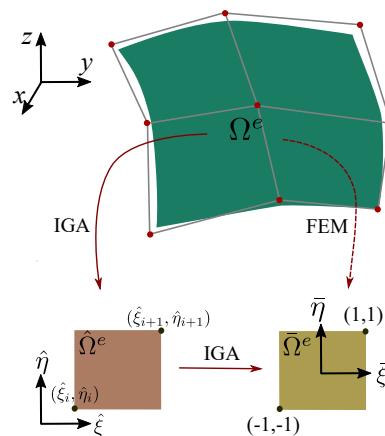


Figure 3. Definition of element domains in IGA. Physical elemental domain Ω^e (Top), elemental domain $\hat{\Omega}^e$ in parametric space (Bottom left) and Gaussian integration domain $\bar{\Omega}^e$ (Bottom right).

4. Computational Method

4.1. Spatial Discretization

Let us define the trial and weighting function spaces $\mathcal{V} \subset \mathcal{H}^2(\Omega)$. Here, $\mathcal{H}^2(\Omega)$ is the Sobolev space of integrable functions in the domain Ω . The finite-dimensional space for discretization is expressed as $\mathcal{V}^h = \text{span}\{N_i\}_{i=1,2,\dots,n}$, where n is the dimension of $\mathcal{V}^h \subset \mathcal{V} \subset \mathcal{H}^2(\Omega)$, and the N_i 's are linearly independent NURBS basis functions [24,25,28]. Taking the weighting function ω as a trial function, the Galerkin weak form of Equations (6) and (8) can be expressed as

$$\int_{\Omega} \omega^h \cdot \tau \frac{\partial \phi^h}{\partial t} d\Omega = \int_{\Omega} \omega^h \cdot \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \nabla \cdot \varepsilon^2 \nabla \phi^h - \mu(\phi^h, T^h) \right] d\Omega \quad (20)$$

$$\int_{\Omega} \omega^h \cdot \frac{\partial T^h}{\partial t} d\Omega = \int_{\Omega} \omega^h \cdot \nabla \cdot \nabla T^h d\Omega + \int_{\Omega} \omega^h \cdot K \frac{\partial \phi^h}{\partial t} d\Omega \quad (21)$$

After applying the integral by parts and divergence theorem (see Appendix A for details), the Galerkin weak form of Equations (20) and (21) can be written as

$$\begin{aligned} & \int_{\Omega} \left[- \left(\frac{\partial \omega^h}{\partial x}, \frac{\partial \omega^h}{\partial y} \right) \cdot \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x}, \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) \right] d\Omega \\ & - \int_{\Omega} \left[\left(\frac{\partial \omega^h}{\partial y}, \frac{\partial \omega^h}{\partial x} \right) \cdot \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x}, -\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) \right] d\Omega + \int_{\Omega} \left[-\omega^h \cdot \mu(\phi^h, T^h) - \omega^h \cdot \tau \frac{\partial \phi^h}{\partial t} \right] d\Omega = 0 \end{aligned} \quad (22)$$

$$\int_{\Omega} \omega^h \cdot \frac{\partial T^h}{\partial t} d\Omega + \int_{\Omega} \nabla \omega^h \cdot \nabla T^h d\Omega - \int_{\Omega} \omega^h \cdot K \frac{\partial \phi^h}{\partial t} d\Omega = 0 \quad (23)$$

The discrete solution ϕ and T are defined as

$$\phi(x, y, t)^h = \sum_{i=1}^n \phi_i(t) N_i(x, y), \quad (24)$$

$$T(x, y, t)^h = \sum_{i=1}^n T_i(t) N_i(x, y), \quad (25)$$

respectively. The ϕ_i and T_i are the control variables. The weighting function ω^h is expressed as

$$\omega(x, y)^h = \sum_{i=1}^n \omega_i N_i(x, y) \quad (26)$$

where the ω_i is the control variable.

4.2. Time Discretization

The dynamic Equations (22) and (23) are discretized by the generalized- α method [57] and solved by the Newton–Raphson method. Notations $\vec{\Phi}_n$ and \vec{T}_n are used for the global vector of control variables of the unknowns ϕ_n^h and T_n^h where the subscript n refers to the time step. We call $\dot{\vec{\Phi}}_n$ and $\dot{\vec{T}}_n$ the time derivatives of the control variable vectors. We further introduce $\vec{S}_n = \{\vec{\Phi}_n, \vec{T}_n\}$. Using this notation, the residual vectors are defined as

$$R_A^\phi(\vec{S}_{n+\alpha_f}, \dot{\vec{S}}_{n+\alpha_m}) = (N_A, \tau \phi_{n+\alpha_m}^h)_\Omega + (\nabla N_A, \varepsilon^2 \nabla \phi_{n+\alpha_f}^h)_\Omega - (\nabla N'_A, \varepsilon \varepsilon' \nabla \phi_{n+\alpha_f}^h)_\Omega + (N_A, \mu_\phi(\phi_{n+\alpha_f}^h, T_{n+\alpha_f}^h))_\Omega \quad (27)$$

$$R_A^T(\vec{S}_{n+\alpha_f}, \dot{\vec{S}}_{n+\alpha_m}) = (N_A, \dot{T}_{n+\alpha_m}^h)_\Omega + (\nabla N_A, \nabla T_{n+\alpha_f}^h)_\Omega - (N_A, K \phi_{n+\alpha_m}^h)_\Omega \quad (28)$$

where (\cdot, \cdot) is the inner product in the domain Ω . $\nabla N_A = (\partial N_A / \partial x, \partial N_A / \partial y)$. $\nabla N'_A = (\partial N_A / \partial y, \partial N_A / \partial x)$. Variables at the intermediate time level are given as [28,35]

$$\dot{\vec{S}}_{n+\alpha_m}^h = \dot{\vec{S}}_n^h + \alpha_m (\dot{\vec{S}}_{n+1}^h - \dot{\vec{S}}_n^h) \quad (29)$$

$$\vec{S}_{n+\alpha_f}^h = \vec{S}_n^h + \alpha_f (\vec{S}_{n+1}^h - \vec{S}_n^h) \quad (30)$$

$$\vec{S}_{n+1}^h = \vec{S}_n^h + \Delta t_n \dot{\vec{S}}_n^h + \gamma \Delta t_n (\dot{\vec{S}}_{n+1}^h - \dot{\vec{S}}_n^h) \quad (31)$$

Then, solve the below linear system using the generalized minimal residual method algorithm

$$\vec{K}_{n+1}^i \Delta \dot{\vec{S}}_{n+1}^i = -\vec{R}_{n+1}^i \quad (32)$$

where $\Delta \dot{\vec{S}}_{n+1}^i$ is the increment of time derivatives of the control variable vectors in each iteration. The tangent matrix involved in the i th iteration is given as

$$\vec{K}_{n+1}^i = \partial \vec{R}_{n+1}^i / \partial \dot{\vec{S}}_{n+1}^i \quad (33)$$

Following the instruction in reference [57], we take

$$\alpha_m = \frac{1}{2} \left(\frac{3 - \rho_\infty}{1 + \rho_\infty} \right); \alpha_f = \frac{1}{1 + \rho_\infty}; \gamma = \frac{1}{2} + \alpha_m - \alpha_f. \quad (34)$$

The choice of $\rho_\infty = 1/2$ in this work guarantees second-order accuracy and unconditional A-stability [57].

5. Numerical Results

In order to demonstrate the performance of the presented IGA method, two different geometric domains are adopted. One computational domain is square, and the other is an irregular area. In the former case, IGA and FDM are applied to solve the problems, and the results are compared. For the comparison purpose of computational efficiency, FEM is also adopted. In the latter case, only the IGA method is used, and an improved way to apply the supercooling rate to the droplet boundary is proposed.

5.1. Dendritic Formation in Square Domain

In this example, IGA, FEM and FDM are implemented for dendrite growth. All the parameters in each following test are set to be identical in the three methods, and their values are listed in Table 1 if no special statement is addressed. The effects of three typical parameters, such as latent heat, anisotropy mode and initial angle, on dendrite growth are observed. The square dimension is 9×9 , and we assume that the initial crystal is a circular shape and placed in the center of the square. The whole domain is discretized with a uniform computational mesh composed of 512×512 elements when IGA is implemented, while mesh with 512×512 bilinear quadrilateral elements is adopted in the case of FEM. The square is represented by 512×512 regular grids when FDM is carried out.

Table 1. Parameters used for dendrite growth.

Parameters	Symbols	In Silico Values
Relaxation time	τ	0.0003
Phase width	$\bar{\varepsilon}$	0.01
Anisotropy strength	δ	0.04
Anisotropy mode	j	6
Initial angle	θ_0	90°
Melting temperature	T_{eq}	1
Latent heat	K	1.6
Constant in Equation (3)	α	0.9
Constant in Equation (3)	γ	10

5.1.1. Effect of Latent Heat

Two different values of latent heat $K = 1.6$ and $K = 2.0$ are considered for the comparison of IGA, FEM and FDM. Other parameters are listed in Table 1. The dendrite structures obtained by IGA at two time stages $t = 0.28$ and $t = 0.36$ are plotted in Figure 4. All the results illustrate that the dendrite morphologies change with latent heat. The six primary dendrites grow along the selected directions. The angle between each primary

dendrite and y axis stays at 60° since the anisotropy mode is taken as $j = 6$ and the initial angle is set to be $\theta_0 = 90^\circ$. As we know, latent heat reflects the heat-released energy in the process of phase solidification [58]. With the increase in K , the heat at the front of dendrites accumulates and results in smaller supercooling. Thus, dendrites grow slower and are finer. These phenomena can be easily observed from panels (a) to (b) and panels (c) to (d). For instance, at time $t = 0.36$, the morphology in the case of $K = 1.6$ (panel (b)) reaches the boundary of the computational domain while it occupies less space and is still far to the boundary in the case of $K = 2.0$ (panel (d)).

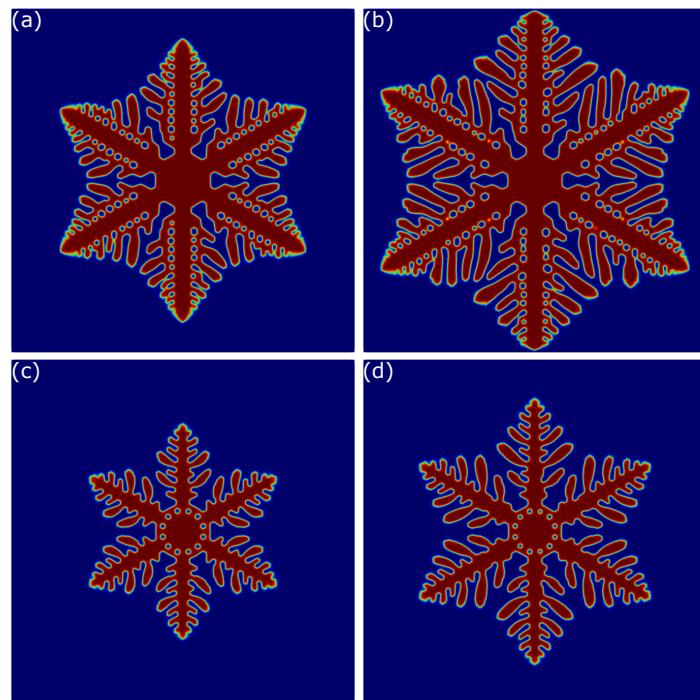


Figure 4. Dendrite structures obtained by IGA with $K = 1.6$ and $K = 2.0$ at different time stages. The first row shows the morphologies of dendrites in the case of $K = 1.6$ at $t = 0.28$ (a) and $t = 0.36$ (b). The second row shows those in the case of $K = 2$ at $t = 0.28$ (c) and $t = 0.36$ (d).

A similar conclusion can be obtained in cases of FEM and FDM (see Figures 5 and 6). From the structures in Figures 4–6, we notice that the interfaces obtained by IGA and FEM are much smoother than those from FDM. It can be explained that the NURBS-based IGA, as well as FEM, can capture the feathers of interfaces easily and accurately, even across mesh boundaries.

A comparison of computational efficiency among the three methods is presented in Table 2. The same numerical case $K = 2.0$ is carried out by all of them. $\Delta t = 0.000025$ and total time step $N = 7200$ is chosen by FDM due to the computational stability in the case of 512×512 grids. Constant time step $\Delta t = 0.001$ and total time step $N = 360$ are used by IGA and FEM in the case of 512×512 elements. The numbers of elements used by IGA and FEM, for example, 256×256 , 512×512 and 1024×1024 , are chosen for comparison. The implementation of IGA is carried out by an in-house developed message passing interface (MPI) Fortran code, while the FEM code is from the open-source, parallel object-oriented C++ finite element framework, MOOSE (Multiphysics Object Oriented Simulation Environment) [59]. Both of them are implemented in the same high-performance cluster (Intel(R) Xeon(R) CPU E5-2692 v2 2.20 GHz). The Matlab code of FDM for the solidification can be found in reference [60]. From Table 2, the computational efficiencies of IGA and FEM are almost at the same level, while FDM shows high efficiency as expected. Indeed, FDM takes less computational effort. However, it is not easy to deal with a complicated computational domain and apply a boundary condition. In the

following section, a simulation of a solidification problem in a droplet-like geometry is implemented by IGA.

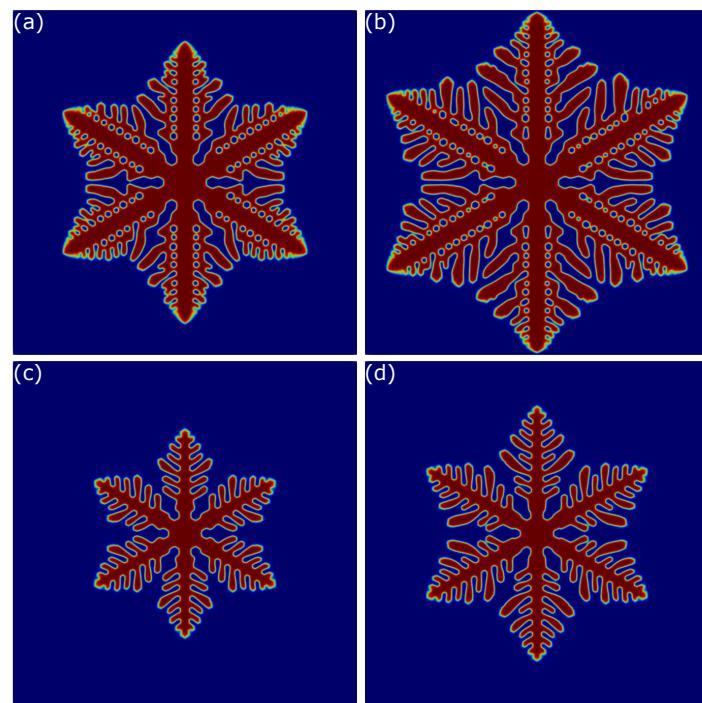


Figure 5. Dendrite structures obtained by FEM with $K = 1.6$ and $K = 2.0$ at different time stages. The first row shows the morphologies of dendrites in the case of $K = 1.6$ at $t = 0.28$ (a) and $t = 0.36$ (b). The second row shows those in the case of $K = 2$ at $t = 0.28$ (c) and $t = 0.36$ (d).

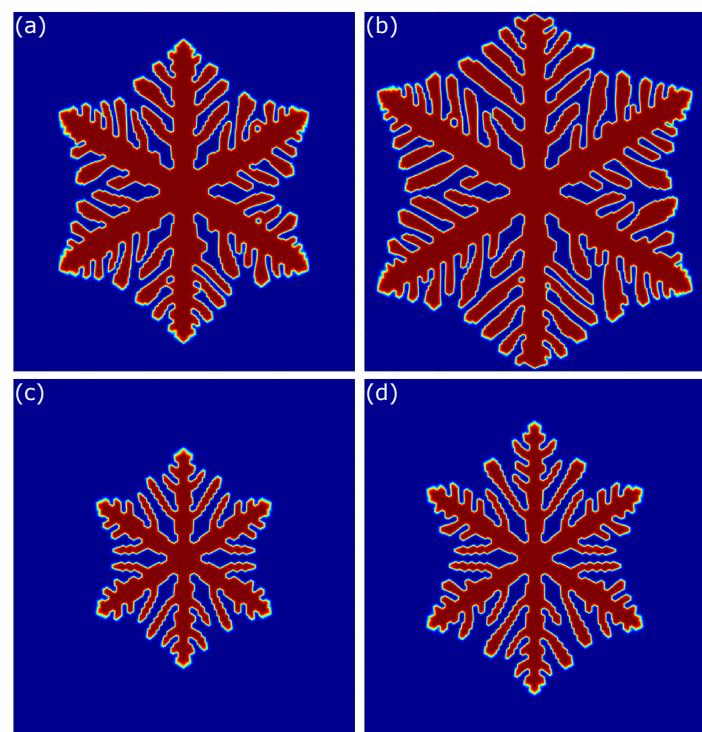


Figure 6. Dendrite structures obtained by FDM with $K = 1.6$ and $K = 2.0$ at different time stages. The first row shows the morphologies of dendrites in the case of $K = 1.6$ at $t = 0.28$ (a) and $t = 0.36$ (b). The second row shows those in the case of $K = 2$ at $t = 0.28$ (c) and $t = 0.36$ (d).

Table 2. CPU time consumed by IGA, FEM and FDM for the problem with $K = 2.0$ (unit of CPU time: minute).

Method	Mesh	Processor Number	CPU Time
The proposed method	256 × 256	4	8.2
	256 × 256	16	2.7
	256 × 256	64	0.9
	512 × 512	4	37
	512 × 512	16	12
	512 × 512	64	4.2
	1024 × 1024	4	160
	1024 × 1024	16	54
	1024 × 1024	64	19
FEM	256 × 256	4	9.1
	256 × 256	16	2.9
	256 × 256	64	1.1
	512 × 512	4	44
	512 × 512	16	15
	512 × 512	64	4.8
	1024 × 1024	4	173
	1024 × 1024	16	62
	1024 × 1024	64	23
FDM	512 × 512	1	52

5.1.2. Effect of Anisotropy Mode

The anisotropy mode directly affects the shape of dendrite growth and determines the number of primary dendrites. In this subsection, two different modes $j = 4$ and $j = 6$ are tested, and the corresponding results computed by IGA and FDM are compared in Figure 7.

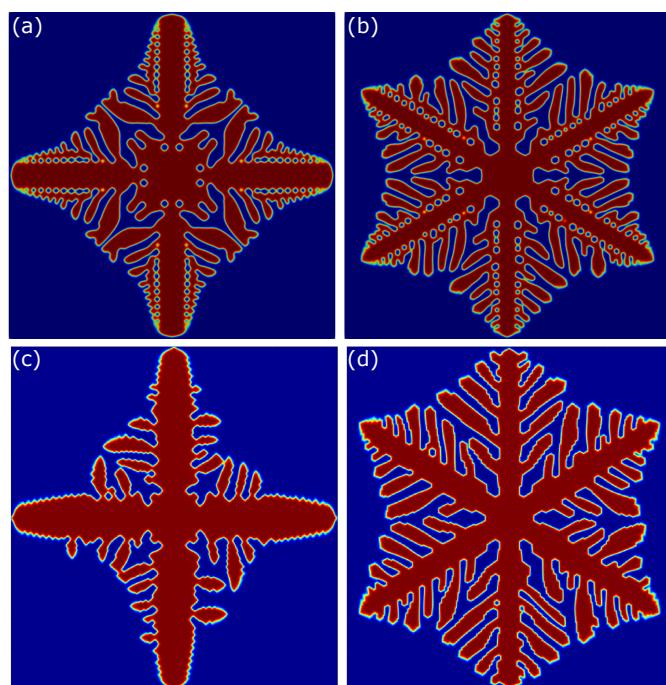


Figure 7. Simulations of dendrite growth with different anisotropy modes solved by IGA (panels (a,b)) and FDM (panels (c,d)). The morphologies of dendrites in panels (a,c) are obtained by using $j = 4$, and those morphologies in panels (b,d) are for $j = 6$ at the same time $t = 0.36$.

Four primary branches are obtained both by IGA and FDM, as shown in Figure 7a,c, in the case of mode $j = 4$, while six branches are obtained by IGA and FDM in Figures 7b,d in the case of mode $j = 6$. As the anisotropy mode increases, the number of primary dendrites increases. From the results plotted in Figure 7, the anisotropy mode significantly affects the subsequent morphologies of dendrites and plays an important role in the physical and chemical properties of crystals. Comparing the lengths of the main branches, we can conclude that the growth rates of the primary crystals keep the same in both numerical methods when mode $j = 4$ is chosen. However, the second branch grows faster when $j = 6$ is adopted. Although similar morphologies of crystals are obtained by both IGA and FDM, we still notice that the interfaces from IGA are more smooth than those from FDM. This may be explained by the intrinsic merit of NURBS basis functions.

5.1.3. Effect of Initial Angle

The physical meaning of the initial angle θ_0 is the orientation of the anisotropy axis, and it determines the growth angles of consequent crystals from primary dendrites. The effect of the initial angle on crystal orientation can be observed easily in numerical simulation. As shown in Figure 8, two different angles $\theta_0 = 90^\circ$ and $\theta_0 = 135^\circ$ are observed. Panels in the first and second rows in Figure 8 demonstrate the structures obtained from IGA and FDM, respectively.

Comparing with panels (a) and (b), as well as panels (c) and (d), the change in initial angle only affects the orientation of the primary dendrite. The shapes and structures of the formed dendrites should remain the same theoretically. However, the two morphologies in panels (a) and (b) are not exactly identical when structures in panel (b) rotate 45° in a clockwise direction. Similar situations can also be found in panels (c) and (d). There are several tiny differences in front of the interfaces and sub-branches. These differences result from the errors in numerical computations, initial condition settings, the meshing generation in the interface front and so on.

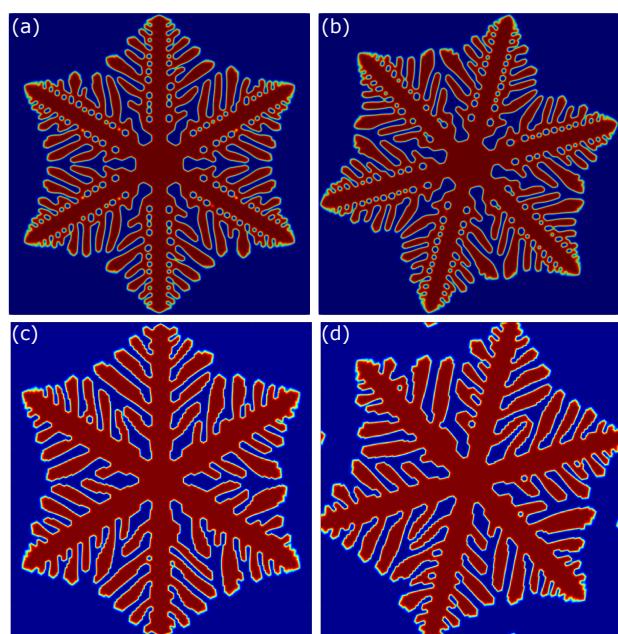


Figure 8. Simulation results of dendrite growth affected by initial angle. The morphologies of dendrites with $\theta_0 = 90^\circ$ and $\theta_0 = 135^\circ$ are compared at $t = 0.36$. (a,b) The morphologies of dendrites with $\theta_0 = 90^\circ$ and $\theta_0 = 135^\circ$ from IGA. (c,d) Under the same conditions, the morphologies of dendrites from FDM.

5.2. Dendrite Growth under Different Cooling Rates on Droplet Surface

We study the effect of different supercooling rates on the crystallization in a melt droplet. Panel (a) of Figure 9 illustrates the geometry of the melt droplet, and panel

(b) shows an example of an IGA mesh representation of the semicircular domain. We can see that the boundary of the droplet is exactly represented by a few IGA elements. A uniform computational mesh composed of $512 \times 256 \mathcal{C}^2$ elements is used in all the following simulations.

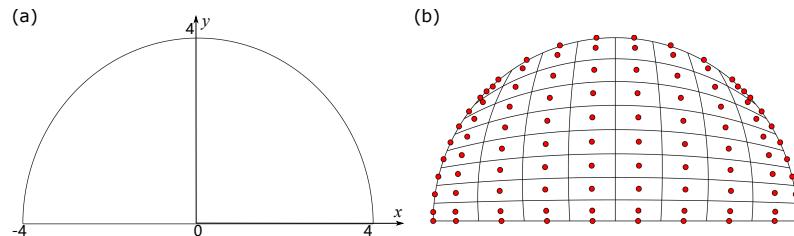


Figure 9. Geometry of the melt droplet (a) and an example of IGA mesh (b). The domain is discretized by 8×8 elements just for viewing purposes. The corresponding control points are indicated by solid red circles.

In order to describe the supercooling on the drop surface, an extra term is added to Equation (8). Then, the updated formula for temperature evolution is expressed as

$$\frac{\partial T}{\partial t} = \nabla \cdot \nabla T + K \frac{\partial \phi}{\partial t} - c \mathcal{H}(d) \quad (35)$$

where c is the dimensionless undercooling rate, and $\mathcal{H}(\cdot)$ is the Heaviside function that takes the value of 1 on the boundary and 0 in the remaining region. d is the distance to the boundary surface. In this sense, any complex interior/exterior boundary conditions and supercooling rate can be applied freely by adopting the least squares method [61]. Three different cooling rates, namely $c = 0$, $c = 10^2$ and $c = 10^5$, are studied. All other parameters take the values listed in Table 1. The initial crystal is circular with a radius 0.1, and its center is located at point (4,0), as shown in Figure 10a. The initial temperature in the droplet is expressed by $T_0 = 1 - (-7.785d_0^2 + 2289)/2289$, where d_0 is the distance to the droplet center and is given in Figure 10b.

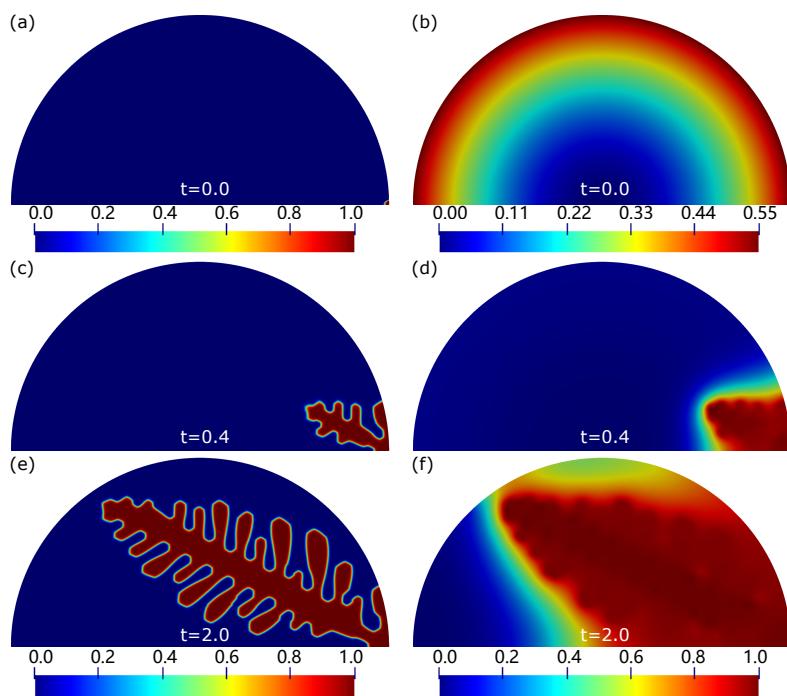


Figure 10. Evolution of dendrite structures and temperature at different stages in the case of $c = 0$. Initial nucleus (a) and distribution of normalized temperature within the domain (b); dendrite structures at $t = 0.4$ (c) and $t = 2.0$ (e), and temperature evolution at $t = 0.4$ (d) and $t = 2.0$ (f).

Initially, we study the dendrite growth without supercooling at the surface, which keeps the rate at $c = 0$. Due to the faster temperature gradient inside the droplet than that on the surface, the main branch grows faster in the interior region than the other one (see panels (c) and (e) in Figure 10). As the energy is released in front of the interface during the solidification process, the temperature increases gradually in the solid phase (see panels (d) and (f) in Figure 10). See Video S1 in the Supplemental Material for an illustration, and a detailed description of the movie can be found in Appendix B.

Then, we increase the cooling rate to $c = 10^2$ and keep other parameters the same. The crystallization on the boundary surface shows faster growth than that in the case of $c = 0$ at time $t = 0.2$, by comparing the two panels of Figures 10c and 11a. Furthermore, we notice that the first branch on the surface grows faster than that in the interior as time evolves (see panels in Figure 11). Eventually, the surface growth stops when the dendrite reaches the other end of the semicircle, and its second sub-branches grow toward the droplet center due to the temperature gradient. The corresponding illustration can be found in the Supplemental Material (Video S2).

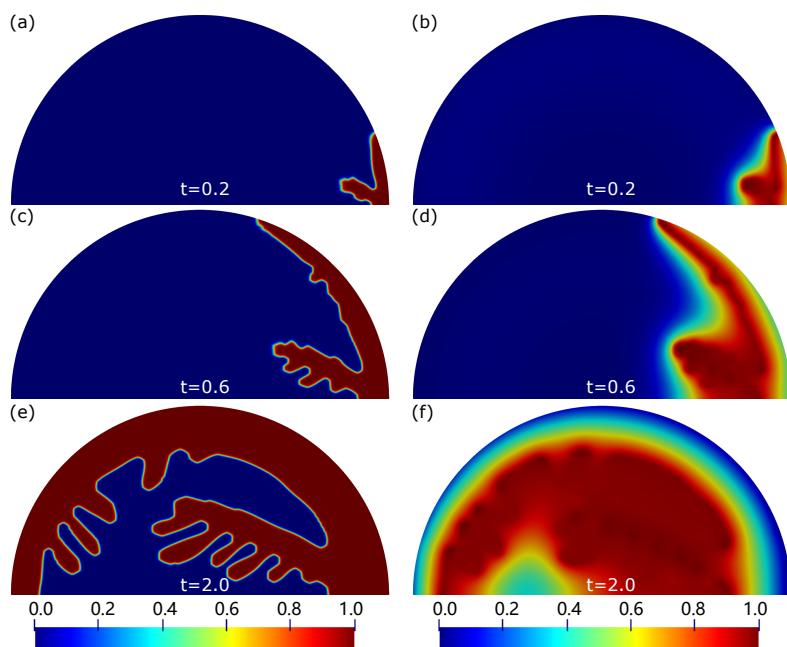


Figure 11. Evolution of dendrite structures and temperature at different stages in the case of $c = 10^2$. Initial nucleus (a) and distribution of normalized temperature within the domain (b); Dendrite structures at $t = 0.6$ (c) and $t = 2.0$ (e), and temperature evolution at $t = 0.6$ (d) and $t = 2.0$ (f).

We further increase the supercooling rate up to $c = 10^5$ and obtain the dendrite and temperature evolution, as illustrated in Figure 12 and Video S3 in the Supplemental Material. Unlike the co-growth of the dendrite on the surface and in the interior, the crystallization growth is almost along the boundary, and no sub-branch is developed except the other main branch at the very beginning ($t = 0.2$). When the dendrite on the surface reaches the other end of the semicircle, it also gradually grows toward the center. As expected, the distribution of temperature on the surface tends to be zero because of the rapid cooling speed (see panel (f) of Figure 12).

According to the findings in the three cases, we can conclude that the dendrite growth can be controlled by the undercooling rate. With the increase in rate, the growth on the surface is dominant, and eventually, only surface growth exists when an extreme rate is reached. This observation matches the numerical results from reference [5] and experimental observations in [52]. As shown in Figure 13, the dendrite growth towards the interior is dominant without the boundary supercooling (panels (a) and (d)). With small cooling rates on the surface, growth along the boundary and towards the inside co-exists (panels (b) and (e)). When a much bigger rate is applied to the boundary surface, the solidification only happens along the circumferential direction initially and then develops uniformly towards the center of the droplet (panels (c) and (f)).

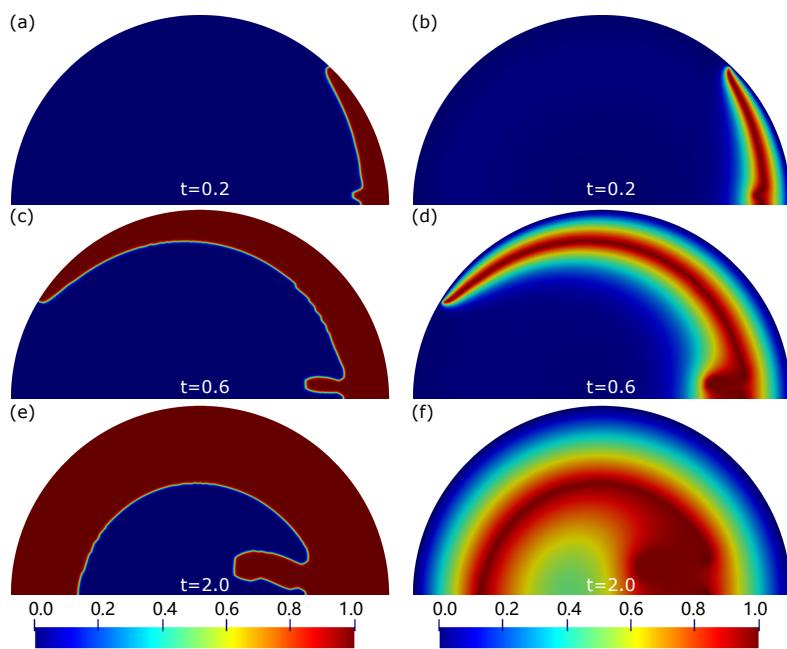


Figure 12. Evolution of dendrite structures and temperature at different stages in the case of $c = 10^5$. Initial nucleus (a) and distribution of normalized temperature within the domain (b); dendrite structures at $t = 0.6$ (c) and $t = 2.0$ (e), and temperature evolution at $t = 0.6$ (d) and $t = 2.0$ (f).

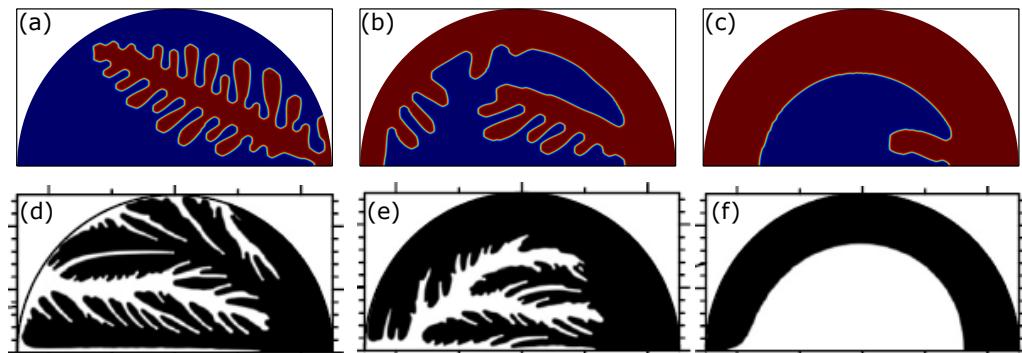


Figure 13. Evolution of dendrite structures at different stages in cases of various cooling rates. Panels (a–c) and (d–f) are obtained from the proposed method and FDM in reference [5], respectively. Reproduced with permission from Miura et al. [5] (d–f).

6. Conclusions

The phase-field method has been commonly embedded in FDM to describe the dendrite growth during solidification. As we know, FDM is hard to deal with regarding problems with irregular boundaries. In order to overcome this bottleneck, we present an IGA-based computational framework for 2D dendrite growth, which can be easily extended for 3D simulation. Numerical studies show that the performance of IGA is comparable to that of FEM and is even better than FDM in some cases. In addition, an improved way to deal with irregular boundary supercooling is also illustrated, and numerical findings agree well with other researchers' work and experimental observations. In this work, the geometry of the droplet is not complex but is typical. In the case of extremely complex geometry, no matter how complicated the boundary is, we just need to identify it through the Heaviside function $\mathcal{H}(\cdot)$. We believe that IGA creates another way to study dendrite growth and other related fields in material science.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/met12111836/s1>, Video S1: Solidification simulation in a droplet

with a supercooling rate $c = 0$ on the surface; Video S2: Solidification simulation in a droplet with a supercooling rate $c = 10^2$ on the surface; Video S3: Solidification simulation in a droplet with a supercooling rate $c = 10^5$ on the surface.

Author Contributions: Conceptualization, J.X. and Y.W. (Yuan Wang); formal analysis, J.X., T.Y., Z.Y., Y.W. (Yun Wang) and Y.W. (Yuan Wang); funding acquisition, J.X.; investigation, J.X. and Y.W. (Yuan Wang); methodology, J.X. and Y.W. (Yuan Wang); supervision, J.X.; writing—review & editing, J.X., Y.L. and Y.W. (Yuan Wang); data curation, T.Y., Y.L. and Z.Y.; software, T.Y., Y.L., Z.Y. and Y.W. (Yun Wang); validation, T.Y. and Y.W. (Yuan Wang); visualization, T.Y.; writing—original draft, T.Y.; resources, Y.L., Z.Y. and Y.W. (Yuan Wang). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Senior Talent Foundation of Jiangsu University grant number 19JDG022.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Formula Derivation

The right side of Equation (20) can be rewritten as

$$\begin{aligned} & \int_{\Omega} \omega^h \cdot \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \nabla \cdot \varepsilon^2 \nabla \phi^h - \mu(\phi^h, T^h) \right] d\Omega \\ &= \int_{\Omega} \omega^h \cdot \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) - \mu(\phi^h, T^h) \right] d\Omega \end{aligned} \quad (\text{A1})$$

Applying an integral by parts in Equation (A1), we obtain

$$\begin{aligned} & \int_{\Omega} \omega^h \cdot \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \nabla \cdot \varepsilon^2 \nabla \phi^h - \mu(\phi^h, T^h) \right] d\Omega \\ &= \int_{\Omega} \left\{ \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial y} \right) \right] + \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial x} + \left[\frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial x} \right) \right] - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial y} \right. \\ & \quad \left. + \left[\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial x} \right) \right] - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial x} + \left[\frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial y} \right) \right] - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial y} - \omega^h \cdot \mu(\phi^h, T^h) \right\} d\Omega \\ &= \int_{\Omega} \left\{ \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial y} \right) \right] + \left[\frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial x} \right) \right] + \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial x} - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial y} \right. \\ & \quad \left. + \left[\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial x} \right) \right] + \left[\frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \omega^h \frac{\partial \phi^h}{\partial y} \right) \right] - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial x} - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial y} - \omega^h \cdot \mu(\phi^h, T^h) \right\} d\Omega \end{aligned} \quad (\text{A2})$$

Using divergence theorem, Equation (A2) can be rewritten as

$$\begin{aligned} & \int_{\Omega} \omega^h \cdot \left[-\frac{\partial}{\partial x} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) + \frac{\partial}{\partial y} \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \right) + \nabla \cdot \varepsilon^2 \nabla \phi^h - \mu(\phi^h, T^h) \right] d\Omega \\ &= \int_{\Omega} \left[\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial x} - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial y} - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x} \frac{\partial \omega^h}{\partial x} - \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \frac{\partial \omega^h}{\partial y} - \omega^h \cdot \mu(\phi^h, T^h) \right] d\Omega \\ &= \int_{\Omega} \left[- \left(\frac{\partial \omega^h}{\partial x}, \frac{\partial \omega^h}{\partial y} \right) \cdot \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x}, \varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) - \left(\frac{\partial \omega^h}{\partial y}, \frac{\partial \omega^h}{\partial x} \right) \cdot \left(\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial x}, -\varepsilon \varepsilon' \frac{\partial \phi^h}{\partial y} \right) - \omega^h \cdot \mu(\phi^h, T^h) \right] d\Omega \end{aligned} \quad (\text{A3})$$

Appendix B. Movie Description

Movie1.avi: Solidification simulation in a droplet with a supercooling rate $c = 0$ on the surface. The dendrite structure can be controlled by the cooling rate. In this case, rate $c = 0$ means non-cooling is applied on the droplet surface, and dendrite growth mainly happens inside the droplet. Parameter values used in movies 1–3 are listed in Table 1.

Movie2.avi: Solidification simulation in a droplet with a supercooling rate $c = 10^2$ on the surface. With a non-zero cooling rate on the boundary, dendrites grow both on the surface and in the interior region.

Movie3.avi: Solidification simulation in a droplet with a supercooling rate of $c = 10^5$ on the surface. By increasing the cooling rate on the droplet surface, the dendrite primarily grows along the surface boundary and then towards the center.

References

1. Zhou, J.; Qi, L. Numerical simulation of liquid-solid extrusion process based on the mechanical model coupled with solidification. *Adv. Mech. Eng.* **2013**, *5*, 932348. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Ferreira, A.F.; de Castro, J.A.; Ferreira, I.L. 2D phase-field simulation of the directional solidification process. *Appl. Mech. Mater.* **2015**, *704*, 17–21. [\[CrossRef\]](#)
3. Amar, M.B.; Pomeau, Y. Theory of dendritic growth in a weakly undercooled melt. *Europhys. Lett.* **1986**, *2*, 307. [\[CrossRef\]](#)
4. Rappaz, M.; Gandin, C.-A. Probabilistic modelling of microstructure formation in solidification processes. *Acta Metall. Mater.* **1993**, *41*, 345–360. [\[CrossRef\]](#)
5. Miura, H.; Yokoyama, E.; Nagashima, K.; Tsukamoto, K.; Srivastava A. Phase-field simulation for crystallization of a highly supercooled forsterite-chondrule melt droplet. *J. Appl. Phys.* **2010**, *108*, 114912. [\[CrossRef\]](#)
6. Chen, L.Q. Phase-field models for microstructure evolution. *Annu. Rev. Mater. Res.* **2002**, *32*, 113–140. [\[CrossRef\]](#)
7. Militzer, M. Phase field modeling of microstructure evolution in steels. *Curr. Opin. Solid State Mater. Sci.* **2011**, *15*, 106–115. [\[CrossRef\]](#)
8. Elder, K.R.; Provatas, N.; Berry, J.; Stefanovic, P.; Grant, M. Phase-field crystal modeling and classical density functional theory of freezing. *Phys. Rev. B* **2007**, *75*, 064107. [\[CrossRef\]](#)
9. Chen, L.; Chen, J.; Lebensohn, R.A.; Ji, Y.Z.; Heo, T.W.; Bhattacharyya, S.; Chang, K.; Mathaudhu, S.; Liu, Z.K.; Chen, L.-Q. An integrated fast fourier transform-based phase-field and crystal plasticity approach to model recrystallization of three dimensional polycrystals. *Comput. Meth. Appl. Mech. Eng.* **2015**, *285*, 829–848. [\[CrossRef\]](#)
10. Jeong, J.-H.; Goldenfeld, N.; Dantzig, J.A. Phase field model for three-dimensional dendritic growth with fluid flow. *Phys. Rev. E* **2001**, *64*, 041602. [\[CrossRef\]](#)
11. Karma, A.; Kessler, D.A.; Levine, H. Phase-field model of mode III dynamic fracture. *Phys. Rev. Lett.* **2001**, *87*, 045501. [\[CrossRef\]](#)
12. Borden, M.J.; Hughes, T.J.R.; Landis, C.M.; Verhoosel, C.V. A higher-order phase-field model for brittle fracture: Formulation and analysis within the isogeometric analysis framework. *Comput. Meth. Appl. Mech. Eng.* **2014**, *273*, 100–118. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Zheng, X.; Babaee, H.; Dong, S.; Chryssostomidis, C.; Karniadakis, G. A phase-field method for 3D simulation of two-phase heat transfer. *Int. J. Heat Mass Transf.* **2015**, *82*, 282–298. [\[CrossRef\]](#)
14. Zhang, X.; Kang, J.; Guo, Z.; Han, Q. Effect of the forced flow on the permeability of dendritic networks: A study using phase-field-lattice Boltzmann method. *Int. J. Heat Mass Transf.* **2019**, *131*, 196–205. [\[CrossRef\]](#)
15. Sedaghatkish, A.; Sadeghiseraji, J.; Jamalabadi, M.Y.A. Numerical simulation of magnetic nanofluid (MNF) film boiling on cylindrical heated magnet using phase field method. *Int. J. Heat Mass Transf.* **2020**, *152*, 119546. [\[CrossRef\]](#)
16. Zhu, H.; Zhang, Z.; Xu, J.; Ren, Y.; Zhu, Z.; Xu, K.; Wang, Z.; Wang, C. A numerical study of picosecond laser micro-grooving of single crystalline germanium: Mechanism discussion and process simulation. *J. Manuf. Process.* **2021**, *69*, 351–367. [\[CrossRef\]](#)
17. Zhu, H.; Wang, C.; Mao, S.; Zhang, Z.; Zhao, D.; Xu, K.; Liu, Y.; Li, L.; Zhou, J. Localized and efficient machining of germanium based on the auto-coupling between picosecond laser irradiation and electrochemical dissolution: Mechanism validation and surface characterization. *J. Manuf. Process.* **2022**, *77*, 665–677. [\[CrossRef\]](#)
18. Bueno, J.; Bona-Casas, C.; Bazilevs, Y.; Gomez, H. Interaction of complex fluids and solids: Theory, algorithms and application to phase-change-driven implosion. *Comput. Mech.* **2015**, *55*, 1105–1118. [\[CrossRef\]](#)
19. Mokbel, D.; Abels, H.; Aland, S. A phase-field model for fluid–structure interaction. *J. Comput. Phys.* **2018**, *372*, 823–840. [\[CrossRef\]](#)
20. Frieboes, H.B.; Jin, F.; Chuang, Y.-L.; Wise, S.M.; Lowengrub, J.S.; Cristini, V. Three-dimensional multispecies nonlinear tumor growth-II: Tumor invasion and angiogenesis. *J. Theor. Biol.* **2010**, *264*, 1254–1278. [\[CrossRef\]](#)
21. Oden, J.T.; Hawkins-Daarud, A.J.; Prudhomme, S. General diffuse-interface theories and an approach to predictive tumor growth modeling. *Math. Models Methods Appl. Sci.* **2010**, *20*, 477–517. [\[CrossRef\]](#)
22. Oden, J.T.; Prudencio, E.E.; Hawkins-Daarud, A.J. Selection and assessment of phenomenological models of tumor growth. *Math. Models Methods Appl. Sci.* **2013**, *23*, 1309–1338. [\[CrossRef\]](#)
23. Moure, A.; Gomez, H. Computational model for amoeboid motion: Coupling membrane and cytosol dynamics. *Phys. Rev. E* **2016**, *94*, 042423. [\[CrossRef\]](#)

24. Xu, J.; Vilanova, G.; Gomez, H. A mathematical model coupling tumor growth and angiogenesis. *PLoS ONE* **2016**, *53*, 449–464. [[CrossRef](#)]
25. Xu, J.; Vilanova, G.; Gomez, H. Full-scale, three-dimensional simulation of early-stage tumor growth: The onset of malignancy. *Comput. Meth. Appl. Mech. Eng.* **2016**, *314*, 126–146. [[CrossRef](#)] [[PubMed](#)]
26. Xue, S.L.; Li, B.; Feng, X.Q.; Gao, H.J. A non-equilibrium thermodynamic model for tumor extracellular matrix with enzymatic degradation. *J. Mech. Phys. Solids* **2017**, *104*, 32–56. [[CrossRef](#)]
27. Feng, X.Q. Advances in tumor biomechanics. *J. Med. Biomech.* **2019**, *34*, 115–120. [[CrossRef](#)]
28. Xu, J.; Vilanova, G.; Gomez, H. Phase-field model of vascular tumor growth: Three-dimensional geometry of the vascular network and integration with imaging data. *Comput. Meth. Appl. Mech. Eng.* **2020**, *359*, 112648. [[CrossRef](#)]
29. Pismen, L.M. *Patterns and Interfaces in Dissipative Dynamics*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006.
30. Steinbach, I. Phase-field models in materials science. *Model. Simul. Mater. Sci. Eng.* **2009**, *17*, 073001. [[CrossRef](#)]
31. Moelans, N.; Blanpain, B.; Wollants, P. An introduction to phase-field modeling of microstructure evolution. *Calphad* **2008**, *32*, 268–294.
32. Kobayashi, R.; Warren, J.A.; Carter, W.C. Vector-valued phase field model for crystallization and grain boundary formation. *Phys. D Nonlinear Phenom.* **1998**, *119*, 415–423. [[CrossRef](#)]
33. Warren, J.A.; Boettinger, W.J. Prediction of dendritic growth and microsegregation patterns in a binary alloy using the phase-field method. *Acta Metall. Mater.* **1995**, *43*, 689–703. [[CrossRef](#)]
34. Trivedi, R.; Kurz, W. Dendritic growth. *Int. Mater. Rev.* **1994**, *39*, 49–74. [[CrossRef](#)]
35. Hughes, T.J.; Cottrell, J.A.; Bazilevs, Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Meth. Appl. Mech. Eng.* **2005**, *194*, 4135–4195. [[CrossRef](#)]
36. Shojaee, S.; Ghelichi, M.; Izadpanah, E. Combination of isogeometric analysis and extended finite element in linear crack analysis. *Struct. Eng. Mech.* **2013**, *48*, 125–150. [[CrossRef](#)]
37. Singh, S.; Singh, I.V.; Bhardwaj, G.; Mishra, B. A Bézier extraction based XIGA approach for three-dimensional crack simulations. *Adv. Eng. Softw.* **2018**, *125*, 55–93. [[CrossRef](#)]
38. Chen, L.; Lingen, E.J.; de Borst, R. Adaptive hierarchical refinement of NURBS in cohesive fracture analysis. *Int. J. Numer. Methods Eng.* **2017**, *112*, 2151–2173. [[CrossRef](#)]
39. Luu, A.-T.; Kim, N.-I.; Lee, J. NURBS-based isogeometric vibration analysis of generally laminated deep curved beams with variable curvature. *Compos. Struct.* **2015**, *119*, 150–165. [[CrossRef](#)]
40. Valizadeh, N.; Natarajan, S.; Gonzalez-Estrada, O.A.; Rabczuk, T.; Bui, T.Q.; Bordas, S.P. NURBS-based finite element analysis of functionally graded plates: Static bending, vibration, buckling and flutter. *Compos. Struct.* **2013**, *99*, 309–326. [[CrossRef](#)]
41. Faroughi, S.; Shafei, E.; Rabczuk, T. Multi-patch NURBS formulation for anisotropic variable angle tow composite plates. *Compos. Struct.* **2020**, *241*, 11196. [[CrossRef](#)]
42. Xu, G.; Mourrain, B.; Duvigneau, R.; Galligo, A. A new error assessment method in isogeometric analysis of 2D heat conduction problems. *Adv. Sci. Lett.* **2012**, *10*, 508–512. [[CrossRef](#)]
43. Gómez, H.; Calo, V.M.; Bazilevs, Y.; Hughes, T.J.R. Isogeometric analysis of the Cahn–Hilliard phase-field model. *Comput. Meth. Appl. Mech. Eng.* **2008**, *197*, 4333–4352.
44. Gomez, H.; Hughes, T.J.R.; Nogueira, X.; Calo, V.M. Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations. *Comput. Meth. Appl. Mech. Eng.* **2010**, *199*, 1828–1840. [[CrossRef](#)]
45. Nagy, A.P.; Abdalla, M.M.; Gürdal, Z. Isogeometric sizing and shape optimisation of beam structures. *Comput. Meth. Appl. Mech. Eng.* **2010**, *199*, 1216–1230. [[CrossRef](#)]
46. Manh, N.D.; Evgrafov, A.; Gersborg, A.R.; Gravesen, J. Isogeometric shape optimization of vibrating membranes. *Comput. Meth. Appl. Mech. Eng.* **2011**, *200*, 1343–1353. [[CrossRef](#)]
47. Xu, G.; Mourrain, B.; Duvigneau, R.; Galligo, A. Parameterization of computational domain in isogeometric analysis: Methods and comparison. *Comput. Meth. Appl. Mech. Eng.* **2011**, *200*, 2021–2031. [[CrossRef](#)]
48. Ashour, M.; Valizadeh, N.; Rabczuk, T. Isogeometric analysis for a phase-field constrained optimization problem of morphological evolution of vesicles in electrical fields. *Comput. Meth. Appl. Mech. Eng.* **2021**, *377*, 113669. [[CrossRef](#)]
49. Dhote, R.P.; Gomez, H.; Melnik, R.N.V.; Zu, J. Isogeometric analysis of a dynamic thermo-mechanical phase-field model applied to shape memory alloys. *Comput. Mech.* **2014**, *53*, 1235–1250. [[CrossRef](#)]
50. Gomez, H.; Bures, M.; Moure, A. A review on computational modelling of phase-transition problems. *Philos. Trans. R. Soc. A* **2019**, *377*, 20180203. [[CrossRef](#)]
51. Valizadeh, N.; Rabczuk, T. Isogeometric analysis for phase-field models of geometric PDEs and high-order PDEs on stationary and evolving surfaces. *Comput. Meth. Appl. Mech. Eng.* **2019**, *351*, 599–642. [[CrossRef](#)]
52. Tsukamoto, K.; Kobatake, H.; Nagashima, K.; Satoh, H.; Yurimoto, H. Crystallization of cosmic materials in microgravity. *Lunar Planet. Sci.* **2001**, *32*, 1846. [[CrossRef](#)] [[PubMed](#)]
53. Kobayashi, R. A brief introduction to phase field method. *AIP Conf. Proc.* **2010**, *1270*, 282–291. [[CrossRef](#)]
54. Takaki, T. Phase-field modeling and simulations of dendrite growth. *ISIJ Int.* **2014**, *54*, 437–444.
55. Koric, S.; Thomas, B.G.; Voller, V.R. Enhanced latent heat method to incorporate superheat effects into fixed-grid multiphysics simulations. *Numer. Heat Transf. Part B* **2010**, *57*, 396–413.
56. de Boor, C. Package for Calculating with B-Splines. *SIAM J. Numer. Anal.* **1977**, *14*, 441–472. [[CrossRef](#)]

57. Jansen, K.E.; Whiting, C.H.; Hulbert, G.M. A generalized-method for integrating the filtered Navier-Stokes equations with a stabilized finite element method. *Comput. Meth. Appl. Mech. Eng.* **2000**, *190*, 305–319. [[CrossRef](#)]
58. Tiaden, J.; Nestler, B.; Diepers, H.-J.; Steinbach, I. The multiphase-field model with an integrated concept for modelling solute diffusion. *Phys. D Nonlinear Phenom.* **1998**, *115*, 73–86. [[CrossRef](#)]
59. Permann, C.J.; Gaston, D.R.; Andrš, D.; Carlsen, R.W.; Kong, F.; Lindsay, A.D.; Miller, J.M.; Peterson, J.W.; Slaughter, A.E.; Stogner, R.H.; et al. MOOSE: Enabling massively parallel multiphysics simulation. *SoftwareX* **2020**, *11*, 100430. [[CrossRef](#)]
60. Sanal, R. Numerical simulation of dendritic crystal growth using phase field method and investigating the effects of different physical parameter on the growth of the dendrite. *arXiv* **2014**, *1412*, 3197. [[CrossRef](#)]
61. Nguyen, V.P.; Anitescu, C.; Bordas, S.P.; Rabczuk, T. Isogeometric analysis: An overview and computer implementation aspects. *Math. Comput. Simulat.* **2015**, *117*, 89–116. [[CrossRef](#)]