Phase field study of the tip operating state of a freely growing dendrite against convection using a novel parallel multigrid approach

Z. Guo [a *], J. Mi [b], S. Xiong [a] and P.S. Grant [c]

[a] School of Materials Science and Engineering, Tsinghua University, Beijing 100084, China

[b] Department of Engineering, University of Hull, East Yorkshire, HU6 7RX, UK

[c] Department of Materials, University of Oxford, Parks Road, Oxford, OX1 3PH, UK

[*]Correspondence author: zhipeng_guo@mail.tsinghua.edu.cn

Abstract: Alloy dendrite growth during solidification with coupled thermal-solute-convection fields has been studied by phase field modeling and simulation. The coupled transport equations were solved using a novel parallel-multigrid numerical approach with high computational efficiency that has enabled the investigation of dendrite growth with realistic alloy values of Lewis number ~ 10^4 and Prandtl number ~ 10^{-2}. The detailed dendrite tip shape and character were compared with widely recognized analytical approaches to show validity, and shown to be highly dependent on undercooling, solute concentration and Lewis number. In a relatively low flow velocity regime, variations in the ratio of growth selection parameter with and without convection agreed well with theory.

Keywords: Dendrite formation; Phase-field method; Solidification microstructure; Parallel computing

1. Introduction

In the shaped casting industry the growth behavior of dendrites in a solidifying alloy controls

the as-cast microstructure and has a strong influence on final component mechanical properties. Cast structures in practice, even in simple binary alloys, are complex and rarely conform to easy classification as homogeneously columnar or equiaxed, and frequently present complex cellular/dendritic patterns that vary from place to place. There has been significant effort to better understand the underlying physics controlling the shape, length scale and solute redistribution processes occurring at a growing dendrite tip in an attempt to control the factors that determine final cast microstructure [1-5]. Both analytical and numerical approaches have been developed, but despite the well-known strong influence of liquid movement and convection on final microstructure in practice, only a small number of recent studies have begun to account for its influence on the prior, more developed thermal-solute approaches.

The operating state of a growing dendrite can be defined by the tip radius R_{tip} and the tip velocity v_{tip}. By assuming the tip to be a parabola (in 2-D) or a paraboloid of revolution (in 3-D) with parabolic tip radius R_{p} and the steady dendrite is isothermal with the solid at the melting temperature, Ivantsov [6] proposed the most widely quoted relationship for dendrite operating state for a purely thermally-controlled growing dendrite, comprising the relationship between external imposed undercooling $\Delta = (T_m - T_\infty)/(L/C_p)$ and the thermal Peclet number at the tip $Pe_T = R_p v_{tip}/(2\alpha)$ as $\Delta = Iv(Pe_T)$, where T_m is the melting temperature, T_∞ is the temperature of the undercooled melt, L is the latent of fusion, C_p is the specific heat, α is the thermal diffusivity and $Iv(x) = \sqrt{\pi x}\exp(x)erfc(\sqrt{x})$ is the Ivantsov function (in 2-D). Ivantsov's theory predicted that for a given undercooling, there were infinite pairs of (R_{tip}, v_{tip})

for the solution of the linked expression since only their product (Peclet number) could be determined. While convenient, this implication is in conflict with experiment where R_{tip} and v_{tip} are invariant for a fixed undercooling. Two subsequent approaches were then developed by the introduction of a *selection constant* defined as σ^*. The *marginal stability* theory was developed by Langer and Muller-Krumbhaar [7] and involved another relationship between the tip radius and velocity given as $R_p^2 v_{tip} / (d_0 \alpha) = (1/\sigma^*)^2$ where d_0 is the material thermal capillary length. Drawing on a stability analysis [8] based on the allowable shape of a perturbed, non-flat solid-liquid interface, they proposed $\sigma^* = 1/(2\pi)$. Ben-Jacob *et al* [9, 10] and Kessler *et al* [11, 12] developed other approaches that allowed for anisotropic surface energy to give a single, paired solution for R_p and v_{tip}, deduced from the fastest growing mode of perturbed solid-liquid interface, which led to an expression similar to the one given by Langer and Muller-Krumbhaar [7] i.e. $R_p^2 v_{tip}$ = constant. Kessler and Levine [13] extended this idea and found that the dendrite tip shape computed in this way generally displays a cusp (non-zero slope) at the tip and at a unique (R_p, v_{tip}) pair; the cusp reduces to a smooth shape with zero slope at the tip, which is called the *microscopic solvability* condition. Further numerical experiments revealed that the selection constant σ^* was dependent on the strength of the surface anisotropy ε i.e. $R_p^2 v_{tip} = f(\varepsilon)$. Nevertheless, experimental validation of these increasingly complicated analytical/numerical approaches has been difficult since they rely on controlling stable and well-characterized growth conditions, generally far from the more dynamic conditions expected in practice [14].

The extension of the *microscopic solvability* theory to binary alloys, where both solute and

thermal diffusion are important, was performed by Lipton, Glicksman and Kurz (LGK) [15] and Lipton, Kurz and Trivedi (LKT) [16]. These approaches are also characterized by the use of a selection constant σ^* (σ^*_{LGK} and σ^*_{LKT} will be used for the LGK and LTK theories respectively):

$$\sigma^* = \frac{d_0}{R_p Pe_T \left[\xi_T + 2\xi_c Le \left(\frac{Mc_\infty}{1-(1-k)\Delta_C} \right) \right]} \tag{1}$$

where $M = |m|(1-k)/(L/C_p)$ is the scaled dimensionless liquidus slope, m is the actual liquidus slope from the phase diagram, k is the solute partition coefficient, $Le = \alpha/D$ is the Lewis number, D is the solute diffusivity in liquid, c is the solute concentration and c_∞ is the equilibrium solute concentration. $\Delta_C = \left(c_{tip} - c_\infty\right) \Big/ \left((1-k)c_{tip}\right)$ is the dimensionless solutal undercooling and c_{tip} is the solute concentration at dendrite tip. For the LGK theory, both ξ_T and ξ_c are unity but for the LKT theory:

$$\xi_c = 1 + \frac{2k}{1-2k-\sqrt{1+\frac{1}{\sigma^*(Pe_c)^2}}} \tag{2}$$

and

$$\xi_T = 1 - \frac{1}{\sqrt{1+\frac{1}{\sigma^*(Pe_T)^2}}} \tag{3}$$

The overall undercooling is then given by:

$$\Delta T = \frac{L}{C_p}\Delta_T + \frac{k\Delta T_0 \Delta_c}{1-(1-k)\Delta_c} + \frac{\Gamma}{R_p} \tag{4}$$

where the three terms on the right correspond to thermal, solutal and capillary undercooling, respectively. $\Delta T_0 = |m|c_\infty(1-k)/k$ is the equilibrium freezing range corresponding to c_∞ and

Γ is the Gibbs-Thomson coefficient. $\Delta_T = \left(T_{tip} - T_\infty\right) \Big/ \left(L / C_p\right)$ is the dimensionless thermal undercooling. Eqs. (1) and (4) together uniquely determine the tip radius and tip velocity.

Convection in the melt – almost always significant in practice – has long been realized to have a profound effect on dendritic growth [17]. But it is presently unclear how the preceding theories (LGK and LKT) for binary alloys may remain valid or how they might be modified when convection is present. Ananth and Gill [18] and Saville and Beaghton [19] studied the motion of the freezing front between a needle-shaped crystal and a supercooled liquid for situations where there is forced convection aligned along the crystal growth. Analysis was conducted by modeling the transport problem for a pure material solidifying as a paraboloid of revolution in an infinite undercooled melt. The imposed external undercooling could be characterized by the thermal Peclet number Pe_T, the flow Peclet number $Pe_f = R_{tip}v_\infty/2\alpha$ (where v_∞ is the imposed external flow velocity) and the Prandtl number $Pr = \mu/\alpha$ (ratio between kinematic viscosity and thermal diffusivity) i.e. $\Delta = \Delta\left(Pe_T, Pe_f, \mathrm{Pr}\right)$. Through a so-called *linear solvability* analysis, Boissou and Pelce [20] considered the stability of this solution, and found that the ratio of the selection parameters with convection (σ^*) and without convection (σ_0^*) could be characterized by a dimensionless parameter χ_e:

$$\frac{\sigma_0^*}{\sigma^*} = 1 + b\chi_e^{11/14} \tag{5}$$

where b is a numerical constant,

$$\chi_e = a\left(\mathrm{Re}\right) d_0 v_\infty \Big/ \left(\left(15\varepsilon\right)^{0.75} R_p v_{tip}\right) \tag{6}$$

and

$$a\left(\mathrm{Re}\right)=\sqrt{2\,\mathrm{Re}/\pi}\exp\left(-\,\mathrm{Re}/2\right)/\,erfc\left(\sqrt{\mathrm{Re}/2}\right) \tag{7}$$

where Re = $v_\infty R_p/\nu$ is the Reynolds number and ε is the anisotropic strength. The theory predicts that parameter b in Eq. (5) is very close to zero i.e. $\sigma_0^* \approx \sigma^*$ when χ_e << 1, and therefore the flow only has an effect on the tip-selection parameter if $d_0 v_\infty/(R_p v_{tip}) = Pe_f \sigma^*$ is of the order of unity or greater [20].

The applicability of this approach has not yet been confirmed conclusively by numerical modeling or experiments. Beckermann *et al* [21] found a weak dependence of the selection parameter on flow. Tong *et al.* [22] adopted a 2-D thermal convection phase field model and studied the tip operating state corresponding to a pure material dendrite growing against forced convection. Within the parameter range of $0<\chi_e<0.2$, the simulation results agreed well with theoretical predictions and the ratio of the selection parameters with and without convection was very close to unity, as suggested by Boissou and Pelce. Jeong *et al*. [17] performed 3-D phase field simulations and compared the results with available theories and experimental results: the selection constant σ^* decreased slightly in the presence of convection, agreeing with the solutions of both Saville and Beaghton [19] and again with Boissou and Pelce [20], although the idealized conditions required in the theory were unlikely to be present during the experiments. A similar 3-D simulation approach was also adopted by Lu *et al.* [23] who showed again reasonable agreement with the Boissou and Pelce's theory that the selection constants under different conditions were similar whether convection was, or was not, present.

Apart from these pure material studies, binary *alloys* have also been considered in which the influence of external flow on dendrite growth was investigated. By employing an adaptive mesh methodology, Lan *et al.* [24] studied the influence of thermal-solutal convection on morphology changes of a freely growing dendrite. They showed the importance of employing the so-called "anti-trapping" current developed by Karma [25] in the phase field model in order to produce qualitatively acceptable predictions. Recently, Siquieri *et al.* [26] calculated the shape, using the phase field method, of a freely growing alloy dendrite under isothermal conditions where convection was present.

The phase field method for simulating dendritic solidification and other phase transitions has become popular in recent years due to its thermodynamic rigour and the perceived benefits of avoiding explicit tracking of the complex shaped and evolving solid-liquid interface. The phase field method adopts one or several order parameters to characterize the transition of the phases during solidification. In particular, to model the dendritic growth of alloys usually one order parameter, i.e. the phase filed ϕ, is used to distinguish the solid and liquid phases in which ϕ varies smoothly but steeply from -1 in bulk liquid to 1 in bulk solid over the diffuse interface region of width W_0.

As discussed in our earlier work [27], despite apparent benefits, a major problem for phase field modeling remains the enormous computing overhead required to resolve the detail of the solid-liquid interface in complex dendritic geometries. Several methods have been developed and proposed to help address the practical difficulties, of which the most important

was the so-called *thin interface limit* analysis, developed by Karma and co-workers [25, 28, 29]. This methodology enabled a diffuse interface width larger than the capillary length, which has the practical effect of greatly reducing the computing overhead. However, for alloy solidification processes in the casting industry, where multiple dendrite growth proceeds according to coupled thermal, solutal and fluid flow fields, use of the phase field approach has been restricted by the need thus far to make limiting simplifications to the real process physics.

The primary difficulties arise from three aspects: (1) the different length- and time-scales in the coupled energy and mass transport processes (high Lewis number $Le = \alpha/D \sim 10^4$ and low Prandtl number $Pr = \nu/\alpha \sim 10^{-2}$ for metallic alloys); (2) enormous computing demands arising from the requirement to achieve practically useful and realistic spatial resolution of the solid-liquid interface; and (3) a lack of robust numerical schemes that can solve such a complex multi-scale problem in a reasonable time span.

In this paper, an approach combining a full approximation storage (FAS) multigrid [30] algorithm and a parallel computing architecture is developed for the first numerical solutions of coupled thermal-solute-convection fields in the dendritic growth of a real alloy. The phase field model including convection and the parallel multigrid approach are firstly described and then numerical experiments are performed to investigate convergence and computational efficiency. Predictions of dendrite tip radius, tip velocity and particularly the selection constant are compared with widely accepted theory as a way to validate various behaviors

predicted by the model under different liquid flow conditions. In particular, we focus on the effect of convection on the selection constant σ^* according to Boissou and Pelce [20] as this requires a study of the linked effects of temperature, solute concentration and flow.

2. Mathematical model

The following assumptions were made:

(1) Physical properties of materials including thermal diffusivity α, liquid solute diffusivity D, latent heat of fusion L, alloy specific heat C_p, and kinematic viscosity υ are constant unless stated otherwise.

(2) Solute diffusion in the solid phase is neglected.

(3) Only forced convection is considered, i.e. the effect of gravity is neglected.

(4) The flow is incompressible i.e. the alloy density ρ is constant.

Formulation of the phase field model starts from the Ginzburg-Landau free energy functional [25, 31, 32]:

$$F = \int dV \left(\frac{\sigma}{2} |\nabla \phi|^2 + f_{AB}(\phi, c, T) \right) \qquad (8)$$

Where F is the free energy, σ is a gradient energy coefficient, f_{AB} is a free energy density function for a binary mixture [32]. With convection the governing equations for the phase field variable ϕ, solute concentration c and temperature T can then be written as (assuming the phase field is not convective [22])

$$\tau_0 \frac{\partial \phi}{\partial t} = - K_\phi \frac{dF}{d\phi} \qquad (9)$$

$$\frac{\partial c}{\partial t} + \vec{v} \cdot \nabla c = \nabla \cdot \left(K_c \nabla \frac{dF}{dc} - \vec{j}_{at} \right) \qquad (10)\ [26]$$

$$\frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T = a \nabla^2 T + \frac{L}{2C_p} \frac{\partial f}{\partial t} \qquad (11)\ [22]$$

where t_0 is the relaxation time, K_ϕ and K_c are constants, $\vec{v} = f_l \vec{v}_{act}$ is the superficial velocity with f_l the liquid fraction and $\vec{v}_{act}$ the actual velocity. The velocity can be expanded as $\vec{v} = (u, v)$ in the 2D case, and

$$\vec{j}_a = - \frac{W_0}{\sqrt{2}} \frac{c / c_\infty}{\left[1 + k - (1 - k) f \right]} \frac{\partial f}{\partial t} \frac{\nabla f}{|\nabla f|} \qquad (12).$$

is the so-called "anti-trapping" current developed by Karma [25] to counterbalance spurious effects during the phase field simulation.

The conservation of the momentum and mass follows the Navier-Stokes equation and the continuity equation, respectively [22]

$$\frac{\partial \vec{v}}{\partial t} + \vec{v} \cdot \nabla \left(\frac{\vec{v}}{f_l} \right) = - f_l \nabla P + \mu \nabla^2 \vec{v} + \vec{M}_1^d \qquad (13)$$

$$\nabla \cdot \vec{v} = 0 \qquad (14)$$

where P is the pressure, $\vec{M}_1^d = - 2\mu h_c (1 - f_l)^2 \vec{v}$ is a dissipative interfacial force per unit volume developed by Beckerman [21]. The constant h_c is 2.757 according to an asymptotic analysis of a plane flow past the diffuse interface. This term serves as a distributed momentum sink in the diffuse interface region that forces the liquid velocity to zero approaching the solid and vanishes in the bulk liquid.

Applying the dimensionless forms of solute and temperature respectively:

$$U = \frac{\frac{2c/c_\infty}{1+k-(1-k)\phi}-1}{1-k} \tag{15}$$

$$q = \frac{T - T_M - mc_\infty}{L/C_p} \tag{16}$$

where T_M is the melting point of the solvent, and by scaling length and time to the interfacial width W_0 and relaxation time τ_0, the final governing equations (incorporating crystal anisotropy) are:

$$\begin{aligned} &A(y)^2\left[\frac{1}{Le}+Mc_\infty\left(1+(1-k)U\right)\right]\frac{\partial f}{\partial t} = \\ &\nabla\cdot\left(A(y)^2\nabla f\right) - \frac{\partial}{\partial x}\left(A(y)A(y)'\frac{\partial f}{\partial y}\right)+\frac{\partial}{\partial y}\left(A(y)A(y)'\frac{\partial f}{\partial x}\right) \\ &+f\left(1-f^2\right)-l\left(1-f^2\right)^2\left(q+Mc_\infty U\right) \end{aligned} \tag{17}$$

$$\begin{aligned} &\left(\frac{1+k}{2}-\frac{1-k}{2}f\right)\frac{\partial U}{\partial t} = \\ &\nabla\cdot\left(\tilde{D}\frac{1-f}{2}\nabla U+\frac{1}{2\sqrt{2}}\left[1+(1-k)U\right]\frac{\partial f}{\partial t}\frac{\nabla f}{|\nabla f|}\right)+\frac{1}{2}\left[1+(1-k)U\right]\frac{\partial f}{\partial t} \\ &-\frac{1}{2}\tilde{v}\cdot\left\{\left[1+k-(1-k)f\right]\nabla U-\left[1+(1-k)U\right]\nabla f\right\} \end{aligned} \tag{18}$$

$$\frac{\partial q}{\partial t} = \tilde{a}\nabla^2 q+\frac{1}{2}\frac{\partial f}{\partial t}-\tilde{v}\cdot\nabla q \tag{19}$$

$$\frac{\partial \tilde{v}}{\partial t}+\tilde{v}\cdot\nabla\left(\frac{\tilde{v}}{f_l}\right) = -f_l\nabla\tilde{P}+\tilde{u}\nabla^2\tilde{v}-2\tilde{u}h_c\left(1-f_l\right)^2\tilde{v} \tag{20}$$

$$\nabla\cdot\tilde{v} = 0 \tag{21}$$

where variables with tilde are dimensionless and $\tilde{D} = l/a_2$, $\tilde{a} = Le\times\tilde{D}$, $\tilde{v} = \vec{v}\times t_0/W_0$, $\tilde{P} = P\times t_0^2/\left(r\times W_0^2\right)$ and $\tilde{u} = ut_0/W_0^2$. The anisotropic effect is characterized by function $A(y) = 1+e\cos\left[w(y-y_0)\right]$ where ψ is the angle between the interface norm $\vec{n}$ and the axis x ($y = \arctan\left(f_x'/f_y'\right)$ and $f_x' = \partial f/\partial x$, $f_y' = \partial f/\partial y$), e weights the magnitude of

the anisotropy strength, W is the symmetry or harmonic factor and γ_0 stands for the predefined growth orientation. The reciprocal of the scaling parameter

$$\lambda = \frac{15L^2}{16HC_pT_M} \quad (22)$$

measuring the energy barrier height H in the double well potential. The dimensionless variables are related to physical units by

$$W_0 = \lambda d_0 / a_1 \quad (23)$$

and

$$t_0 = d_0^2 a_2 \lambda^3 / \left(D a_1^2\right) \quad (24)$$

where

$$d_0 = \Gamma / \left(L / C_p\right) \quad (25)$$

is the thermal capillary length and a_1 = 0.8839 and a_2 = 0.6267 according to the thin interface limit analysis [25].

3. The numerical approach

3.1. Discretization of equations

It is easy to notice that Eqs. (17) – (21) can be re-written in a simple form as:

$$\Phi_t \frac{\partial E}{\partial t} = \Phi_l \nabla^2 E + \Phi_s E + \Phi_x \frac{\partial E}{\partial x} + \Phi_y \frac{\partial E}{\partial y} + \Phi_C + \Phi_E \quad (26)$$

where E is the target variable such as ϕ, U and q. Φ_t, Φ_l, Φ_s, Φ_x, Φ_y and Φ_C are constants which are always related to the values from the previous time step, Φ_E is a nonlinear term related to variable E which is nonzero only for phase field. Eqs. (17) – (21) were then all

discretized onto a 2D rectangular domain (as shown in Fig. 1) with equal spacing $\Delta x = \Delta y = h$ using a finite difference method.

Special attention must be paid when applying the finite difference discretization schemes on the phase field equations because spurious anisotropy effect might be introduced, as discussed in [33]. In this respect, an isotropic nine-point discretization scheme was employed here. The key idea of achieving isotropic discretization is by putting the residual error into a rationally invariational form which is basically the laplacian operation [28, 29, 33]. Accordingly (taking the phase field variable for instance),

$$\left.\frac{\partial f}{\partial x}\right|_{i,j} \approx \frac{f_{i+1,j+1} - f_{i-1,j+1} + f_{i+1,j-1} - f_{i-1,j-1} + 4\left(f_{i+1,j} - f_{i-1,j}\right)}{12h} \tag{27}$$

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{i,j} \approx \frac{1}{12h^2}\begin{Bmatrix} f_{i+1,j+1} + f_{i-1,j+1} + f_{i+1,j-1} + f_{i-1,j-1} \\ +10\left(f_{i+1,j} + f_{i-1,j}\right) - 2\left(f_{i,j+1} + f_{i,j-1}\right) \\ -20 f_{i,j} \end{Bmatrix} \tag{28}$$

$$\left.\nabla^2 f\right|_{i,j} \approx \frac{1}{6h^2}\begin{Bmatrix} f_{i+1,j+1} + f_{i-1,j+1} + f_{i+1,j-1} + f_{i-1,j-1} \\ +4\left(f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1}\right) \\ -20 f_{i,j} \end{Bmatrix} \tag{29}$$

$$\left.\frac{\partial^2 f}{\partial x \partial y}\right|_{i,j} \approx \frac{f_{i+1,j+1} + f_{i-1,j-1} - f_{i+1,j-1} - f_{i-1,j+1}}{4h^2} \tag{30}$$

Eqs. (27) – (30) were also employed for discretization of solute and temperature fields i.e. Eqs. (18) – (19). For Eqs. (20) – (21), a staggered grid was employed to solve the fluid flow field. Taking the *x* component of velocity i.e. *u* for instance, a simple first order upwind scheme was employed for the first derivative of velocity and a second order five-point stencil

was applied for the Laplacian operation [34] i.e.

$$\left.\frac{\partial u}{\partial x}\right|_{i,j} = \frac{\left[1 - \operatorname{sgn}(u^*)\right]u_{i+1,j} + 2\operatorname{sgn}(u^*)u_{i,j} - \left[1 + \operatorname{sgn}(u^*)\right]u_{i-1,j}}{2h} \qquad (31)$$

$$\nabla^2 u_{i,j} = \frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}}{4h^2} \qquad (32)$$

where

$$\operatorname{sgn}(u) = \begin{cases} +1 & ,u \geq 0 \\ -1 & ,u < 0 \end{cases} \qquad (33)$$

And u^* is the velocity value from the previous time step. In this respect and as shown in Fig. 1, for one grid cell, the corner points (solid spheres) were specified for phase field, temperature and solute, velocity was specified at the cell walls and pressure the cell center.

A second order backward difference formulae (BDF2) was employed for the first derivative of variable *E* with respect to time [27]:

$$\frac{\partial E}{\partial t} = \frac{E^{n+1} + a_0 E^n + a_1 E^{n-1}}{b\Delta t} \qquad (34)$$

where a_0 = -4/3, a_1 = 1/3, and b = 2/3.

3.2. The parallel multigrid approach

3.2.1. The multigrid algorithm

For a discretized algebra system i.e. $A' E = b$, where A is the parameter matrix (always sparse), E is the solution vector and b is the right-hand side vector. The key idea for solving this system using the multigrid algorithm is to remove the defect $d = b - AE$ by employing (recursively) the coarse grid correction. It is well known that with fixed grid size,

the computational error of the target algebra system can be divided into high frequency and low frequency categories. Smoothing techniques such as Gauss-Seidel can efficiently remove the high frequency error in a few iteration steps but can hardly or at least very inefficiently remove low frequency error. In this respect, by composing a series of grid levels with different grid sizes and by employing Gauss-Seidel method, the multigrid algorithm can remove all high frequency error on each level in a very efficient way i.e. it normally only takes a few smoothing steps during relaxation. Because nonlinearity is involved in the phase field equation, the full approximation storage (FAS) multigrid algorithm [30] is employed.

To solve $A_l E_l = b_l$ the FAS multigrid algorithm includes the following steps [30]:

(1) Pre-smoothing

– Compute $\overline{E}_l^n$ by applying N_1 smoothing steps to E_l^n i.e. $\overline{E}_l^n = S\left(E_l^n, A_l, b_l\right)$.

(2) Coarse-grid correction

– Compute the defect $\overline{d}_l^n = b_l - A_l \overline{E}_l^n$.

– Restrict the defect: $\overline{d}_{l-1}^n = R_l^{l-1} \overline{d}_l^n$.

– Restrict $\overline{E}_l^n$ i.e. $\overline{E}_{l-1}^n = \hat{R}_l^{l-1} \overline{E}_l^n$.

– Compute the right-hand side $b_{l-1} = d_{l-1}^n + A_{l-1} \overline{E}_{l-1}^n$.

– Compute an approximate solution $\hat{w}_{l-1}^n$ of the coarse grid equation on grid level l-1 i.e.

$$A_{l-1} w_{l-1}^n = b_{l-1} \qquad (35)$$

If $l = 1$ employ a fast solver for this purpose.

If $l > 1$ solve Eq. (35) by performing γ (≥ 1) FAS cycles using $\overline{E}_{l-1}^n$ as initial approximation

$$\hat{w}_{l-1}^{n} = \mathrm{FASCYC}^{g}\left(l-1, g, \bar{E}_{l-1}^{n}, A_{l-1}, b_{l-1}, N_1, N_2\right)$$

– Compute the correction $\hat{v}_{l-1}^{n} = \hat{w}_{l-1}^{n} - \bar{E}_{l-1}^{n}$.

– Interpolate the correction $\hat{v}_{l}^{n} = I_{l-1}^{l}\hat{v}_{l-1}^{n}$.

– Compute the corrected approximation on level l i.e. $\tilde{E}_{l}^{n} = \bar{E}_{l}^{n} + \hat{v}_{l}^{n}$.

(3) Post-smoothing

– Compute E_{l}^{n+1} by applying N_2 (≥ 0) smoothing steps to $\tilde{E}_{l}^{n}$: $E_{l}^{n+1} = S\left(\tilde{E}_{l}^{n}, A_l, b_l\right)$.

where the subscript l indicates the index of the current grid level, S is the relaxation operation, R_{l}^{l-1} and $\hat{R}_{l}^{l-1}$ are restriction operations and I_{l-1}^{l} is the interpolation (prolongation) operation. As seen, the multigrid algorithm mainly comprises three components including relaxation, restriction and interpolation.

After discretization, using the stencils in Eqs. (27) – (32), Eq. (26) can be written as:

$$G\left(E^{n+1}\right) = b_{E}^{n} \qquad (36)$$

where $G\left(E^{n+1}\right)$ is a general form of function related to variable E, and for a nine-point stencil

$$G\left(E^{n+1}\right) = p_{i,j}^{n}E_{i,j}^{n+1} + \begin{bmatrix} p_{i+1,j}^{n}E_{i+1,j}^{n+1} + p_{i-1,j}^{n}E_{i-1,j}^{n+1} + \\ p_{i,j+1}^{n}E_{i,j+1}^{n+1} + p_{i,j-1}^{n}E_{i,j-1}^{n+1} \end{bmatrix} + \begin{bmatrix} p_{i+1,j+1}^{n}E_{i+1,j+1}^{n+1} + p_{i-1,j+1}^{n}E_{i-1,j+1}^{n+1} + \\ p_{i+1,j-1}^{n}E_{i+1,j-1}^{n+1} + p_{i-1,j-1}^{n}E_{i-1,j-1}^{n+1} \end{bmatrix} \qquad (37)$$

and subscripts i and j indicate the spatial position in the 2D domain, p is a parameter, n+1 indicates the next time step and b_{E}^{n} is a constant term always related to the current time step i.e. n. Formulation of the parameters for different variables is given in detail in Appendix A. The relaxation can then be written:

$$E^{g+1} = E^{g} - w_{sor}\frac{G\left(E^{g}\right) - b_{E}^{n}}{p_{E}^{n}} \qquad (38)$$

where g indicates the iteration step, w_{sor} is a relaxation factor corresponding to successive

over-relaxation (SOR) method and p_E^n is the prime parameter of E in the function $G(E)$. In the current study, we normally take $w_{sor} = 1$, i.e. the Gauss-Seidel method is employed.

3.2.2. Parallelization of the multigrid algorithm

During the solving process using multigrid method, the most time consuming component is the relaxation, which is especially true for large-scale problems. To reduce the computation overhead, distributed parallel computing methods were developed and incorporated into the multigrid method. As discussed in [27, 30], to achieve a high degree of parallelism, two important issues including (1) grid partitioning of the computing domain and (2) communication between adjacent processes must be addressed.

For grid partitioning, the current study adopts the same approach as mentioned in [27] i.e. the 2D rectangular computing domain (with $M \times N$ grid cells) is divided into N_p parts with each having the same number ($M / N_p \times N$) of grid cells. Here N_p denotes the number of the main computing processes, which in the current study always equals to 2 to the power of n_p, i.e. $N_p = 2^{n_p}$, where the exponent n_p is dependent on the scale of the problem. The computing process is also called the basic computing unit and in most cases stands for a central processing unit (CPU) core.

During the simulation, the newly updated data in each time step has to be communicated between neighbor processes. For the sake of efficiency, a thin overlap area of width 1 is applied for each process to copy the data located in its neighbor sub-grids, as indicated by the

first and last rows of data shown in Fig. 1b (e.g “0”, “6” for the bottom process and “4” and “10” for the upper process). In this respect, the total number of control cells owned by each process is thus $(M / N_p + 2)\acute{}\ N$. The parallelization of the multigrid method was then realized by parallelizing the three components i.e. relaxation, restriction and interpolation. Taking relaxation for instance, mechanism of data communication is shown schematically in Fig. 1b for point-wise data including phase field, solute and temperature and Fig. 1c for cell-wise data including velocity and pressure.

During relaxation, a parallel version of the line Gauss-Seidel smoother [30] was adopted for the phase field, solute and temperature. As shown in Fig. 1b, this smoother comprises two half-steps: during the first half step, the odd lines (e.g. 1, 3, 5…) of the grid points are smoothed, after which the even lines (e.g. 2, 4, 6…) are smoothed using the updated values of the neighbor grid points. In this way, each half-step of the relaxation is fully parallelized and because an overlap buffer is applied, data communication is only required at the end of the relaxation.

To solve the Navier-Stokes equation efficiently, a symmetrical coupled Gauss-Seidel (SCGS) smoothing technique [35] was employed. The key idea of the SCGS smoothing technique is to use one grid-cell of data as the basic unit for smoothing. Simply, instead of updating the value associated with one grid point during each smoothing step, all five values associated with a cell (four velocities and a pressure) are updated simultaneously. Detail of this method can be found in [35] and a brief review is given in Appendix B. However, to efficiently solve

the N-S equation, the SCGS method has to be parallelized, and a three step parallel scheme was developed. Because of the structure of the staggered grid, at least two block lines (a block line is a row of grid-cells) of data must be skipped each time the parallel calculation is performed to avoid writing and reading data at the same time, and this results in a "parallel block line" smoothing scheme. In principle, the number of the skipped block lines during each relaxation step could be any number equal to or larger than two. However, because the complexity of the parallelization also increases with this number, a skip of only two was employed. Therefore, the smoothing procedure for the staggered grid mainly comprised of three main steps. As shown in Fig. 1c, in the first step, block lines 1, 4, 7… were smoothed and during the second and third steps block lines 2, 5, 8… and 3, 6, 9… were smoothed respectively. In this way, the relaxation can be fully parallelized in each step. At the end of relaxation, all data belonging to the block line at the overlap area have to be communicated between neighbor processes.

Restriction and interpolation can be performed rather more straightforwardly, detail of which for both point-wise data (phase field, solute and temperature) and cell-wise data (velocity and pressure) is shown schematically in Fig. 2. As shown, the target value pointed by the arrows is updated by a summation of the values from neighbors multiplied by the corresponding weight factors.

3.3. Summary of the algorithm and programming considerations

Accordingly, the parallel multigrid algorithm is organized as follows

(1) Set current time step $n = 0$ and initialization of the computing domain according to each process.

(2) Solve the phase field equation i.e. Eq. (17) and get f at time step n+1 i.e. f^{n+1}.

(3) Solve the convection equations i.e. Eqs. (20) and (21) to retrieve $\tilde{u}^{n+1}$and $\tilde{v}^{n+1}$ by applying the newly updated f^{n+1} from (2) .

(4) Solve the solute and temperature equations, i.e. Eqs. (18) and (19) to retrieve U^{n+1} and θ^{n+1} by applying the newly updated f^{n+1}, $\tilde{u}^{n+1}$and $\tilde{v}^{n+1}$ from (2) and (3).

(5) Set time step $n = n$+1 and repeat (2) until the end of the simulation.

Each solving process is carried out by employing the FAS multigrid algorithm and parallel computation is realized implicitly in multigrid components including relaxation, restriction and interpolation. Besides, for all mathematical operations involving matrices and vectors, parallelization can be realized straightforwardly and no further discussion will be given here.

In addition to the parallel computing scheme, other methods were adopted to speed up the simulation. Firstly, the grid size for evaluating the N-S equations was chosen to be twice as coarse as that used for phase field, temperature and solute. Numerical tests showed that the application of these multi-grid approaches did not introduce any error larger than 1% in any of the results, but brought considerable computational benefit. Secondly, as suggested by Tong *et al.* [22], the time step for evaluating the N-S equation was also increased beyond that used for the other fields (normally two to ten times higher depending on the local growth velocity of the dendrite).

The parallel code was developed and tested on a 640-core (1.9 TB RAM) supercomputing cluster named as SAL housed at the Oxford Supercomputing Center and 8800-core supercomputing cluster housed at National Laboratory for Information Science and Technology in Tsinghua University using C++ language and two message passing libraries namely Message Passing Interface (MPI) [36] and Open Multi-Processing (OpenMP) [37]. In the program, OpenMP was used in conjunction with MPI to provide a second level of parallelism, resulting in an overall so-called hybrid/mixed program structure.

4. Model validation and numerical studies

4.1. Benchmark simulations

Fig. 3 shows the configuration of the benchmark dendrite growth scenario. The 2-D computation domain is rectangular with $M \times N$ cells. Only half of the growing dendrite was simulated because of the symmetry of the case. The dendrite grew from the midpoint of the left side of the domain with an initial solid "seed" radius of $R_0 = 30d_0$. The initial temperature of the seed was set to be zero while for the rest of the domain the temperature was set at an initial undercooling $\theta = \theta_0$. For phase field, solute and temperature, the boundary condition of all sides of the domain were set to a Neumann condition (gradient flux is zero). For convection, the horizontal velocity u was set to zero at all boundaries, while for the vertical velocity v, the boundary condition was set as follows: at the top side of the domain v was set to a constant value v_0; at the bottom, $\partial v/\partial y = 0$; and for both left and right sides, $\partial v/\partial x = 0$. A no-slip boundary condition was applied at the surface of the dendrite.

To calculate the selection constant σ^* according to Eq. (1), the dimensionless solutal undercooling Δ_C must be evaluated first. In phase field modelling the solutal undercooling can be calculated from $\Delta_C = U_i/[1+(1-k)U_i]$ [31] where U_i is the dimensionless solute concentration at the interface. U_i can be obtained either directly from the simulation or evaluated using $U_i = (-d_0(1-15\varepsilon)R_{tip}-\theta_i)/Mc_\infty$, and the difference through the two methods was always within a few percent. Here, as suggested in [28, 29], θ_i is calculated corresponding to $\phi = 0.9$ and the actual tip radius i.e. R_{tip} is used instead of the parabolic tip radius R_p. For the benchmark calculations, a thermo-solutal case with parameters $Mc_\infty = 0.1$, $k = 0.15$, $\varepsilon = 0.02$, $Le = 50$, $v_0 = 1$ W_0/τ_0, $\Delta\theta = 0.55$ and $Pr = 23.1$was assumed.

4.2. Convergence and efficiency tests

Firstly, a convergence study was performed by varying both the grid cell dimension $\Delta x=\Delta y$ and the time step dt for the benchmark case shown in Fig. 3. Because the convergence tests for the case without convection have been performed extensively in our previous work [27], here only the case with convection for the first time was considered. All the simulations were firstly developed on the Oxford University Supercomputing Centre and then performed in final stage using 48 processes (cores) on the Tsinghua supercomputing system, except for case #3 and #4 for which 96 processes were used instead because of the bigger domain employed (for these two cases, the domain sizes were 6144×3072 and 12288×6144 while for all others a domain of 3072×1536 was used). Table 1 presents key parameters used in the literature to describe the condition or state at a growing dendrite tip, relating to geometry ($v_{tip}d_0/D$, R_a/d_0, R_p/d_0), Pc, σ^*_{LGK} and σ^*_{LKT}, and shows that as different combinations of

dx=dy and *dt* were chosen, all key dendrite tip parameters remained essentially constant over the range studied, even when the time step was increased to $dt = 3.2\times\Delta x^2/(4D)$, which was 160 times bigger than the time step limit required for a stable explicit method. The results were also stable, despite the time step for the N-S equations being 5 times larger than that employed for phase field. Further, using the hybrid parallel multigrid computing scheme described here, all computations for the evolution of a single dendrite to a mature shape, in this case with $\Delta x = 0.8$, $Le = 50$ and $Pr = 23.1$, took less than three hours, despite the modest number of cores. This computing efficiency made it possible to conduct a thorough parameter study related to dendrite growth under convection in sensible times.

To illustrate further the parallel efficiency and capability of the algorithm, the benchmark simulation was repeated, but varying the number of cores (processes) from 1 to 128 and the size of the domain (#Ω) from a minimum size of 512×256 grid cells to a maximum size of 8192×8192. The computation results are summarized in terms of the relationship between calculation time and number of process N_p and are shown in Fig. 4. Here, the calculation time was defined as the time taken to compute one time step after the algorithm had reached stability, i.e. approximately the same time is consumed for each step of calculation. For each point in Fig. 4, the calculation time was effectively the sum of all the separate calculation times corresponding to the solution of Eqs. (17) – (21).

As shown in Fig. 4 and compared with the results in [27], with convection the computation time increased significantly. Generally speaking, the time used to solve the Navier-Stokes

equation i.e. Eqs. (20) – (21), even with all the speed-up methods as mentioned before, was always twice as long as the time used to solve Eqs. (17) – (19) combined. Nonetheless, the overall parallel efficiency of the algorithm was excellent. For example, for a constant size of computing domain, as shown by indicative lines L_1 and L_2, the calculation time decreased proportionally and very close to linearity with increasing process number. For L_1, significant non-linearity was established only when $N_p > 32$ while for L_2, this occurred when $N_p > 128$. This is to be expected because using "too many" processes to simulate a "small" case does not extract the benefit of this particular method. The algorithm scalability is shown for example, by indicative line L_3 that shows when N_p was fixed at 16, the calculation time increased only linearly with increasing domain size #Ω. The parallel scalability is also reflected in indicative line L_4 – before N_p reached128, the calculation time remained approximately constant when both N_p and #Ω were increased simultaneously at the same proportional rate.

5. Tip operating state of a thermo-solutal dendrite growing against fluid convection

5.1. Operating state and dendrite morphology transition

A parametric study of dendrite growth was performed using the same domain as shown in Fig. 3 and full details are given in Table 2. There were 126 simulations and five main categories of parameters were investigated, including Mc_∞, $\Delta\theta$, Le, Pr and v_∞, i.e. the main factors that could influence the growth of the dendrite tip. The spatial step was fixed at $\Delta x=\Delta y= 0.8$ and the time step at $dt = 0.8\ \Delta x^2/(4D)$. The size of the computing domain was chosen according to the parameters used in the simulation, but was mostly determined by the Lewis number and

for all simulations when Le = 200 or 500, the domain size was set to $\#\Omega$ = 4096×2048; for Le = 1000, a medium size of domain of $\#\Omega$ = 6144×3072 was applied; and for Le > 1000, a domain size of $\#\Omega$ = 8192×4096 was used. For each case the imposed external fluid flow velocity entering the top of the domain was varied from 1, 2, 5, 10, 15 to 20 with units of W_0/τ_0. All simulations were initiated by "planting" a circular solid seed with a radius of $30d_0$ at the middle of the left boundary, and the simulation was judged complete when the upstream primary dendrite arm tip reached a steady state. Other key parameters included the scaling parameter λ = 2 and the solute partition coefficient k = 0.15. A simulation (with convection) of domain size $\#\Omega$ = 4096×2048 and 48 cores typically required 3 – 3.5 hours for completion while 6 – 6.5 hours were needed for a simulation when the domain size increased to $\#\Omega$ = 8192×4096.

Of all the simulations, particular interest concerns those cases when the key thermophysical parameters approach those of practical metallic alloys, such as Le ~ 10^4 and Pr ~ 10^{-2}, which is simulation 115 in Table 2 i.e. v_∞ = 2. Fig. 5 shows the evolution of the tip velocity v_{tip} with and without convection, the parabolic tip radius R_p and the selection constants σ^* corresponding to the LKT theory as the upstream dendrite arm grows. In this case, the flow was introduced only after the dendrite tip had grown a distance of approximately 272 d_0 from the seed centre. As expected, before convection was introduced, v_{tip} of the upstream and downstream primary dendrite arms were identical, and tip radius was very similar. Once flow around the dendrite was introduced, the upstream growth velocity nearly doubled and the tip radius was reduced by 28%. All the dendrite tip parameters tended towards stable values after

the dendrite upstream tip had reached a distance of $800d_0$.

Figs. 6a and b show the corresponding contour line maps of solute concentration and temperature respectively. To the best knowledge of the authors, this type of fully coupled, non-isothermal, realistic alloy parameter simulation has not been achieved previously. As expected, as the dendrite solidified the latent heat released spread quickly over the computational domain, only weakly distorted from top-bottom symmetry by the flow. In contrast, differences in solute distribution due to convection were distinct, close to the solid/liquid interface, as shown by the inset figure in Fig. 6a. As also shown in the inset magnification of the dendrite in "Z", convection for the upstream dendrite arm significantly reduced both the thickness of the layer in which solute accumulated and reduced the peak solute concentration in the liquid to ~ $2.5c_\infty$ compared with ~ $3.4c_\infty$ for the downstream direction.

Differences in heat and solute distributions are usually explained by comparing the magnitude of Prandtl and Schmidt numbers in the simulation. The Prandtl number is the kinematic viscosity divided by the thermal diffusivity and characterizes the relative influences of conducting and fluid flow on thermal transport, while the Schmidt ($Sc = Pr \times Le$) characterizes the relative influences of the fluid flow and diffusion on solute transport. A relatively small $Pr = 0.02$ used here indicated that it was very difficult for fluid flow to influence the distribution of temperature while on the other hand, the relatively large $Sc = 200$ here indicated that fluid flow strongly influences solute profiles.

Figs. 6c and 6d show the same simulation but with Pr = 2 and Le = 200 so that Sc = 400. The 100 fold increase in Pr increased the influence of convection on temperature, with now more pronounced differences in the upstream and downstream distribution. On the other hand, doubling Sc had little effect on the solute distribution: the peak solute concentration at upstream and downstream tip was 2.5 c_∞ and 3.5 c_∞ respectively, little changed from Figs. 6a and b, and consequently there was little change in overall dendrite shape.

5.2. Influence of growth parameters

For dendrite growth in a coupled thermo-solutal field, the parameters expected to influence strongly the evolving dendrite morphology are initial solute concentration and external imposed undercooling.

To investigate the influence of these factors on dendrite growth, the calculated dendrite shape when the upstream arm tip reached the same distance of ~ 1000 d_0 from the seed center is shown in Fig. 7 for variations in Mc_∞, $\Delta\theta$ and Le, all of which directly affected the temperature – solute distribution. The default values of Mc_∞, $\Delta\theta$ and Le were 0.1, 0.55 and 200, and then keeping two of these constant, the third was varied in the range Mc_∞ = 0.06 to 0.15, $\Delta\theta$ = 0.35 to 0.65, and Le = 200 to 10^4. To better distinguish the different profiles, arrows (with hollow heads) are used in the figures to indicate the trend change in shape as the variable under consideration was increased. For each parameter set, the shape with and without convection is shown, and the inset figures show corresponding contour maps of

solute concentration.

The dendrite morphology usually comprised two types of feature: for primary arms and minor dendrite arms in-between, as the solute concentration increased in Fig. 7a, the upper primary arm became slightly wider and there was an increase in dendrite tip radius. The minor arm became more pronounced and extended at an angle of 45^{o} to the symmetry boundary, as solutal undercooling in interdendritic regions became more pronounced. When convection was present in Fig. 7b, the horizontal side and downstream primary arms were significantly shortened. As flow propagated from top to bottom, solute rejected by the growing upstream primary dendrite arm, across the solid-liquid interface, and into the local liquid was convected downwards with the flow and around the horizontal primary arm. Solute then became relatively concentrated in the more stagnant, recirculating flow on the downstream side of the side arm, as shown in the zoom-in areas in Fig. 7 (also Fig. 6). According to the phase diagram, a higher solute concentration leads to a lower liquidus temperature and solute pile-up depresses the local melting point and therefore the level of local undercooling – the driving force for solidification is reduced, resulting in shorter, less developed side and downstream arms.

Reducing the imposed undercooling in Figs. 7c and d significantly widened the primary arms, with a broadly similar effect to increasing solute concentration. The increase in undercooling greatly reduced the growth of solid in-between the primary arms. However, the opposite behaviour developed when the undercooling was increased from $\Delta\theta$ = 0.35 to 0.45, when

growth between primary dendrites became more pronounced. This behavior was accompanied by the transition in the morphology of the secondary dendrite arm structure, as indicated by "B" in Fig. 7c. For the case $\Delta\theta = 0.35$, as shown by the insets in both Figs. 7c and 7d, the packing of the solute iso-concentration contours became less dense. Interestingly, with convection present and as Mc_∞ increased in Fig. 7, the upstream part of the horizontal primary arm (indicated by "A") was stimulated to develop a broad secondary dendrite arm, growing at approximately 45^o to the symmetry axis, which was qualitatively similar to the morphology changes induced by decreasing undercooling with convection in Fig. 7d. In both cases, local solute redistribution due to the fluid flow caused these shape changes.

The influence of Lewis number on dendrite growth is shown in Figs. 7e and 7f. An increase in Lewis number was broadly equivalent to an increase in the effective undercooling and a decrease in solute concentration, leading to a decrease in tip radius of the upstream primary arm, inhibition of the minor arms between primary arms, and elongation of both side and downstream primary arms when convection was present. The overall dendrite shape for Le = 2000 (inset in Fig. 6e) was very similar to that in the inset in Fig. 7a, even though the undercooling in Fig. 7a of $\Delta\theta = 0.45$ was smaller than $\Delta\theta = 0.55$ in Fig. 7e i.e. the difference in undercooling was largely compensated by the larger Lewis number.

5.3. The selection parameter σ^*

The tip operating state of the upstream primary arm was determined for each case above, and the selection parameters with (σ^*) and without (σ_0^*) convection corresponding to both LGK

and LKT theories were calculated. Because one of the aims of the simulations was to consider if Bouissou and Pelce's theory of convection effects on dendrite tip state [20] holds for this type of simulation, Fig. 8 shows σ_0^*/σ^* as a function of $\chi_e^{11/14}$ from Eq. (5). Figs. 8a and 8b show the influence of solute concentration and imposed undercooling on σ_0^*/σ^*, while Figs. 8c and 8d show the influence of Lewis number (*Le*) and Prandtl number (*Pr*). In all the figures the magnitude of the imposed fluid flow velocity entering the domain was varied from 0, 1, 2, 5, 10, 15 to 20 in units of W_0/τ_0. Since $W_0/\tau_0 = Da_1/(\lambda^2 d_0 a_2)$, then the magnitude of the velocity is linked to the scaling parameter λ. Constant parameters of $\varepsilon = 0.02$, $\lambda = 2$, $k = 0.15$, $\Delta\theta = 0.55$, Mc_∞=0.10 and $Le = 200$ were used in all the situations except where stated in the figure. For Figs. 8c and 8d, a smaller undercooling i.e. $\Delta\theta = 0.45$ was used to ensure $W_0 v_{tip}/D < 1$ (required for phase field simulation to be accurate).

Fig. 8a shows the selection parameter corresponding to LGK theory i.e. σ^*_{LGK} calculated from the fully coupled phase field simulations with and without convection were similar when $\chi_e^{11/14} < 0.2$ i.e. $\sigma^*_0/\sigma^* \sim 1$. Above 0.2 σ^*_0/σ^* increased. The most significant increase of σ^*_0/σ^* with $\chi_e^{11/14}$ occurred for the lowest value of $\Delta\theta = 0.35$ because this relatively low undercooling made the dendrite more sensitive to the influence of convection, as previously described. As undercooling was then increased from 0.35 to 0.55 convection had less influence on σ^*_{LGK}. Taking $\Delta\theta = 0.55$ for instance, $\sigma^*_0/\sigma^* \sim 1$ even when $\chi_e^{11/14} > 1$. However, a further increase of $\Delta\theta$ from 0.55 to 0.75 led to a progressing increase in σ^*_0/σ^* σ^*_{LGK}. Interestingly, there was a similar trend with an increase in solute concentration $Mc_\infty - \sigma^*_0/\sigma^*$ first decreased and then increased, and there was a certain solute concentration beyond which

σ^*_{LGK} became relatively insensitive to fluid flow. This may be explained as follows. When the imposed undercooling was comparatively small ($\Delta\theta$ = 0.35), the flow reduced the temperature at the dendrite tip when *Le* = 200, which promoted dendrite growth and decreased the selection parameter. As $\Delta\theta$ was increased ($\Delta\theta$ = 0.55), dendrite tip growth was promoted, reducing the sensitivity of local temperature on the flow condition. As the imposed undercooling was further increased ($\Delta\theta$ = 0.75), the dendrite tip grows still faster but more solute must be rejected into the liquid. The influence of flow at these higher concentrations again becomes effective, promoting dendrite growth and decreasing the selection parameter.

The selection parameter corresponding to the LKT theory σ^*_{LKT} in Fig. 8b varied similarly to σ^*_{LGK}, although the critical value of $\chi_e^{11/14}$ beyond which there was a significant increase in σ^*_0/σ^* was higher at ~ 0.4. For both selection parameters there was little difference with and without convection below a critical value of $\chi_e^{11/14}$, which agreed well with the theory due to Bouissou and Pelce [20]. In some cases, this regime even extended to $\chi_e^{11/14} > 1$.

More insight was obtained by investigating the change of the selection parameters when the Lewis number was varied. As shown in Fig. 8c, an increase in Lewis number significantly reduced the dependence of σ^*_{LGK} on χ_e, and the higher the *Le*, the more delayed the transition from $\sigma^*_0 \sim \sigma^*$ to $\sigma^*_0 > \sigma^*$. As discussed before, the increase of *Le* is equivalent to an increase on the effective undercooling and as shown in Fig. 8a, this decreases the dependence of σ^*_{LGK} on χ_e. Changing the Prandtl number did not produce a strong change in the transition behaviour of σ^*_{LGK} as shown in Fig. 8c, although a decrease of *Pr* from 2.0 to 0.02 (*Le* = 10^4)

significantly increased the effect of convection which can be seen by comparing the maximum value of χ_e from both cases. As shown in Fig. 8d, once $\chi_e^{11/14}$ is greater than 0.2, an increase of the Lewis number to > 2000 decreased $\sigma_0^*{}_{LKT}/\sigma^*{}_{LKT}$ to less than unity.

6. Conclusions

Free dendritic growth against forced convection has been investigated using phase field modeling in which fully coupled and non-linear equations for heat, solute and liquid flow were solved using a novel parallel-multigrid numerical approach. Convergence and efficiency tests confirmed that this approach was very robust and enabled the solution of coupled dendrite growth during solidification of alloys of practical interest with $Le \sim 10^4$ and $Pr \sim 10^{-2}$. Dendrite growth dynamics were studied by investigating the influence of growth related parameters such as solute concentration, undercooling and Lewis number on dendrite morphology and tip operating state. The selection constants with convection (σ^*) and without convection (σ_0^*) according to both LGK and LKT theories were calculated using the model, and their ratio σ_0^*/σ^* was then compared with the theory. The ratio σ_0^*/σ^* remained close to unity when χ_e was small and increased significantly beyond a critical value of χ_e, which agreed well with theory.

However, the typical magnitude of the product $Pe_f\sigma^*$ in the present work was around 0.005 – 0.035 because $\sigma^* \sim 0.05$ and $Pe_f \sim 0.1 – 0.7$. This is much lower than $Pe_f\sigma^* \sim 1$ used by Bouissou and Pelce who only considered the temperature-flow field since they studied only pure materials. Therefore, it is proposed that in the present work, recognizing the dominant

role of solute distributions in the liquid in the evolving morphology, the appropriate Peclet number is supposed to be not $Pe_f = R_{tip}v_{tip}/2\alpha$ but $Pe_f = R_{tip}v_{tip}/2D$. When this form is used, the range of Peclet number studied in the present work lies in the range spanning unity, and the insight provided by the model can therefore be even more closely reconciled with the work of Bouissou and Pelce [20].

Nonetheless, for the fully coupled simulations, the Peclet number is dependent on solute concentration, undercooling and Lewis number, which cannot be accounted for by Bouissou and Pelce's approach. In this respect, and in spite of the complexity of model formulation, the present computational approach provides a more flexible method to explore factors governing dendrite evolution under flow conditions. Using this flexibility and performing a large number of different simulations revealed that the basic approach of Bouissou and Pelce provided surprisingly robust validity.

The extension of the present work to three dimensions requires further development of the numerical algorithm. A promising approach is to incorporate the adaptive mesh refinement (AMR) into the current parallel multigrid computing scheme. Because the multigrid algorithm is intrinsically composed of grids with different sizes, the incorporation of the AMR is theoretically straightforward. However, careful attention must be paid on the load balance during data communication to ensure the highest parallel efficiency.

Acknowledgement

The authors would like to thank the Natural Science Foundation of China (Project No. 51205229), UK Royal Academy of Engineering/Royal Society through the Newton International Fellowship Scheme, and the EPSRC Centre for Innovative Manufacture: Liquid Metal Engineering (EP/H026177/1) for financial support, and the Oxford Supercomputer Centre and the National Laboratory for Information Science and Technology in Tsinghua University for access to supercomputing facilities and support for parallel programming. The authors would also like to thank the innovation platform for through process modeling and simulation of advanced materials processing technologies project in China for access to research facilities and financial support.

Appendix A

(1) For phase field f, The related parameters in Eqs. (36) and (37) are:

$$p_{i,j}^{n} = \left[1/Le + Mc_{\infty}\left(1+(1-k)U_{i,j}^{n}\right)\right]B_{y}^{n} + \beta\Delta t\left[\frac{10}{3}\frac{B_{y}^{n}}{h^{2}} - 1 + 3\left(f_{i,j}^{n}\right)^{2} - 4/f_{i,j}^{n}\left(1-\left(f_{i,j}^{n}\right)^{2}\right)\left(q_{i,j}^{n} + Mc_{\infty}U_{i,j}^{n}\right)\right] \quad (39)$$

$$p_{i+1,j}^{n} = 4p_{f0} - \frac{\beta\Delta t}{2h}B_{1f}^{n} \quad (40)$$

$$p_{i-1,j}^{n} = 4p_{f0} + \frac{\beta\Delta t}{2h}B_{1f}^{n} \quad (41)$$

$$p_{i,j+1}^{n} = 4p_{f0} - \frac{\beta\Delta t}{2h}B_{2f}^{n} \quad (42)$$

$$p_{i,j-1}^{n} = 4p_{f0} + \frac{\beta\Delta t}{2h}B_{2f}^{n} \quad (43)$$

$$p_{i+1,j-1}^{n} = p_{i+1,j+1}^{n} = p_{i-1,j-1}^{n} = p_{i-1,j+1}^{n} = p_{f0} \quad (44)$$

$$p_{f0} = \frac{-\beta \Delta t}{6h^2} B_y^n \qquad (45)$$

$$b_f^n = -\left[1/Le + Mc_\infty \left(1 + (1-k) U_{i,j}^n\right)\right]\left[a_0 f_{i,j}^n + a_1 f_{i,j}^{n-1}\right] \qquad (46)$$

where

$$B_{1f}^n = 2A_y A_y' y_x' + \left[\left(A_y'\right)^2 + A_y A_y''\right] y_y' \qquad (47)$$

$$B_{2f}^n = 2A_y A_y' y_y' - \left[\left(A_y'\right)^2 + A_y A_y''\right] y_x' \qquad (48)$$

$$B_y^n = \left(A_y^n\right)^2 \qquad (49)$$

$$A_y^n = A\left(y^n\right) \qquad (50)$$

and

$$A_y' = \left.\frac{\partial A(y)}{\partial y}\right|_{y^n} \qquad (51)$$

$$A_y'' = \left.\frac{\partial^2 A(y)}{\partial y^2}\right|_{y^n} \qquad (52)$$

$$y_x' = \frac{\partial y}{\partial x} = \frac{f_x' f_{xy}'' - f_y' f_{xx}''}{\left|\nabla f\right|^2} \qquad (53)$$

$$y_y' = \frac{\partial y}{\partial y} = \frac{f_x' f_{yy}'' - f_y' f_{xy}''}{\left|\nabla f\right|^2} \qquad (54)$$

$$f_{xx}'' = \frac{\partial^2 f}{\partial x^2} \qquad (55)$$

$$f_{yy}'' = \frac{\partial^2 f}{\partial y^2} \qquad (56)$$

$$f_{xy}'' = \frac{\partial^2 f}{\partial x \partial y} \qquad (57)$$

(2) For the Navier-Stokes equations and according to the staggered grid structure, let the

related variables of a grid cell (with indexes *i* and *j*) be $\tilde{P}_{i,j}^{n+1}$, $\tilde{u}_{i-1/2,j}^{n+1}$, $\tilde{u}_{i+1/2,j}^{n+1}$, $\tilde{v}_{i,j-1/2}^{n+1}$ and $\tilde{v}_{i,j+1/2}^{n+1}$. Eq. (37) is then written (taking $\tilde{u}_{i-1/2,j}^{n+1}$ for instance),

$$G\left(\tilde{u}_{i-1/2,j}^{n+1}\right)=\left(p_C^n\right)_{i-1/2,j}\tilde{u}_{i-1/2,j}^{n+1}-\begin{bmatrix}\left(p_W^n\right)_{i-1/2,j}\tilde{u}_{i-3/2,j}^{n+1}+\left(p_E^n\right)_{i-1/2,j}\tilde{u}_{i+1/2,j}^{n+1}+\\ \left(p_S^n\right)_{i-1/2,j}\tilde{u}_{i-1/2,j-1}^{n+1}+\left(p_N^n\right)_{i-1/2,j}\tilde{u}_{i-1/2,j+1}^{n+1}\end{bmatrix}=\left(b_u^n\right)_{i-1/2,j} \quad (58)$$

the related parameters are

$$\left(p_C^n\right)_{i-1/2,j}=1+\frac{\beta\Delta t}{h}\left[\left(B_E^n\right)_{i-1/2,j}+\left(B_W^n\right)_{i-1/2,j}+\left(B_N^n\right)_{i-1/2,j}+\left(B_S^n\right)_{i-1/2,j}\right]+\frac{4\beta\Delta t}{h^2}\tilde{\nu}+\beta\Delta t C_{i-1/2,j}^n \quad (59)$$

$$\left(p_E^n\right)_{i-1/2,j}=\frac{\beta\Delta t}{h}\left(\frac{\tilde{u}}{h}+\left(B_E^n\right)_{i-1/2,j}\right) \quad (60)$$

$$\left(p_W^n\right)_{i-1/2,j}=\frac{\beta\Delta t}{h}\left(\frac{\tilde{u}}{h}+\left(B_W^n\right)_{i-1/2,j}\right) \quad (61)$$

$$\left(p_N^n\right)_{i-1/2,j}=\frac{\beta\Delta t}{h}\left(\frac{\tilde{u}}{h}+\left(B_N^n\right)_{i-1/2,j}\right) \quad (62)$$

$$\left(p_S^n\right)_{i-1/2,j}=\frac{\beta\Delta t}{h}\left(\frac{\tilde{u}}{h}+\left(B_S^n\right)_{i-1/2,j}\right) \quad (63)$$

$$\left(p_P^n\right)_{i-1/2,j}=\frac{\beta\Delta t}{h}\left(f_l^n\right)_{i-1/2,j} \quad (64)$$

$$\left(b_u^n\right)_{i-1/2,j}=-\left(a_0\tilde{u}_{i-1/2,j}^n+a_1\tilde{u}_{i-1/2,j}^{n-1}\right)-\left(p_P^n\right)_{i-1/2,j}\left(\tilde{P}_{i,j}^n-\tilde{P}_{i-1,j}^n\right) \quad (65)$$

where

$$\left(B_E^n\right)_{i-1/2,j}=-\frac{1}{2}\frac{\tilde{u}_{i-1/2,j}^n}{\left(f_l^n\right)_{i-1/2,j}}\left(1-\operatorname{sgn}\left(\tilde{u}_{i-1/2,j}^n\right)\right) \quad (66)$$

$$\left(B_W^n\right)_{i-1/2,j}=\frac{1}{2}\frac{\tilde{u}_{i-1/2,j}^n}{\left(f_l^n\right)_{i-1/2,j}}\left(1+\operatorname{sgn}\left(\tilde{u}_{i-1/2,j}^n\right)\right) \quad (67)$$

$$\left(B_N^n\right)_{i-1/2,j}=-\frac{1}{2}\frac{\tilde{v}_{i-1/2,j}^n}{\left(f_l^n\right)_{i-1/2,j}}\left(1-\operatorname{sgn}\left(\tilde{v}_{i-1/2,j}^n\right)\right) \quad (68)$$

$$\left(B_S^n\right)_{i-1/2,j}=\frac{1}{2}\frac{\tilde{v}_{i-1/2,j}^n}{\left(f_l^n\right)_{i-1/2,j}}\left(1+\operatorname{sgn}\left(\tilde{v}_{i-1/2,j}^n\right)\right) \quad (69)$$

$$C^n_{i-1/2,j} = 2h_c\tilde{u}\left(1 - \left(f_l^n\right)_{i-1/2,j}\right)^2 - \left[\frac{1}{\left(f_l^n\right)_{i-1/2,j}}\right]^2\left(\tilde{u}^n_{i-1/2,j}\left.\frac{\partial f_l}{\partial x}\right|_{i-1/2,j} + \tilde{v}^n_{i-1/2,j}\left.\frac{\partial f_l}{\partial y}\right|_{i-1/2,j}\right) \qquad (70)$$

and

$\left(\tilde{P}^n_{i,j} - \tilde{P}^n_{i-1,j}\right)$ is the pressure correction. Also

$$\tilde{v}^n_{i-1/2,j} = \frac{\tilde{v}^n_{i,j-1/2} + \tilde{v}^n_{i-1,j-1/2} + \tilde{v}^n_{i,j+1/2} + \tilde{v}^n_{i-1,j+1/2}}{4} \qquad (71)$$

(3) For solute *U*, the related parameters are

$$p^n_{i,j} = \left[\frac{1+k}{2} - \frac{1-k}{2}f^{n+1}_{i,j}\right] - \beta\Delta t \cdot B^n_U + \frac{5\beta\Delta t\tilde{D}}{3h^2}\left(1 - f^{n+1}_{i,j}\right) \qquad (72)$$

$$p^n_{i+1,j} = 4p_{U0} - \frac{\beta\Delta t}{2h}B^n_{1U} \qquad (73)$$

$$p^n_{i-1,j} = 4p_{U0} + \frac{\beta\Delta t}{2h}B^n_{1U} \qquad (74)$$

$$p^n_{i,j+1} = 4p_{U0} - \frac{\beta\Delta t}{2h}B^n_{2U} \qquad (75)$$

$$p^n_{i,j-1} = 4p_{U0} + \frac{\beta\Delta t}{2h}B^n_{2U} \qquad (76)$$

$$p^n_{i+1,j-1} = p^n_{i+1,j+1} = p^n_{i-1,j-1} = p^n_{i-1,j+1} = p_{U0} \qquad (77)$$

$$p_{U0} = \frac{-\beta\Delta t\tilde{D}\left(1 - f^{n+1}_{i,j}\right)}{12h^2} \qquad (78)$$

$$b^n_f = -\left[\frac{1+k}{2} - \frac{1-k}{2}f^{n+1}_{i,j}\right]\left(a_0U^n_{i,j} + a_1U^{n-1}_{i,j}\right) + \beta\Delta tC^n_U \qquad (79)$$

where

$$B^n_U = (1-k)D^n_U \qquad (80)$$

$$B_{1U}^{n} = \left[-\frac{1}{2}\tilde{D}f_{x}^{'} + \frac{1-k}{2\sqrt{2}} f_{t}^{'} \frac{f_{x}^{'}}{|\nabla f|} \right]_{i,j} - \tilde{u}_{i,j}^{n+1} \left[\frac{1+k}{2} - \frac{1-k}{2} f_{i,j}^{n+1} \right] \qquad (81)$$

$$B_{2U}^{n} = \left[-\frac{1}{2}\tilde{D}f_{y}^{'} + \frac{1-k}{2\sqrt{2}} f_{t}^{'} \frac{f_{y}^{'}}{|\nabla f|} \right]_{i,j} - \tilde{v}_{i,j}^{n+1} \left[\frac{1+k}{2} - \frac{1-k}{2} f_{i,j}^{n+1} \right] \qquad (82)$$

$$C_{U}^{n} = D_{U}^{n} + \frac{1}{2}\left(\tilde{u}_{i,j}^{n+1} f_{x}^{'}\Big|_{i,j} + \tilde{v}_{i,j}^{n+1} f_{y}^{'}\Big|_{i,j} \right) \qquad (83)$$

$$D_{U}^{n} = \frac{1}{2\sqrt{2}} \left[\frac{\partial}{\partial x}\left(f_{t}^{'} \frac{f_{x}^{'}}{|\nabla f|} \right) + \frac{\partial}{\partial y}\left(f_{t}^{'} \frac{f_{y}^{'}}{|\nabla f|} \right) \right]_{i,j} + \frac{1}{2} f_{t}^{'}\Big|_{i,j} \qquad (84)$$

$$f_{t}^{'}\Big|_{i,j} = \left(\frac{\partial f}{\partial t} \right)_{i,j} = \frac{f_{i,j}^{n+1} - f_{i,j}^{n}}{\Delta t} \qquad (85)$$

(4) For temperature q the parameters are

$$p_{i,j}^{n} = 1 + \frac{10\beta\Delta t}{3h^{2}} \tilde{a} \qquad (86)$$

$$p_{i+1,j}^{n} = 4p_{q0} + \frac{\beta\Delta t}{2h} \tilde{u}_{i,j}^{n+1} \qquad (87)$$

$$p_{i-1,j}^{n} = 4p_{q0} - \frac{\beta\Delta t}{2h} \tilde{u}_{i,j}^{n+1} \qquad (88)$$

$$p_{i,j+1}^{n} = 4p_{q0} + \frac{\beta\Delta t}{2h} \tilde{v}_{i,j}^{n+1} \qquad (89)$$

$$p_{i,j-1}^{n} = 4p_{q0} - \frac{\beta\Delta t}{2h} \tilde{v}_{i,j}^{n+1} \qquad (90)$$

$$p_{i+1,j-1}^{n} = p_{i+1,j+1}^{n} = p_{i-1,j-1}^{n} = p_{i-1,j+1}^{n} = p_{q0} \qquad (91)$$

$$p_{q0} = \frac{-\beta\Delta t}{6h^{2}} \tilde{a} \qquad (92)$$

$$b_{q}^{n} = \frac{1}{2}\beta\Delta t f_{t}^{'}\Big|_{i,j} - \left(a_{0} q_{i,j}^{n} + a_{1} q_{i,j}^{n-1} \right) \qquad (93)$$

Appendix B

Let the related variables of a grid cell (with indexes *i* and *j*) be $\tilde{P}_{i,j}^{n+1}$, $\tilde{u}_{i-1/2,j}^{n+1}$, $\tilde{u}_{i+1/2,j}^{n+1}$, $\tilde{v}_{i,j-1/2}^{n+1}$ and $\tilde{v}_{i,j+1/2}^{n+1}$. For the SCGS method, during relaxation all five variables in one grid cell will be updated simultaneously, which is equivalent to solve

$$A_p^n \cdot Z^{n+1} = \hat{b}^n \qquad (94)$$

where

$$A_p^n = \begin{bmatrix} \left(p_C^n\right)_{i-1/2,j} & -\left(p_E^n\right)_{i-1/2,j} & 0 & 0 & \left(p_P^n\right)_{i-1/2,j} \\ -\left(p_W^n\right)_{i+1/2,j} & \left(p_C^n\right)_{i+1/2,j} & 0 & 0 & -\left(p_P^n\right)_{i+1/2,j} \\ 0 & 0 & \left(p_C^n\right)_{i,j-1/2} & -\left(p_N^n\right)_{i,j-1/2} & \left(p_P^n\right)_{i,j-1/2} \\ 0 & 0 & -\left(p_S^n\right)_{i,j+1/2} & \left(p_C^n\right)_{i,j+1/2} & -\left(p_P^n\right)_{i,j+1/2} \\ -1 & 1 & -1 & 1 & 0 \end{bmatrix} \qquad (95)$$

$$Z^{n+1} = \left[\tilde{u}_{i-1/2,j}^{n+1}, \tilde{u}_{i+1/2,j}^{n+1}, \tilde{v}_{i,j-1/2}^{n+1}, \tilde{v}_{i,j+1/2}^{n+1}, \tilde{P}_{i,j}^{n+1}\right]^{\mathrm{T}} \qquad (96)$$

$$\hat{b}^n = \left[b_{i-1/2,j}^n, b_{i+1/2,j}^n, b_{i,j-1/2}^n, b_{i,j+1/2}^n, 0\right]^{\mathrm{T}} \qquad (97)$$

and

$$b_{i-1/2,j}^n = -\left(a_0\tilde{u}_{i-1/2,j}^n + a_1\tilde{u}_{i-1/2,j}^{n-1}\right) + \left(p_P^n\right)_{i-1/2,j}\tilde{P}_{i-1,j}^n + \begin{bmatrix} \left(p_W^n\right)_{i-1/2,j}u_{i-3/2,j}^n + \left(p_N^n\right)_{i-1/2,j}u_{i-1/2,j+1}^n \\ +\left(p_S^n\right)_{i-1/2,j}u_{i-1/2,j-1}^n \end{bmatrix} \qquad (98)$$

$$b_{i+1/2,j}^n = -\left(a_0\tilde{u}_{i+1/2,j}^n + a_1\tilde{u}_{i+1/2,j}^{n-1}\right) - \left(p_P^n\right)_{i+1/2,j}\tilde{P}_{i+1,j}^n + \begin{bmatrix} \left(p_E^n\right)_{i+1/2,j}u_{i+3/2,j}^n + \left(p_N^n\right)_{i+1/2,j}u_{i+1/2,j+1}^n \\ +\left(p_S^n\right)_{i+1/2,j}u_{i+1/2,j-1}^n \end{bmatrix} \qquad (99)$$

$$b_{i,j-1/2}^n = -\left(a_0\tilde{v}_{i,j-1/2}^n + a_1\tilde{v}_{i,j-1/2}^{n-1}\right) + \left(p_P^n\right)_{i,j-1/2}\tilde{P}_{i,j-1}^n + \begin{bmatrix} \left(p_E^n\right)_{i,j-1/2}\tilde{v}_{i+1,j-1/2}^n + \left(p_W^n\right)_{i,j-1/2}\tilde{v}_{i-1,j-1/2}^n \\ +\left(p_S^n\right)_{i,j-1/2}\tilde{v}_{i,j-3/2}^n \end{bmatrix} \qquad (100)$$

$$b^{n}_{i,j+1/2} = -\left(a_0 \tilde{v}^{n}_{i,j+1/2} + a_1 \tilde{v}^{n-1}_{i,j+1/2}\right) - \left(p^{n}_{P}\right)_{i,j+1/2} \tilde{P}^{n}_{i,j+1} + \left[\begin{array}{l} \left(p^{n}_{E}\right)_{i,j+1/2} \tilde{v}^{n}_{i+1,j+1/2} + \left(p^{n}_{W}\right)_{i,j+1/2} \tilde{v}^{n}_{i-1,j+1/2} \\ + \left(p^{n}_{N}\right)_{i,j+1/2} \tilde{v}^{n}_{i,j+3/2} \end{array}\right] \quad (101)$$

where the explicit formulation of the parameters in Eqs. (98) to (101) is given in Appendix A. Eq. (94) can be easily solved using the Gaussian elimination method.

References

[1] M. Asta, C. Beckermann, A. Karma, W. Kurz, R. Napolitano, M. Plapp, G. Purdy, M. Rappaz, R. Trivedi, Solidification microstructures and solid-state parallels: Recent developments, future directions, Acta Materialia, 57 (2009) 941-971.

[2] E. Ben-Jacob, P. Garik, The formation of patterns in non-equilibrium growth, Nature, 343 (1990) 523-530.

[3] L. Granasy, T. Pusztai, T. Borzsonyi, J.A. Warren, J.F. Douglas, A general mechanism of polycrystalline growth, Nat Mater, 3 (2004) 645-650.

[4] T. Haxhimali, A. Karma, F. Gonzales, M. Rappaz, Orientation selection in dendritic evolution, Nat Mater, 5 (2006) 660-664.

[5] W. Kurz, D.J. Fisher, Fundamentals of Solidification, Fourth ed., Trans Tech Publications Ltd, 1998.

[6] G.P. Ivantsov, Temperature field around spherical, cylindrical and needle-shaped crystals which grow in supercooled melt, Doklady Akademiya Nauk SSR, 58 (1947) 567.

[7] J.S. Langer, H. Muller-Krumbhaar, Theory of dendritic growth - I. elements and stability analysis, Acta Metallurgica et Materialia, 26 (1978) 1681.

[8] W.W. Mullins, R.F. Sekerka, Stability of a planar interface during solidification of a dilute binary alloy, J. Appl. Phys., 35 (1964) 444.

[9] E. Ben-Jacob, N. Goldenfeld, J.S. Langer, G. Schon, Dynamics of Interfacial Pattern Formation, Physical Review Letters, 51 (1983) 1930-1932.

[10] E. Ben-Jacob, N. Goldenfeld, J.S. Langer, G. Schon, Boundary-layer model of pattern formation in solidification, Physical Review A, 29 (1984) 330-340.

[11] R.C. Brower, D.A. Kessler, J. Koplik, H. Levine, Geometrical models of interface evolution, Physical Review A, 29 (1984) 1335-1342.

[12] D.A. Kessler, J. Koplik, H. Levine, Pattern selection in fingered growth phenomena, Advances in Physics, 37 (1988) 255-339.

[13] D.A. Kessler, H. Levine, Velocity selection in dendritic growth, Physical Review B, 33 (1986) 7867-7870.

[14] J. Dantzig, M. Rappaz, Solidification, EPFL Press, Lausanne, Switzerland, 2009.

[15] J. Lipton, M.E. Glicksman, W. Kurz, Equiaxed dendrite growth in alloys at small supercooling, Metallurgical Transactions A, 18 (1987) 341-345.

[16] J. Lipton, W. Kurz, R. Trivedi, Rapid dendrite growth in undercooled alloys, Acta Metallurgica, 35 (1987) 957-964.

[17] J.-H. Jeong, J. Dantzig, N. Goldenfeld, Dendritic growth with fluid flow in pure materials, Metallurgical and Materials Transactions A, 34 (2003) 459-466.

[18] R. Ananth, W.N. Gill, The effect of convection on axisymmetric parabolic dendrites, Chemical Engineering Communications, 68 (1988) 1-14.

[19] D.A. Saville, P.J. Beaghton, Growth of needle-shaped crystals in the presence of convection, Physical Review A, 37 (1988) 3423-3430.

[20] P. Bouissou, P. Pelce, Effect of a forced flow on dendritic growth, Physical Review A, 40 (1989) 6673-6680.

[21] C. Beckermann, H.J. Diepers, I. Steinbach, A. Karma, X. Tong, Modeling Melt Convection in Phase-Field Simulations of Solidification, Journal of Computational Physics, 154 (1999) 468-496.

[22] X. Tong, C. Beckermann, A. Karma, Q. Li, Phase-field simulations of dendritic crystal growth in a forced flow, Physical Review E, 63 (2001) 061601.

[23] Y. Lu, C. Beckermann, J.C. Ramirez, Three-dimensional phase-field simulations of the effect of convection on free dendritic growth, Journal of Crystal Growth, 280 (2005) 320-334.

[24] C.W. Lan, C.J. Shih, Phase field simulation of non-isothermal free dendritic growth of a binary alloy in a forced flow, Journal of Crystal Growth, 264 (2004) 472-482.

[25] A. Karma, Phase-Field Formulation for Quantitative Modeling of Alloy Solidification, Physical Review Letters, 87 (2001) 115701.

[26] R. Siquieri, H. Emmerich, Phase-field investigation of microstructure evolution under the influence of convection, Philosophical Magazine, 91 (2011) 45 - 73.

[27] Z. Guo, J. Mi, P.S. Grant, An implicit parallel multigrid computing scheme to solve coupled thermal-solute phase-field equations for dendrite evolution, Journal of Computational Physics, 231 (2012) 1781-1796.

[28] A. Karma, W.-J. Rappel, Quantitative phase-field modeling of dendritic growth in two and three dimensions, Physical Review E, 57 (1998) 4323.

[29] A. Karma, W.-J. Rappel, Phase-field model of dendritic sidebranching with thermal noise, Physical Review E, 60 (1999) 3614-3625.

[30] U. Trottenberg, C. Oosterlee, A. Schuller, Multigrid, Academic Press., Harcourt place, 32 Jamestown Road, London NW1 7BY, UK, 2001.

[31] J.C. Ramirez, C. Beckermann, Examination of binary alloy free dendritic growth theories with a phase-field model, Acta Materialia, 53 (2005) 1721-1736.

[32] J.C. Ramirez, C. Beckermann, A. Karma, H.J. Diepers, Phase-field modeling of binary alloy solidification with coupled heat and solute diffusion, Physical Review E, 69 (2004) 051607.

[33] A. Kumar, Isotropic finite-differences, Journal of Computational Physics, 201 (2004) 109-118.

[34] S.V. Patankar, Numerical heat transfer and fluid flow, Hemisphere Publishing Corporation, The United States, 1980.

[35] S.P. Vanka, Block-implicit multigrid solution of Navier-Stokes equation in primitive variables, Journal of Computational Physics, 65 (1986) 138-158.

[36] W. Gropp, E. Lusk, A. Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, second ed., MIT Press, Cambridge, 1999.

[37] B. Chapman, G. Jost, R. van der Pas, Using OpenMP Portable Shared Memory Parallel Programming, MIT Press, Cambridge, 2007.

Figure Captions:

Figure 1: Schematic illustration of (a) grid partitioning approach used in solving the discretized equations, and the data communication mechanism with respect to the (b) point-wise data including phase field, solute and temperature, and (c) cell-wise data including velocity and pressure.

Figure 2: Schematic illustration of (a) restriction and (b) interpolation procedures used in the multigrid algorithm for both types of grids.

Figure 3: Configuration and boundary conditions for the benchmark case of a dendrite growing freely in a flow from top to bottom of the computational domain.

Figure 4: Calculation time per step for the benchmark dendrite growth case as a function of process or core number, indicating excellent computing efficiency and scalability. The lines L_1 – L_4 indicate different data-sets, where L_1 and L_2 are at constant size and varying process number; L_3 is at constant process number and increasing grid size; L_4 is at constant calculation time and increasing grid size and process number in the same proportion.

Figure 5: The variation of key dendrite parameters of parabolic tip radius R_p, selection constant σ^*_{LKT} and the tip velocity v_{tip} as a function of the growing tip position during the phase field simulation, for the case of $Le = 10^4$ and $Pr = 0.02$. Both stagnant and with forced convection are shown.

Figure 6: Dendrite shape under coupled thermal-solutal-convection conditions, showing contours of (a) solute concentration and (b) temperature corresponding to $Le = 10^4$ and $Pr = 0.02$; and (c) and (d) solute concentration and temperature respectively corresponding to Le = 200 and $Pr = 2.0$. Insets show the distribution of the solute along the mid-line of the dendrite from upstream to downstream.

Figure 7: Superimposed phase field simulations of dendrite morphology with arrows (hollow head) indicating the direction of changing shape with an increase of the parameter under study, for (a) and (b) solute concentration, (c) and (d) undercooling and (e) and (f) Lewis number, without and with convection. Indicative insets show solute iso-concentration variations.

Figure 8: Variation of dendrite selection constants as a function of the dimensionless parameter χ_e. Influences of solute concentration and undercooling are shown in (a) and (b) corresponding to the LGK and LKT theories, respectively. Influences of the Lewis number and Prandtl number are shown in (c) and (d) corresponding to the LGK and LKT theories, respectively.

Figure 9: The variation of the solutal Peclet number Pe_c as a function of the flow Peclet number Pe_f for (a) changes in solute concentration and undercooling, and (b) changes in the Lewis number.